

## 70-532.microsoft

Number: 70-532  
Passing Score: 800  
Time Limit: 120 min



<https://www.gratisexam.com/>

## Question Set 1

### QUESTION 1

You store data in an Azure blob. Data accumulates at a rate of 0.10 GB per day.

You must use storage analytics data to verify that the service level agreement (SLA) has been met and to analyze the performance of VHDs, including the pattern of usage.

Analytics data must be deleted when it is older than 100 days or when the total amount of data exceeds 10 GB.

You need to configure storage analytics and access the storage analytics data.

Which two approaches will achieve the goal? Each correct answer presents part of the solution.



- A. Disable the data retention policy.
- B. Access analytics data by using the Service Management REST API
- C. Access analytics data by using the APIs used to read blob and table data.
- D. Configure a data retention policy of 100 days.

**Correct Answer:** CD

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

### QUESTION 2

HOTSPOT

You store JSON data in a blob by using the Azure Blob service. Web applications access the JSON data by using client-side JavaScript calls.

JSON data is stored in a container that is configured to allow anonymous access. Web applications that are allowed to make updates to the data have access to any necessary shared access signatures (SASs) and storage keys.

You configure one Cross-Origin Resource Sharing (CORS) rule for the https://fabrikam.com domain and then run the following method. Line numbers are provided

<https://www.gratisexam.com/>

for reference only.

```
01 void ConfigureBlobCorsRules(CloudBlobClient blobClient)
02 {
03     var blobServiceProperties = blobClient.GetServiceProperties();
04     var partnerCorsRule = new CorsRule();
05     partnerCorsRule.AllowedOrigins.Add("https://contoso.com");
06     partnerCorsRule.AllowedMethods = CorsHttpMethods.Post | CorsHttpMethods.Put;
07     partnerCorsRule.ExposedHeaders.Add("");
08     partnerCorsRule.AllowedHeaders.Add("");
09     blobServiceProperties.Cors.CorsRules.Add(partnerCorsRule);
10     var publicCorsRule = new CorsRule();
11     publicCorsRule.AllowedOrigins.Add("");
12     publicCorsRule.AllowedMethods = CorsHttpMethods.Get;
13     publicCorsRule.ExposedHeaders.Add("");
14     publicCorsRule.AllowedHeaders.Add("");
15     blobServiceProperties.Cors.CorsRules.Add(publicCorsRule);
16     blobClient.SetServiceProperties(blobServiceProperties);
17 }
```

For each of the following statements, select Yes if the statement is true. Otherwise, select No.

Hot Area:

**Answer Area**

	Yes	No
The CORS rule that was previously configured for https://fabrikam.com is no longer in effect after this method runs.	<input type="radio"/>	<input type="radio"/>
Partners from the https://contoso.com domain can access the configured storage by using the <b>HTTP HEAD</b> operation.	<input type="radio"/>	<input type="radio"/>
Partners from the https://contoso.com domain can access the configured storage service by using the <b>HTTP GET</b> operation.	<input type="radio"/>	<input type="radio"/>

Correct Answer:

### Answer Area

	Yes	No
The CORS rule that was previously configured for https://fabrikam.com is no longer in effect after this method runs.	<input checked="" type="radio"/>	<input type="radio"/>
Partners from the https://contoso.com domain can access the configured storage by using the <b>HTTP HEAD</b> operation.	<input type="radio"/>	<input checked="" type="radio"/>
Partners from the https://contoso.com domain can access the configured storage service by using the <b>HTTP GET</b> operation.	<input type="radio"/>	<input checked="" type="radio"/>

Section: [none]

Explanation

Explanation/Reference:

### QUESTION 3

You develop a web application that will use the Azure Table service. The web application will store entities in the form of XML data within a single table.

The web application must support high traffic throughput.

You need to avoid exceeding the throttle limit for the table.

Which two actions should you take? Each correct answer presents part of the solution.

- A. Add additional partition keys to the table.
- B. Batch transactions for entities that are in the same partition group in the table.
- C. Compress the entities before storing them in the table.
- D. Store the entities in JSON format.

**Correct Answer:** BD

Section: [none]

Explanation

Explanation/Reference:

Explanation:

#### **QUESTION 4**

You are managing an application. The application uses data that is stored in an Azure SQL database.

You must be able to reset the application to the state that existed on any day in the previous 35 days.

You need to choose a backup solution.

What should you do?

- A. Run SQL replication on the SQL database once a day.
- B. Use Microsoft Azure SQL Database Point in Time Restore
- C. Use the SQL Server Data-Tier Application Framework to build a data-tier application (DAC) file once a day.
- D. Use the bcp utility to export data to an Azure page blob once a day.

**Correct Answer:** B

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

#### **QUESTION 5**

HOTSPOT

Tailspin Toys uses a website to manage its inventory. The website is hosted on Azure. You are writing a Windows Store app that uses data from the blob storage.

You need to retrieve an image from the following URI:

<https://tailspintoys.blob.core.windows.net/Trains/Caboose2.jpg>.

How should you complete the relevant code? To answer, select the appropriate code segments in the answer area.

**Hot Area:**

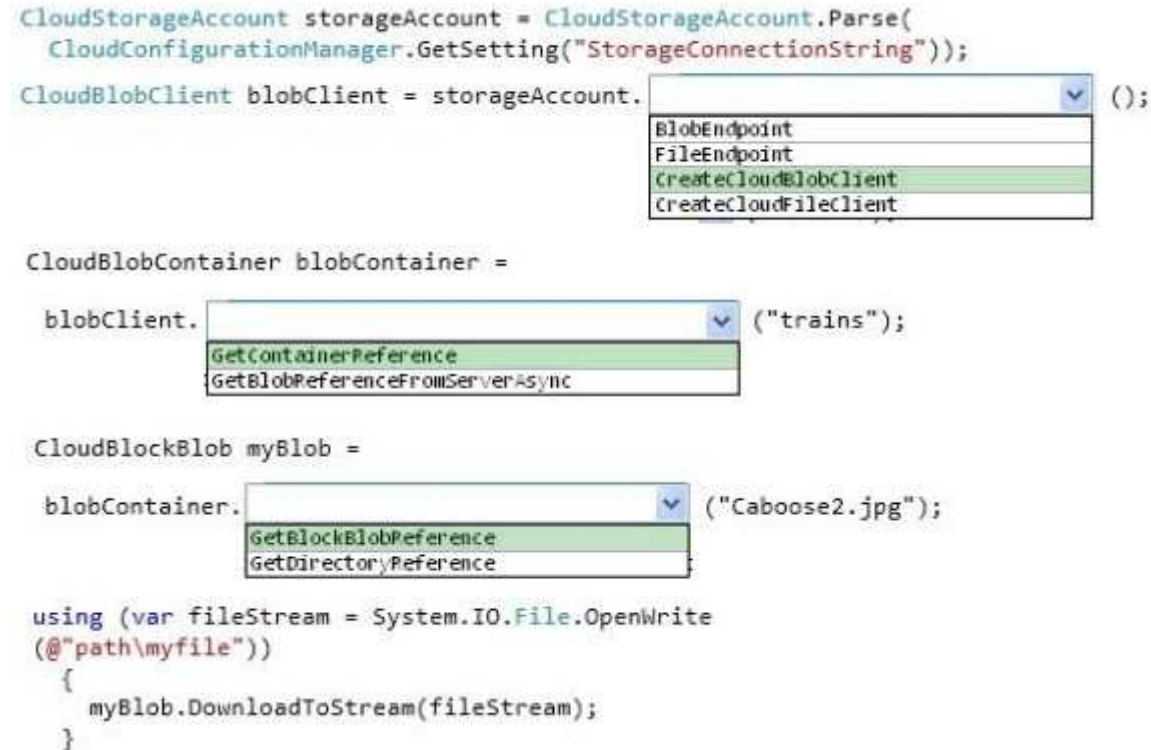
### Answer Area

```
CloudStorageAccount storageAccount = CloudStorageAccount.Parse(  
    CloudConfigurationManager.GetSetting("StorageConnectionString"));  
CloudBlobClient blobClient = storageAccount.  
    CloudBlobContainer blobContainer =  
    blobClient.  
    CloudBlockBlob myBlob =  
    blobContainer.  
using (var fileStream = System.IO.File.OpenWrite  
    (@"path\myfile"))  
    {  
        myBlob.DownloadToStream(fileStream);  
    }
```

Correct Answer:

### Answer Area

```
CloudStorageAccount storageAccount = CloudStorageAccount.Parse(  
    CloudConfigurationManager.GetSetting("StorageConnectionString"));  
CloudBlobClient blobClient = storageAccount.  
    CloudBlobContainer blobContainer =  
    blobClient.  
CloudBlockBlob myBlob =  
    blobContainer.  
using (var fileStream = System.IO.File.OpenWrite  
    (@path\myfile"))  
    {  
        myBlob.DownloadToStream(fileStream);  
    }
```



Section: [none]

Explanation

Explanation/Reference:

### QUESTION 6

You develop a web application that uses table storage in Azure.

You create a storage account named Contoso that stores a table named CityPopulationData.

The web application stores entities in this table.

You need to query the table data by using OData.

Which URL should you use?

- A. <http://contoso.table.core.windows.net/citypopulationdata>
- B. <http://contoso.table.core.windows.net/odata/citypopulationdata>
- C. <http://azurestorage.table.core.windows.net/contoso>
- D. <http://microsoft.table.core.windows.net/contoso>
- E. <http://azure.table.core.windows.net/contoso/citypopulationdata>

**Correct Answer:** A

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

#### **QUESTION 7**

##### **HOTSPOT**

Your company works with trusted partners. These partners upload files into a storage account that you control.

Partners must be able to create, read, and write files. Partners must NOT be allowed to see files from other partners. You generate a shared access signature (SAS) for each partner.

You create the following Windows PowerShell script to create a new container for each partner. Line numbers are included for reference only.



```

01 $containerName = "partner123files"
02 $key = (Get-AzureStorageKey -StorageAccountName $storageAccountName).Primary
03 $context = New-AzureStorageContext -StorageAccountName $storageAccountName `
    -StorageAccountKey $key
05 New-AzureStorageContainer -Name $containerName -Context $context
06 $filepath = "welcome.txt"
07 $blobname = "welcome.txt"
08 Set-AzureStorageBlobContent -Container $containerName -File "$filepath" `
    -Blob $blobname -Context $context -Properties @{"ContentType"="text/plain"}
09 $oneYearFromNow = (Get-Date).AddYears(1)
10 $sasToken = New-AzureStorageContainerSASToken -Name $containerName `
    -Permission 'rwdl' -ExpiryTime $oneYearFromNow -Context $context
11 $sasBlobUri = New-AzureStorageBlobSASToken -Container $containerName `
    -Permission 'r' -ExpiryTime $oneYearFromNow -Context $context `
    -FullUri -Blob $blobname

```

For each of the following statements, select Yes if the statement is true. Otherwise, select No.

**Hot Area:**

**Answer Area**

	Yes	No
Running the command at line 10 a second time invalidates the previously generated SAS token.	<input type="radio"/>	<input type="radio"/>
Web browsers can open the welcome.txt file directly by using the full URI and the SAS token for the file.	<input type="radio"/>	<input type="radio"/>
If the primary storage key is regenerated, the SAS token is still valid until its expiration date is reached.	<input type="radio"/>	<input type="radio"/>

**Correct Answer:**

### Answer Area

	Yes	No
Running the command at line 10 a second time invalidates the previously generated SAS token.	<input type="radio"/>	<input checked="" type="radio"/>
Web browsers can open the welcome.txt file directly by using the full URI and the SAS token for the file.	<input checked="" type="radio"/>	<input type="radio"/>
If the primary storage key is regenerated, the SAS token is still valid until its expiration date is reached.	<input type="radio"/>	<input checked="" type="radio"/>

Section: [none]

Explanation

Explanation/Reference:

### QUESTION 8

A company maintains an Azure storage account. The storage account uses blobs and tables.

Customers access the storage account by using shared access signatures (SASs).

You need to monitor the usage of the storage services. You need to do the following:

- Understand which storage areas perform operations that incur an Azure fee.
- Understand which requests are denied because of insufficient permissions.
- Validate that the performance of the storage account meets the service level agreement (SLA) for the Azure Storage service.

Which three data analysis tasks should you perform? Each correct answer presents part of the solution.



<https://www.gratisexam.com/>

- A. Use data from the logs of the storage services to find individual storage access attempts that do not comply with the SLA.
- B. Use data from the logs of the storage services to calculate aggregate server latency across individual requests. Determine whether the results of this calculation indicate that the Azure Storage service is in compliance with the SLA.
- C. Analyze the logs of the storage services to determine which storage services were inaccessible because of permissions issues.
- D. Review the Azure documentation to determine which storage operations are billable. Then find records of those operations in the logs of the storage services.
- E. Analyze the logs of the storage services to find records of operations that are marked as billable.
- F. Correlate the data logged from the storage service with the permissions to store data in the individual blobs and containers. Determine which storage services were inaccessible because of permissions issues.

**Correct Answer:** BCD

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

### QUESTION 9

#### HOTSPOT

You are creating a set of load-balanced virtual machines (VMs) that are hosted on Azure.

You run the following Windows PowerShell script. Line numbers are included for reference only.

```
01 Add-AzureInternalLoadBalancer -ServiceName "Contoso-Chicago" -InternalLoadBalancerName "Data-LB"  
   -SubnetName "DataFarm1" -StaticVNetIPAddress 192.168.100.10  
02 Get-AzureVM -ServiceName "Contoso-Chicago" -Name "DATA1" | Add-AzureEndpoint -Name "DataFarm"  
   -Protocol "TCP" -LocalPort 1433 -PublicPort 1337 -DefaultProbe -InternalLoadBalancerName "Data-LB" | Update-AzureVM  
03 Get-AzureService -ServiceName "Contoso-Chicago" | Get-AzureInternalLoadBalancer
```

For each of the following statements, select Yes if the statement is true. Otherwise, select No.

**Hot Area:**

### Answer Area

	Yes	No
The internal IP address of the VM named <b>DATA1</b> is 192.168.100.10.	<input type="radio"/>	<input type="radio"/>
The endpoint named <b>DataFarm</b> can be accessed by using external port 1337.	<input type="radio"/>	<input type="radio"/>
The internal load balancer for the Contoso-Chicago service is named <b>Data-LB</b> .	<input type="radio"/>	<input type="radio"/>

Correct Answer:

### Answer Area

	Yes	No
The internal IP address of the VM named <b>DATA1</b> is 192.168.100.10.	<input type="radio"/>	<input checked="" type="radio"/>
The endpoint named <b>DataFarm</b> can be accessed by using external port 1337.	<input checked="" type="radio"/>	<input type="radio"/>
The internal load balancer for the Contoso-Chicago service is named <b>Data-LB</b> .	<input type="radio"/>	<input checked="" type="radio"/>

Section: [none]

Explanation

Explanation/Reference:

### QUESTION 10

You are creating virtual machines (VMs) that are hosted on Azure.

You must be able to change the Remote Desktop access settings for the VMs. You must also be able to change the password for the built-in administrator account on all VMs. You identify the VMAccess VM extensions that have the required capabilities.

You need to enable the VMAccess VM extensions.

Which approach should you use?

- A. Download and install the Microsoft Installer file to enable the VM Agent on each VM.
- B. Use the Azure management portal to restart each VM.
- C. When you configure the new VMs, use the Azure management portal to install the VM Agent.
- D. For each VM, use Windows PowerShell cmdlets to enable the VM Agent and the VMAccess VM extensions.

**Correct Answer:** D

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

#### **QUESTION 11**

HOTSPOT

You use the Windows PowerShell Desired State Configuration (DSC) feature to configure your company's servers. Line numbers are included for reference only.

```

01 $ConfigurationData = @{
02     AllNodes = @(
03         @{NodeName = 'Server1';Role='Web'},
04         @{NodeName = 'Server2';Role='FileShare'}
05         @{NodeName = 'Server3';Role=@('FileShare','Web')}
06     )
07 }
08 configuration RoleConfiguration
09 {
10     param ($Roles)
11     switch ($Roles)
12     {
13         'FileShare'
14         {
15             WindowsFeature FileSharing
16             {
17                 Name = 'FS-FileServer'
18             }
19         }
20         'Web'
21         {
22             WindowsFeature Web
23             {
24                 Name = 'Web-Server'
25                 Ensure = 'Absent'
26             }
27         }
28     }
29 }
30 configuration MyFirstServerConfig
31 {
32     node $allnodes.NodeName
33     {
34         WindowsFeature snmp
35         {
36             Name = 'SNMP-Service'
37         }
38         RoleConfiguration MyServerRoles
39         {
40             Roles = $Node.Role
41             DependsOn = '[WindowsFeature]snmp'
42         }
43     }
44 }

```

For each of the following statements, select Yes if the statement is true. Otherwise, select No.

**Hot Area:**

**Answer Area**

	Yes	No
The script configures SNMP service on all servers.	<input type="radio"/>	<input type="radio"/>
The script configures the Web Server (IIS) role on Server3.	<input type="radio"/>	<input type="radio"/>
Invoking the script within Windows PowerShell applies the desired state to all servers.	<input type="radio"/>	<input type="radio"/>

**Correct Answer:**

**Answer Area**

	Yes	No
The script configures SNMP service on all servers.	<input type="radio"/>	<input checked="" type="radio"/>
The script configures the Web Server (IIS) role on Server3.	<input type="radio"/>	<input checked="" type="radio"/>
Invoking the script within Windows PowerShell applies the desired state to all servers.	<input checked="" type="radio"/>	<input type="radio"/>

**Section:** [none]

**Explanation**

**Explanation/Reference:**

**QUESTION 12**

**HOTSPOT**

You have an existing server that runs Windows Server. You plan to create a base image of this server. You will use this base image to prepare several virtual

servers for future use. After the base image is prepared, you will capture it by using the Azure management portal.

You must use the System Preparation Tool (Sysprep) to prepare the server so that the base image can be captured.

You need to prepare the server so that the base image can be captured.

What should you do? To answer, configure the appropriate options in the dialog box in the answer area.

**Hot Area:**

**System Preparation Tool dialog box**

System Preparation Tool 3.14

System Preparation Tool (Sysprep) prepares the machine for hardware independence and cleanup.

**System Cleanup Action**

Enter System Out-of-Box Experience (OOBE)  
Enter System Audit Mode

☐ Generalize

**Shutdown Options**

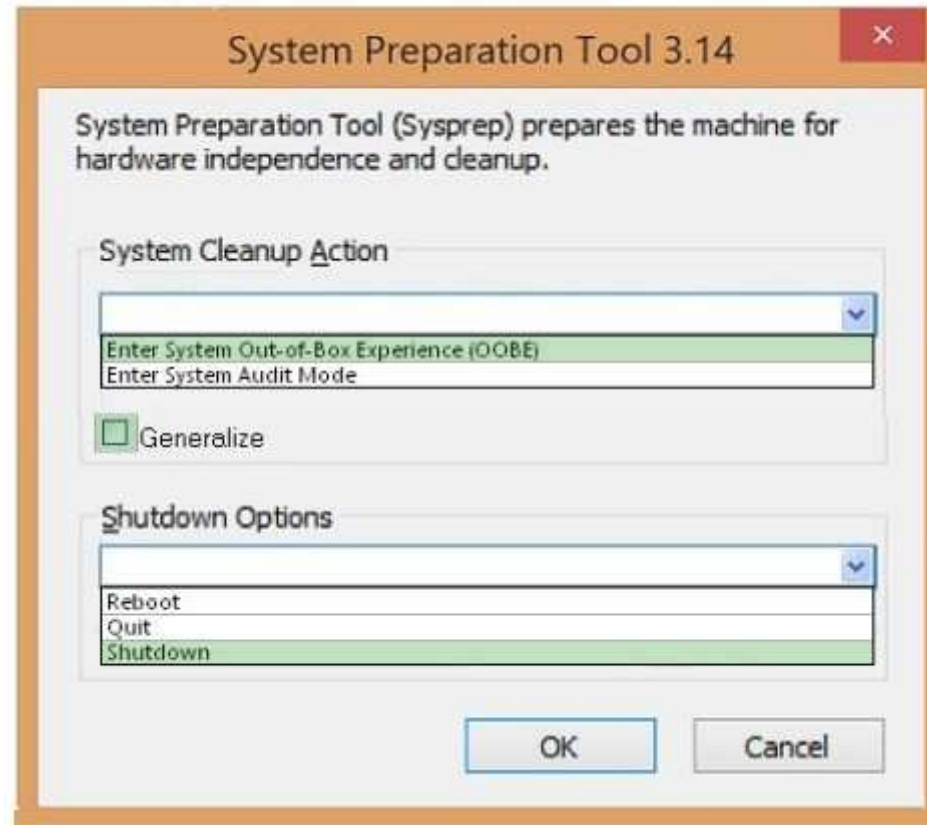
Reboot  
Quit  
Shutdown

OK Cancel

**Correct Answer:**



**System Preparation Tool dialog box**



**Section:** [none]

**Explanation**

**Explanation/Reference:**

**QUESTION 13**

**HOTSPOT**

A company creates an Azure worker role to manage products.

The number of customers who inquire about how many products are in inventory rapidly increases.

You need to ensure that the worker role can scale to accommodate the increased workload.

How should you complete the relevant code? To answer, select the appropriate option or options in the answer area.

Hot Area:

Answer Area

▼	scaler	=
Scaler		
Autoscaler		
Metronome		
Configuration		

```
EnterpriseLibraryContainer.Current.GetInstance<▼>();
```

scaler.	▼	()
Start()		
Create()		
ActivityID(true)		
AllEventsCategory(true)		

▼	>();
Scaler	
Autoscaler	
Metronome	
Configuration	

Correct Answer:

Answer Area

▼	scaler	=
Scaler		
Autoscaler		
Metronome		
Configuration		

```
EnterpriseLibraryContainer.Current.GetInstance<▼>();
```

scaler.	▼	()
Start()		
Create()		
ActivityID(true)		
AllEventsCategory(true)		

▼	>();
Scaler	
Autoscaler	
Metronome	
Configuration	

Section: [none]  
Explanation

**Explanation/Reference:**

**QUESTION 14**

You connect to an existing service over the network by using HTTP. The service listens on HTTP port 80. You plan to create a test environment for this existing service by using an Azure virtual machine (VM) that runs Windows Server.

The service must be accessible from the public Internet over HTTP port 8080.

You need to configure the test environment.

Which two actions should you take? Each correct answer presents part of the solution.

- A. Configure an endpoint to route traffic from port 8080 to port 80.
- B. Configure an endpoint to route traffic from port 80 to port 8080.
- C. Ensure that the public IP address is configured as a static IP address.
- D. Configure the Windows Server firewall to allow incoming and outgoing traffic on port 8080.
- E. Configure the Windows Server firewall to allow incoming and outgoing traffic on port 80.

**Correct Answer:** AE

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

**QUESTION 15**

**HOTSPOT**

You are developing a messaging solution for a financial services company named Adatum. The solution must integrate an application named Enrollment and an application named Activation.

The Enrollment application is used to enroll new customers. The Activation application is used to activate accounts for new customers.

You need to ensure that each message that the Enrollment application sends is stored in a queue for ten minutes before the Activation application uses the message.

How should you complete the relevant code? To answer, select the appropriate option or options in the answer area.

**Hot Area:**

### Answer Area

```
var address =  
ServiceBusEnvironment.CreateServiceUri(" ",  
    " ", string.Empty);  
var ns = new (address, new NamespaceManagerSettings()  
{  
    OperationTimeout =  
});  
ns.CreateQueue("ActivationQueue");
```

Correct Answer:

### Answer Area

```
var address =  
ServiceBusEnvironment.CreateServiceUri("sb",  
"  
sb  
msmq  
adatum.servicebus.windows.net/activate  
", string.Empty);  
var ns = new NamespaceManager (address, new NamespaceManagerSettings()  
{  
    OperationTimeout = new TimeSpan(0,10,0)  
});  
ns.CreateQueue("ActivationQueue");
```

Section: [none]

Explanation

Explanation/Reference:

### QUESTION 16

An application sends Azure push notifications to a client application that runs on Windows Phone, iOS, and Android devices.

Users cannot use the application on some devices. The authentication mechanisms that the application uses are the source of the problem.

You need to monitor the number of notifications that failed because of authentication errors.

Which three metrics should you monitor? Each correct answer presents part of the solution.

- A. Microsoft Push Notification Service (MPNS) authentication errors
- B. External notification system errors
- C. Apple Push Notification Service (APNS) authentication errors
- D. Channel errors
- E. Windows Push Notification Services (WNS) authentication errors
- F. Google Cloud Messaging (GCM) authentication errors

**Correct Answer:** ACF

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

#### **QUESTION 17**

You deploy an application as a cloud service in Azure.

The application consists of five instances of a web role.

You need to move the web role instances to a different subnet.

Which file should you update?

- A. Service definition
- B. Diagnostics configuration
- C. Service configuration
- D. Network configuration

**Correct Answer:** C

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

### QUESTION 18

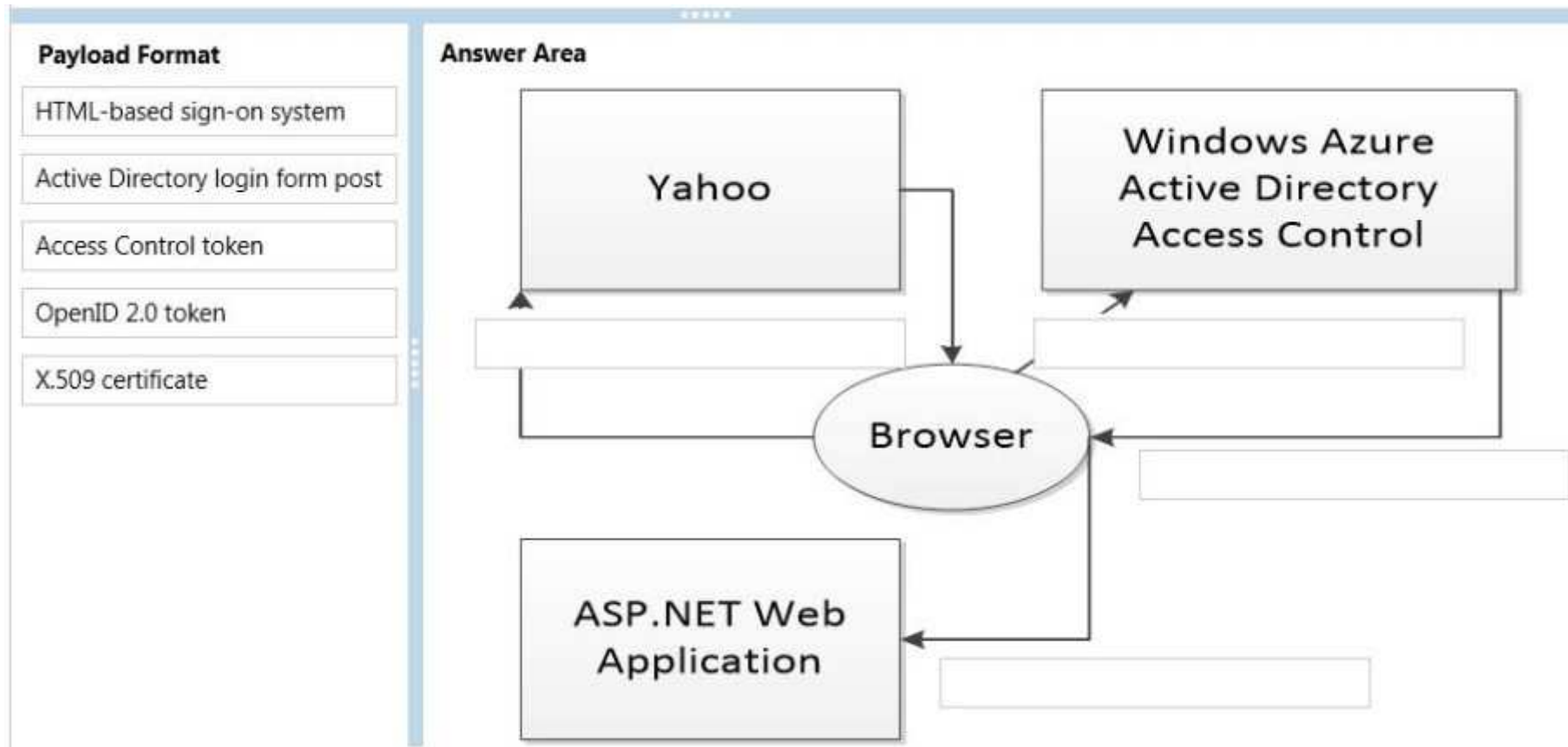
#### DRAG DROP

You are converting an existing ASP.NET web application to use the Azure Active Directory (AD) Access Control service for authentication. The application will authenticate users by using their Yahoo account credentials.

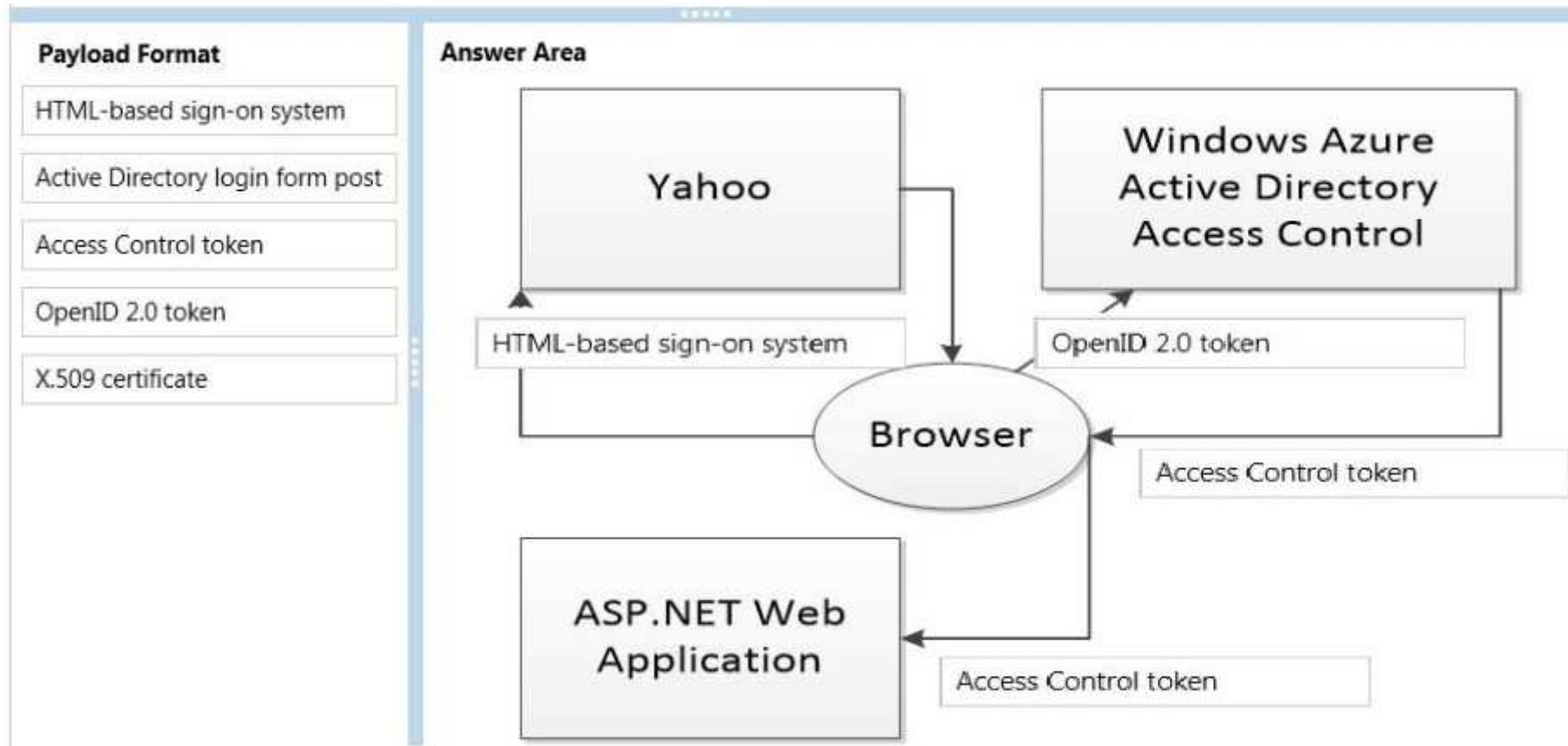
You need to determine the correct payload for each stage of the authentication process.

What should you do? To answer, drag the appropriate payload format to the correct location on the dialog box. Each payload format may be used once, more than once, or not at all. You may need to drag the split bar between panes or scroll to view content.

Select and Place:



Correct Answer:



Section: [none]

Explanation

Explanation/Reference:

#### QUESTION 19

**Note:** This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.

After you answer a question in this sections, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

You plan to deploy an application as a cloud service. The application uses a virtual network to extend your on-premises network into Azure.



You need to configure a site-to-site VPN for cross-premises network connections.

Which two objects should you configure? Each correct answer presents part of the solution.



<https://www.gratisexam.com/>

- A. Dynamic routing gateway
- B. External-facing IPv6 address
- C. VPN gateway
- D. External-facing IPv4 address

**Correct Answer:** AD

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

A Site-to-Site VPN gateway connection is used to connect your on-premises network to an Azure virtual network over an IPsec/IKE (IKEv1 or IKEv2) VPN tunnel.

Verify that you have an externally facing public IPv4 address for your VPN device. This IP address cannot be located behind a NAT.

References: <https://docs.microsoft.com/en-us/azure/vpn-gateway/vpn-gateway-howto-site-to-site-resource-manager-portal#VPNDevice>

## QUESTION 20

### HOTSPOT

You have a WebJob object that runs as part of an Azure website. The WebJob object uses features from the Azure SDK for .NET. You use a well-formed but invalid storage key to create the storage account that you pass into the UploadDataToAzureStorage method.

The WebJob object contains the following code segment. Line numbers are included for reference only.

```

01 void UploadDataToAzureStorage(CloudStorageAccount storageAccount,
    string storageContainerName, string blobpath, string localpath)
02 {
03     var blobClient = storageAccount.CreateCloudBlobClient();
04     var container = blobClient.GetContainerReference(storageContainerName);
05     CloudBlockBlob blockBlob = container.GetBlockBlobReference(blobpath);
06     blockBlob.UploadFromFile(localpath, FileMode.Open);
07 }

```

For each of the following statements, select Yes if the statement is true. Otherwise, select No.

Hot Area:

### Answer Area

	Yes	No
If the storage container does not already exist when the code runs, a file can still be uploaded successfully.	<input type="radio"/>	<input type="radio"/>
If a transient fault occurs when the code segment on line 06 runs, the Azure SDK will attempt to upload the file again.	<input type="radio"/>	<input type="radio"/>
The code segment at line 06 will fail when the code runs.	<input type="radio"/>	<input type="radio"/>

Correct Answer:

### Answer Area

	Yes	No
If the storage container does not already exist when the code runs, a file can still be uploaded successfully.	<input type="radio"/>	<input checked="" type="radio"/>
If a transient fault occurs when the code segment on line 06 runs, the Azure SDK will attempt to upload the file again.	<input checked="" type="radio"/>	<input type="radio"/>
The code segment at line 06 will fail when the code runs.	<input checked="" type="radio"/>	<input type="radio"/>

**Section: [none]**

**Explanation**

**Explanation/Reference:**

**QUESTION 21**

**DRAG DROP**

You create a web application. You publish the source code of the web application to a GitHub repository by using Microsoft Visual Studio. You create an Azure Web App by using the Azure management portal.

You must continuously deploy the web application from the GitHub repository website to the Azure Web App.

You need to deploy the source code of the web application.

Which four actions should you perform in sequence? To answer, move the appropriate actions from the list of actions to the answer area and arrange them in the correct order.

**Select and Place:**

Actions	Answer Area
Select the repository and the branch from which to deploy the Azure Web App.	
Select <b>GitHub</b> as the source control method.	
Configure the Azure Web App to use the <b>Always On</b> option.	
In the Azure management portal, configure web endpoint monitoring.	
In the Azure management portal, choose the option to set up deployment from source control.	
Sign in to GitHub by using your deployment credentials.	

**Correct Answer:**

Actions	Answer Area
	In the Azure management portal, choose the option to set up deployment from source control.
	Select <b>GitHub</b> as the source control method.
Configure the Azure Web App to use the <b>Always On</b> option.	Sign in to GitHub by using your deployment credentials.
In the Azure management portal, configure web endpoint monitoring.	Select the repository and the branch from which to deploy the Azure Web App.

Section: [none]

Explanation

Explanation/Reference:

#### QUESTION 22

A company creates an API and makes it accessible on an Azure Web App. External partners use the API occasionally. The Web App uses the Standard web hosting plan.

Partners report that the first API call in a sequence of API calls occasionally takes longer than expected to run. Subsequent API calls consistently perform as expected.

You need to ensure that all API calls perform consistently.

What should you do?

- A. Configure the Web App to use the Basic web hosting plan.
- B. Enable Always On support.
- C. Configure the Web App to automatically scale.
- D. Add a trigger to the web.config file for the Web App that causes the website to recycle periodically.

**Correct Answer:** B

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

### **QUESTION 23**

#### **HOTSPOT**

You have a cloud service that runs an external process that is named MyStartupTask.cmd. The cloud service runs this external process when the web role starts. The external process writes information to the Windows registry. You set the value of an environment variable named MyID to the deployment ID for the current web role instance.

The external process must complete writing the information to the Windows registry before the web role starts to accept web traffic.

You need to configure the cloud service.

How should you complete the relevant markup? To answer, select the appropriate option or options in the answer area.

**Hot Area:**

### Answer Area

```
<Startup>
  <Task commandLine="MyStartupTask.cmd"
    executionContext="elevated" taskType="simple"
    executionContext="limited" taskType="foreground"
    executionContext="elevated" taskType="foreground"
    executionContext="elevated" taskType="background"

  <Environment>
    <Variable name="MyId">
      <RoleInstanceValue xpath="/RoleEnvironment/Deployment/@id"/>
      <RoleInstanceValue xpath="/DeploymentId"/>
      <RoleEnvironment.DeploymentId />
      <Value>@DeploymentId</Value>
    </Variable>
  </Environment>
</Task>
</Startup>
```

Correct Answer:

## Answer Area

```
<Startup>
  <Task commandLine="MyStartupTask.cmd"
    executionContext="elevated" taskType="simple"
    executionContext="limited" taskType="foreground"
    executionContext="elevated" taskType="foreground"
    executionContext="elevated" taskType="background"

  <Environment>
    <Variable name="MyId">
      <RoleInstanceValue xpath="/RoleEnvironment/Deployment/@id"/>
      <RoleInstanceValue xpath="/DeploymentId"/>
      <RoleEnvironment.DeploymentId />Value>
      <Value>@DeploymentId</Value>
    </Variable>
  </Environment>
</Task>
</Startup>
```

Section: [none]

Explanation

Explanation/Reference:

### QUESTION 24

DRAG DROP

You deploy an application as a cloud service to Azure. The application contains a web role to convert temperatures between Celsius and Fahrenheit.

The application does not correctly convert temperatures. You must use Microsoft Visual Studio to determine why the application does not correctly convert temperatures.

You need to debug the source code in Azure.

Which three actions should you perform in sequence? To answer, move the appropriate actions from the list of actions to the answer area and arrange them in the correct order.



Select and Place:

Actions	Answer Area
Attach the debugger to the role instance of the cloud service.	
Publish the application.	
In the Microsoft Azure Publish Settings dialog, set the build configuration to <b>Release</b> and enable the remote debugger for all roles.	
In the Windows Azure Publish Settings dialog, set the build configuration to <b>Debug</b> .	
In the Microsoft Azure Publish Settings dialog, enable Remote Desktop for cloud configuration and enable the remote debugger for all roles.	

Correct Answer:

Actions	Answer Area
	Publish the application.
	In the Microsoft Azure Publish Settings dialog, set the build configuration to <b>Release</b> and enable the remote debugger for all roles.
	Attach the debugger to the role instance of the cloud service.
In the Windows Azure Publish Settings dialog, set the build configuration to <b>Debug</b> .	
In the Microsoft Azure Publish Settings dialog, enable Remote Desktop for cloud configuration and enable the remote debugger for all roles.	

Section: [none]

Explanation

Explanation/Reference:

#### QUESTION 25

DRAG DROP

Your team uses a proprietary source control product. You use FTP to manually deploy an Azure website.

You must move your source code from the proprietary source control product to a secure on-premises Git versioning system. Instead of deploying the website by using FTP, the website must automatically deploy to Azure each time developers check-in source files.

You need to implement the new deployment strategy.

Which three actions should you perform in sequence? To answer, move the appropriate actions from the list of actions to the answer area and arrange them in the correct order.

**Select and Place:**

Actions	Answer Area
In the Azure management portal, configure websites to support deployment from the local Git repository.	
In the Azure management portal, configure websites to support deployment from external repository sources.	
In the Azure management portal, configure websites to support deployment from Microsoft Visual Studio Online.	
Commit the website to Azure.	
Create the website and add it to the local Git repository.	

**Correct Answer:**

Actions	Answer Area
	Create the website and add it to the local Git repository.
In the Azure management portal, configure websites to support deployment from external repository sources.	In the Azure management portal, configure websites to support deployment from the local Git repository.
In the Azure management portal, configure websites to support deployment from Microsoft Visual Studio Online.	Commit the website to Azure.

**Section:** [none]

**Explanation**

**Explanation/Reference:**

### QUESTION 26

HOTSPOT

You are developing an Azure cloud service for a company. The cloud service monitors a queue for incoming messages and then processes invoices based on the contents of these messages.

Some messages are formed incorrectly and cause exceptions. There is no time limit for how long the service takes to process an individual message.

All messages must be processed at least once by using the ProcessMessage method. Messages must not be processed more than twice by using the ProcessMessage method. Messages that fail normal processing must be processed by using the ProcessPoisonMessage method.

You need to configure message processing.

How should you complete the relevant code? To answer, select the appropriate option or options in the answer area.

Hot Area:

### Answer Area

```
private bool ProcessNextQueueMessage(CloudQueue cloudQueue)
{
    var msg = cloudQueue.GetMessage();

|                                                |
|------------------------------------------------|
| if (msg == null) return false;                 |
| if (msg.DequeueCount > 0) return false;        |
| if (msg.PopReceipt == null) return false;      |
| if (msg.ExpirationTime.HasValue) return false; |



|                             |
|-----------------------------|
| if (msg == null)            |
| if (msg.DequeueCount > 0)   |
| if (msg.DequeueCount > 2)   |
| if (msg.PopReceipt == null) |



    ProcessPoisonMessage(msg);
    else
        ProcessMessage(msg);

|                                 |
|---------------------------------|
| cloudQueue.Delete();            |
| cloudQueue.DeleteMessage(msg);  |
| cloudQueue.EndAddMessage(null); |
| cloudQueue.DeleteMessage(null); |



    return true;
}
```

Correct Answer:

### Answer Area

```
private bool ProcessNextQueueMessage(CloudQueue cloudQueue)
{
    var msg = cloudQueue.GetMessage();

    if (msg == null) return false;
    if (msg.DequeueCount > 0) return false;
    if (msg.PopReceipt == null) return false;
    if (msg.ExpirationTime.HasValue) return false;

    ProcessPoisonMessage(msg);
    else
        ProcessMessage(msg);

    cloudQueue.Delete();
    cloudQueue.DeleteMessage(msg);
    cloudQueue.EndAddMessage(null);
    cloudQueue.DeleteMessage(null);

    return true;
}
```

Section: [none]

Explanation

Explanation/Reference:

### QUESTION 27

HOTSPOT

You deploy a new version of a cloud-service application to a staging slot. The application consists of one web role. You prepare to swap the new version of the application into the production slot. Your Azure account has access to multiple Azure subscriptions. You load the Azure PowerShell cmdlets into the Windows PowerShell command shell. The command shell is NOT configured for certificate-based authentication.

You must use the Windows PowerShell command window to configure the application.

You need to create five instances of the web role.

How should you configure the relevant Windows PowerShell script? To answer, select the appropriate option or options in the answer area.

**Hot Area:**

**Answer Area**

**\$subscription = 'mysubscription'**

**\$service = 'myservice'**

**\$rolename = 'myrole'**

	▼
Add-AzureAccount	
Get-AzureAccount -Name \$subscription	
Get-AzureAccount	

	▼
Select-AzureSubscription -SubscriptionName \$subscription	
Set-AzureSubscription -SubscriptionName \$subscription	
Set-AzureSubscription -SubscriptionId \$subscription	

	▼
Set-AzureRole -ServiceName \$service -Slot Staging -RoleName \$rolename -Count 5	
Set-AzureRole -ServiceName \$service -RoleName \$rolename -Count 5	
Set-AzureRole -ServiceName \$service -Slot Production -RoleName \$rolename -Count 5	
Add-AzureWebRole -Name \$service -Instances 5	

**Correct Answer:**

## Answer Area

```
$subscription = 'mysubscription'  
$service = 'myservice'  
$rolename = 'myrole'
```

```
Add-AzureAccount  
Get-AzureAccount -Name $subscription  
Get-AzureAccount
```

```
Select-AzureSubscription -SubscriptionName $subscription  
Set-AzureSubscription -SubscriptionName $subscription  
Set-AzureSubscription -SubscriptionId $subscription
```

```
Set-AzureRole -ServiceName $service -Slot Staging -RoleName $rolename -Count 5  
Set-AzureRole -ServiceName $service -RoleName $rolename -Count 5  
Set-AzureRole -ServiceName $service -Slot Production -RoleName $rolename -Count 5  
Add-AzureWebRole -Name $service -Instances 5
```

Section: [none]

Explanation

Explanation/Reference:

### QUESTION 28

Which of the following applications would be a good candidate to move to a cloud-based platform?

- A. Mission critical financial data
- B. Customer Relationship Management (CRM)
- C. High-performance computing
- D. Database that requires a low latency for indexing

**Correct Answer: B**



**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

The best place to start is with new applications that are customer-, partner- and employee-facing.

Cloud CRM (or CRM cloud) means any customer relationship management (CRM) technology where the CRM software, CRM tools and the organization's customer data resides in the cloud and is delivered to end-users via the Internet.

Cloud CRM typically offers access to the application via Web-based tools (or Web browser) logins where the CRM system administrator has previously defined access levels across the organization. Employees can log in to the CRM system, simultaneously, from any Internet-enabled computer or device. Often, cloud CRM provide users with mobile apps to make it easier to use the CRM on smartphones and tablets.

References:

<https://azure.microsoft.com/en-us/blog/a-key-it-decision-which-apps-to-move-to-the-cloud/>

[http://www.webopedia.com/TERM/C/crm\\_cloud.html](http://www.webopedia.com/TERM/C/crm_cloud.html)

**QUESTION 29**

Companies that are looking to move from capital expenses to operating expenses benefit from cloud services.

- A. True
- B. False

**Correct Answer: A**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

"Capex vs. Opex" refers to the fact that stocking your own data center requires capital expenditure, while using an external cloud service that offers pay-as-you-go service falls into ongoing operating expenditures: thus the contrast of "Capex vs. Opex."

References: <http://www.cio.com/article/2430099/virtualization/capex-vs--opex--most-people-miss-the-point-about-cloud-economics.html>

**QUESTION 30**

A private cloud is defined as:

- A. A deployment model that uses an external cloud to provide host application services that are Internet accessible.
- B. A deployment model that partners with other industry related companies to provide infrastructure services.
- C. A deployment model that uses virtualization technologies to provide infrastructure on demand within its network.



<https://www.gratisexam.com/>

D. A deployment model that uses an external cloud provider to provide host infrastructure services that are Internet accessible.

**Correct Answer:** C

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Private cloud is a type of cloud computing that delivers similar advantages to public cloud, including scalability and self-service, but through a proprietary architecture. Unlike public clouds, which deliver services to multiple organizations, a private cloud is dedicated to a single organization.

Private cloud expenses include virtualization, cloud software and cloud management tools.

References: <http://searchcloudcomputing.techtarget.com/definition/private-cloud>

### QUESTION 31

Which of the following is the logical progression in internal private cloud adoption?

- A. Virtualize, PaaS, IaaS and SaaS
- B. SaaS, PaaS, IaaS and Virtualize
- C. Virtualize, IaaS, PaaS and SaaS
- D. IaaS, PaaS, Virtualize and SaaS

**Correct Answer:** C

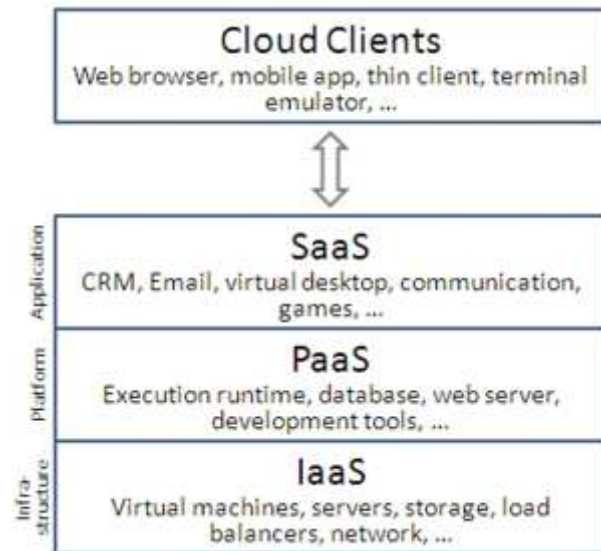
**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Cloud computing service models arranged as layers in a stack.



References: [https://en.wikipedia.org/wiki/Cloud\\_computing#Service\\_models](https://en.wikipedia.org/wiki/Cloud_computing#Service_models)

### QUESTION 32

A cloud computing vendor is focusing on delivering applications to customers. The goal is to simplify the deployment of database functionality while removing the need for customers to manage the operation system and application patching. Which of the following types of solution is the vendor offering?

- A. IT as a Service
- B. Infrastructure as a Service
- C. Anything as a Service
- D. Platform as a Service
- E. Software as a Service

**Correct Answer:** D

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

PaaS includes infrastructure — servers, storage, and networking — but also middleware, development tools, business intelligence (BI) services, database management systems, and more.

Note:



References: <https://azure.microsoft.com/en-us/overview/what-is-paas/>

### QUESTION 33

Which of the following describes what is meant by the ITIL Service Strategy component?

- A. Defining processes required to manage the solution.
- B. Designing the solution to the ITIL specifications.
- C. Ensuring changes are designed to meet customer expectations.
- D. Understanding the intended customer and what services are required.

**Correct Answer:** D

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

The objective of ITIL Service Strategy is to decide on a strategy to serve customers. Starting from an assessment of customer needs and the market place, the Service Strategy lifecycle stage determines which services the IT organization is to offer and what capabilities need to be developed. Its ultimate goal is to make the IT organization think and act in a strategic manner.

References: [http://wiki.en.it-processmaps.com/index.php/ITIL\\_Service\\_Strategy](http://wiki.en.it-processmaps.com/index.php/ITIL_Service_Strategy)

#### **QUESTION 34**

Using https instead of http for accessing a cloud service is considered more secure.

- A. True
- B. False

**Correct Answer: A**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

HTTPS (also called HTTP over TLS, HTTP over SSL, and HTTP Secure) is a protocol for secure communication over a computer network which is widely used on the Internet.

References: <https://en.wikipedia.org/wiki/HTTPS>

#### **QUESTION 35**

A company is designing a new web-based software application that must be highly available and resistant. Which of the following is the BEST environment for the application?

- A. The primary instance of the application will be locally hosted with a weekly copy of the instance sent to a cloud service provider.
- B. The primary instance of the application will be locally hosted with a nightly file-level backup being performed to an off-site location.
- C. The primary instance of the application will be running a cloud service provider's hosted environment with a continuous backup to the company's local infrastructure.
- D. The primary instance of the application will be locally hosted with a nightly copy of the instance sent to a client service provider.

**Correct Answer: C**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

**QUESTION 36**

Which of the following are the MOST important benefits of a cloud computing solution for an application development provider? (Select two.)

- A. Reduced training time for new developers
- B. Reduced storage requirements.
- C. Reduced complexity for users.
- D. Reduced bandwidth usage.
- E. Reduced cost.
- F. Reduced development timeframe.

**Correct Answer:** EF

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

The biggest promise of Azure-based applications is the ability to write them to scale as needed in real-time. Customers will therefore only use the amount of resources they need, rather than budgeting a set amount of resources that can overtax or underutilize their current setup.

References: <http://searchcloudcomputing.techtarget.com/tutorial/An-introduction-to-developing-for-Microsoft-Azure>

**QUESTION 37**

Which of the following virtualization characteristics allows the use of different types of physical types or physical servers?

- A. Security
- B. Hardware independence
- C. Scalability
- D. Variable costs

**Correct Answer:** B

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Virtualization is a conversion process that translates unique IT hardware into emulated and standardized software-based copies. Through hardware independence, virtual servers can easily be moved to another virtualization host, automatically resolving multiple hardware-software incompatibility issues. As a result, cloning and manipulating virtual IT resources is much easier than duplicating physical hardware.

References: [http://whatisccloud.com/virtualization\\_technology/hardware\\_independence](http://whatisccloud.com/virtualization_technology/hardware_independence)

### QUESTION 38

Which of the following cloud computing services requires the MOST involvement from a company's in-house staff?

- A. IaaS
- B. MaaS
- C. PaaS
- D. SaaS

**Correct Answer:** A

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Infrastructure as a service (IaaS) is an instant computing infrastructure, provisioned and managed over the Internet. Quickly scale up and down with demand, and pay only for what you use.

IaaS helps you avoid the expense and complexity of buying and managing your own physical servers and other datacenter infrastructure. Each resource is offered as a separate service component, and you only need to rent a particular one for as long as you need it. The cloud computing service provider manages the infrastructure, while you purchase, install, configure, and manage your own software—operating systems, middleware, and applications.

References: <https://azure.microsoft.com/en-us/overview/what-is-iaas/>

### QUESTION 39

As part of a cloud provider's services, customers can provision a new virtual machine as needed without human interaction with the provider. The scenario is BEST described by which of the following cloud characteristics?

- A. On-demand self-service
- B. Measured service
- C. Broad network access
- D. Rapid elasticity

**Correct Answer:** A

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

On-demand self service refers to the service provided by cloud computing vendors that enables the provision of cloud resources on demand whenever they are required. In on-demand self service, the user accesses cloud services through an online control panel.

On-demand self service resource sourcing is a prime feature of most cloud offerings where the user can scale the required infrastructure up to a substantial level without disrupting the host operations.

References: <https://www.techopedia.com/definition/27915/on-demand-self-service>

#### **QUESTION 40**

A business has recently implemented a hybrid cloud federated solution, which will allow it to rapidly and dynamically allocate resources during high demand, and quickly implement its Disaster Recovery Plan (DRP) and Continuity of Operations (COOP). Given this implementation, the IT director is mostly likely concerned about:

- A. maintaining strategic flexibility
- B. reducing OPEX allocations
- C. hiring additional IT staff
- D. eliminating security risks

**Correct Answer: D**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

Initially, the differences between Disaster Recover and Business Continuity should be understood. In comparison they are very similar in that they are (or should be) detailed plans to prepare an organization for events in which a situation presents itself which can cause internal systems failures, or a disruption of business systems in which they are no longer able to function to meet the requirements to perform day to day tasks. These situations almost always result in loss of revenue, and in some cases, loss of client base. Where these plans differ is in the main concept topic for which they prepare. Business Continuity Plans generally focus on the continuation of business services in the event of any type of interruptions whether its IT based or other. Disaster Recover Plans often refer to a company's strategy if something happens to crucial business data, and how to restore / recover that data (generally in the shortest amount of time possible).

References: <https://stumpj.wordpress.com/2010/10/18/coop-and-drp-what-is-the-difference/>

#### **QUESTION 41**

Which of the following is the primary difference between private and public cloud?

- A. Tenancy of the cloud
- B. Management of the cloud
- C. Service model of the cloud



D. Locations on the cloud

**Correct Answer:** D

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

A private cloud hosting solution, also known as an internal or enterprise cloud, resides on company's intranet or hosted data center where all of your data is protected behind a firewall.

The main differentiator between public and private clouds is that you aren't responsible for any of the management of a public cloud hosting solution. Your data is stored in the provider's data center and the provider is responsible for the management and maintenance of the data center.

References: <https://www.expedient.com/blog/private-vs-public-cloud-whats-difference/>

#### QUESTION 42

After migrating the company's entire datacenter infrastructure to a private IaaS solution, while at the same time maintaining the current network and server logical configuration, the IT director eliminated 50% of the IT engineering staff. The remaining staff has now shifted focus from a daily server maintenance and upkeep role, to more of a service provisioning, performance, and reporting role. Which of the following was MOST impacted by this migration?



<https://www.gratisexam.com/>

- A. Service design
- B. Service strategy
- C. Service operation
- D. Service transitions

**Correct Answer:** C

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

#### QUESTION 43

<https://www.gratisexam.com/>

A small company with an in-house IT staff is considering implementing a new technology that their current IT staff is unfamiliar with. The company would like to implement the new technology as soon as possible but does not have the budget to hire new IT staff. Which of the following should the company consider?

- A. Cloud computing
- B. New hardware
- C. Outsourcing
- D. Virtualization

**Correct Answer:** C

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

#### **QUESTION 44**

A company wants to implement an internal virtualized infrastructure to provide its employees with on demand storage which will be accessible through a web interface over the public Internet. This is an example of which of the following?

- A. Public cloud
- B. Community cloud
- C. Hybrid cloud
- D. Private cloud

**Correct Answer:** C

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

A hybrid cloud is an integrated cloud service utilizing both private and public clouds to perform distinct functions within the same organization.

#### **QUESTION 45**

Which of the following enables hardware independence?

- A. In-sourcing
- B. Outsourcing
- C. Virtualization
- D. Abstraction

**Correct Answer:** C

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Virtualization is a conversion process that translates unique IT hardware into emulated and standardized software-based copies. Through hardware independence, virtual servers can easily be moved to another virtualization host, automatically resolving multiple hardware-software incompatibility issues. As a result, cloning and manipulating virtual IT resources is much easier than duplicating physical hardware.

References: [http://whatiscloud.com/virtualization\\_technology/hardware\\_independence](http://whatiscloud.com/virtualization_technology/hardware_independence)

**QUESTION 46**

Which of the following are the common elements of platform as a service and software as a service? (Select two.)

- A. Both allow the OS to be patched by the customer.
- B. Both take advantage of incremental scalability.
- C. Both require the customer to maintain the hardware.
- D. Both provide granular access to the backend storage.
- E. Both implement hardware abstraction.

**Correct Answer:** BE

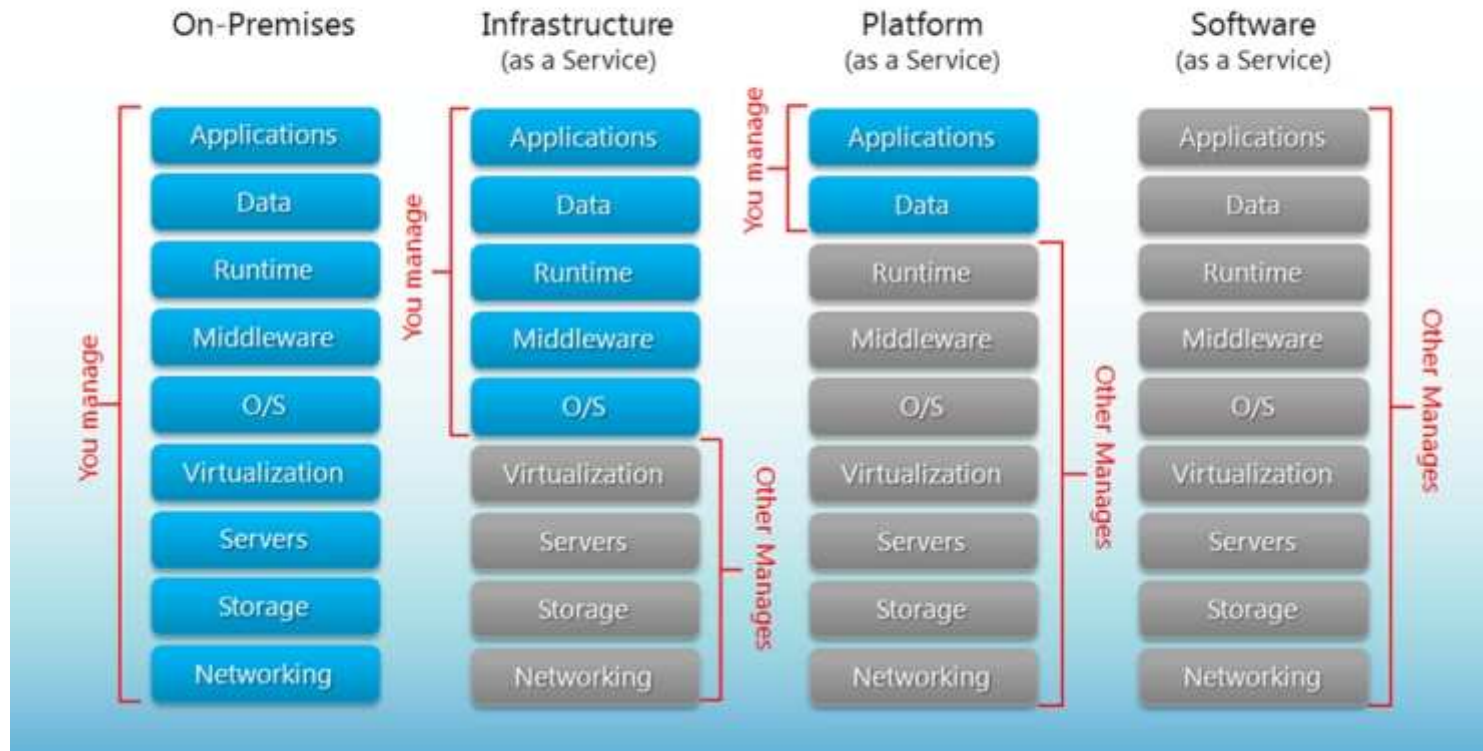
**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

# Separation of Responsibilities



## QUESTION 47

When using SaaS, the cloud computing vendor is responsible to maintain which of the following?

- A. Client infrastructure
- B. Client firewall
- C. Updates and licenses.
- D. Workstation OS version.

**Correct Answer: C**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

**QUESTION 48**

Which of the following is the MOST significant risk to business continuity when using an external cloud service provider?

- A. Unauthorized access to customer data
- B. Vendor being purchased
- C. Virtual server failure
- D. Vendor going out of business

**Correct Answer: A**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

If your application stores and retrieves very sensitive data, you might not be able to maintain it in the cloud. Similarly, compliance requirements could also limit your choices.

References: <http://cloudacademy.com/blog/cloud-migration-benefits-risks/>

**QUESTION 49**

Which of the following is an example of SaaS?

- A. Offshore help desk support
- B. Hosted database software and development tools
- C. Hosted email software
- D. Hosted network hardware

**Correct Answer: C**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

If you've used a web-based email service such as Outlook, Hotmail, or Yahoo! Mail, then you've already used a form of SaaS. With these services, you log into your

account over the Internet, often from a web browser. The email software is located on the service provider's network, and your messages are stored there as well. You can access your email and stored messages from a web browser on any computer or Internet-connected device.

References: <https://azure.microsoft.com/en-us/overview/what-is-saas/>

#### QUESTION 50

Cloud computing relies heavily on which of the following virtualization characteristics? (Select two.)

- A. User federation
- B. Hardware independence
- C. Simplistic setup
- D. Scalable resources
- E. Information sharing

**Correct Answer:** BD

**Section:** [none]

**Explanation**

#### **Explanation/Reference:**

Explanation:

B: Virtualization is a conversion process that translates unique IT hardware into emulated and standardized software-based copies. Through hardware independence, virtual servers can easily be moved to another virtualization host, automatically resolving multiple hardware-software incompatibility issues. As a result, cloning and manipulating virtual IT resources is much easier than duplicating physical hardware.

D: Infrastructure as a Service (IaaS) is a form of cloud computing that provides virtualized computing resources over the Internet. IaaS platforms offer highly scalable resources that can be adjusted on-demand.

References:

[http://whatiscloud.com/virtualization\\_technology/hardware\\_independence](http://whatiscloud.com/virtualization_technology/hardware_independence)

<http://searchcloudcomputing.techtarget.com/definition/Infrastructure-as-a-Service-IaaS>

#### QUESTION 51

Following an IT Service Management lifecycle approach, a Chief Information Officer would take which of the following paths to implement a cloud solution?

- A. Choose the SaaS provider, Design the application; Choose whether to develop the service application in-house or outsource; Operate the service application in the cloud.
- B. Decide whether to implement on the cloud; Choose a XaaS provider, Design the application; Choose where to develop the service application; Operate the service application in the cloud.
- C. Decide whether to implement the application on the cloud; Choose an IaaS provider; Choose whether to develop the service in-house; Operate the Service application in the cloud.

D. Strategize which IaaS provider to use; Design the application; Transition processes to the cloud; Operate the service application in the cloud.

**Correct Answer:** C

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

**QUESTION 52**

An organization wants to host a critical application on two redundant leased servers located on the ISP's datacenter. Which of the following is this an example of?

- A. PaaS
- B. IaaS
- C. Public cloud
- D. SaaS

**Correct Answer:** B

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Infrastructure as a service (IaaS) is an instant computing infrastructure, provisioned and managed over the Internet.

IaaS helps you avoid the expense and complexity of buying and managing your own physical servers and other datacenter infrastructure. Each resource is offered as a separate service component, and you only need to rent a particular one for as long as you need it. The cloud computing service provider manages the infrastructure, while you purchase, install, configure, and manage your own software—operating systems, middleware, and applications.



References: <https://azure.microsoft.com/en-us/overview/what-is-iaas/>

#### QUESTION 53

A cloud usage metering scheme allows for which of the following customer chargeback alternatives?

- A. Cost allocation
- B. Cost amortization
- C. Shared cost
- D. Direct cost

**Correct Answer:** D

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:



## CHARGEBACK METHODS

A range of approaches have been developed for implementing chargeback in an organization, as summarized in the figure below. The degree of complexity, degree of difficulty, and cost to implement decreases from the top of the chart [service-based pricing (SBP)], to the bottom [high-level allocation (HLA)]. HLA is the simplest method; it uses a straight division of IT costs based on a generic metric such as headcount. Slightly more effort to implement is low-level allocation (LLA), which bases consumer costs on something more related to IT activity such as the number of users or servers. Direct cost (DC) more closely resembles a time and materials charge but is often tied to headcount as well.

Figure, Methods for chargeback allocation.

METHOD	DESCRIPTION
<b>Service Based Pricing (SBP)</b>	Charges per a specific measured unit of service
<b>Negotiated Flat Rate (NFR)</b>	Charges based on a negotiated and often projected usage of a service
<b>Tiered Flat Rate (TFR)</b>	Charges based on providing access to a service whether the service is being used or not (fliers or bands pricing)
<b>Measured Resource Usage (MRU)</b>	Charges based on actual measured usage of specific IT resources (e.g., kW consumed, network bandwidth consumed, and storage consumed)
<b>Direct Cost (DC)</b>	Charges based on dedicated ownership of the resource (e.g., time and material based costing)
<b>Low-level Allocation (LLA)</b>	Charges based on simpler user metrics (e.g., user counts and server counts)
<b>High-level Allocation (HLA)</b>	Charges based on user size (e.g., number of employees and amount of revenue)

References: <https://journal.uptimeinstitute.com/it-chargeback-drives-efficiency/>

## QUESTION 54

From a risk assessment perspective, which of the following is MOST important to acquire and review before a business integrates cloud computing into its existing environment?

- A. Cloud provider DRP and COOP
- B. The time to market expectation
- C. The total cost of ownership
- D. The company original RFP

**Correct Answer:** A

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

A disaster recovery plan (DRP) - sometimes referred to as a business continuity plan (BCP) or business process contingency plan (BPCP) - describes how an organization is to deal with potential disasters.

#### **QUESTION 55**

Which of the following describes the commonality between cloud computing and outsourcing?

- A. Shift from CAPEX to OPEX
- B. Reduced compliance cost
- C. Simplified security management
- D. Reduced system architecture complexity.

**Correct Answer:** A

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

"Capex vs. Opex" refers to the fact that stocking your own data center requires capital expenditure, while using an external cloud service that offers pay-as-you-go service falls into ongoing operating expenditures: thus the contrast of "Capex vs. Opex."

References: <http://www.cio.com/article/2430099/virtualization/capex-vs--opex--most-people-miss-the-point-about-cloud-economics.html>

#### **QUESTION 56**

Virtual Desktop Interface (VDI) will present challenges for the network administrator as they move their users to the cloud. Which of the following would be considered a major challenge?



<https://www.gratisexam.com/>

- A. Developing a backup environment for the end user
- B. Troubleshooting the users' applications
- C. Supporting multiple devices (e.g. tablets, thin clients)
- D. Centralizing the applications

**Correct Answer:** C

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

References: <https://msdn.microsoft.com/en-us/library/dn903170.aspx>

#### **QUESTION 57**

A company Chief Information Officer (CIO) who wants to ensure rapid elasticity for the company's cloud solution would MOST likely choose which of the following types of cloud?

- A. Public cloud
- B. Private community cloud
- C. Private cloud
- D. Community cloud

**Correct Answer:** C

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Rapid elasticity is a cloud computing term for scalable provisioning, or the ability to provide scalable services.

Software that can scale in a private cloud faces two security related issues:

- Although the private cloud infrastructure can enable rapid elasticity in the supply of virtual resources, hosted applications and services must be designed correctly if they are to function securely when they are scaled out.
- Hosted applications and services that initiate scaling requests automatically based on monitored demand or a timetable must perform these operations without impacting their own or other services availability within the cloud.

References: <http://social.technet.microsoft.com/wiki/contents/articles/6810.private-cloud-security-challenges-rapid-elasticity.aspx>

#### **QUESTION 58**

Consumption statistics for individual cloud service offerings is used by which of the following ITIL processes?

- A. Supplier management
- B. Continuous service improvement
- C. Service level management
- D. Information security management

**Correct Answer: C**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

ITIL defines Service Management as “a set of specialized organisational capabilities for providing value to customers in the form of services”.

The managed service provider will intermediate between the cloud service provider and consumer, aligning the two and ensuring minimal service disruptions.

ITIL Service Level Management aims to negotiate Service Level Agreements with the customers and to design services in accordance with the agreed service level targets. Service Level Management is also responsible for ensuring that all Operational Level Agreements and Underpinning Contracts are appropriate, and to monitor and report on service levels.

References:

<https://blog.kloud.com.au/2016/04/06/consumption-based-service-management/>

[http://wiki.en.it-processmaps.com/index.php/Service\\_Level\\_Management](http://wiki.en.it-processmaps.com/index.php/Service_Level_Management)

#### **QUESTION 59**

An application development company is considering implementing a cloud solution to help improve time to market with new software upgrades. The existing application has been in use by customers for several years and contains a large amount of code. Which of the following types of clouds would be BEST for this company to implement?

- A. IaaS
- B. XaaS

- C. PaaS
- D. SaaS

**Correct Answer:** C

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Platform as a service (PaaS) is a complete development and deployment environment in the cloud, with resources that enable you to deliver everything from simple cloud-based apps to sophisticated, cloud-enabled enterprise applications. You purchase the resources you need from a cloud service provider on a pay-as-you-go basis and access them over a secure Internet connection.

Like IaaS, PaaS includes infrastructure — servers, storage, and networking — but also middleware, development tools, business intelligence (BI) services, database management systems, and more. PaaS is designed to support the complete web application lifecycle: building, testing, deploying, managing, and updating.

PaaS allows you to avoid the expense and complexity of buying and managing software licenses, the underlying application infrastructure and middleware or the development tools and other resources. You manage the applications and services you develop, and the cloud service provider typically manages everything else.

References: <https://azure.microsoft.com/en-us/overview/what-is-paas/>

#### **QUESTION 60**

A critical internal IT server provisioning process is under review and the IT manager is considering moving the process to the cloud. The IT staff has selected the cloud provider and must now migrate the process. Which of the following **MUST** the IT staff do to ensure the transaction meets the IT manager's requirements?

- A. Pilot the process using cloud resources and perform a comprehensive test.
- B. Survey the business users and implement the solution that received the most positive feedback.
- C. Ask the server administrator to sign off and approve the implementation plan.
- D. Shift the current process to the cloud since the SLA will guarantee 99.999% availability.

**Correct Answer:** A

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

#### **QUESTION 61**

One of the strategic reasons to source component technology purchases from multiple providers is to:

- A. Avoid vendor lock-in.
- B. Influence governmental organizations.
- C. Keep vendor prices down.
- D. Encourage vendor control.

**Correct Answer:** A

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

When it comes to building applications for the cloud, John Gossman, an employee of Microsoft, thinks agility and portability are essential. "You don't want to get locked in too much to a particular vendor, strategy, technology, whatever," he says. Likewise, he added, you aren't likely to last long if your plan is to pick a single public cloud vendor and host everything there.

References: [http://www.theregister.co.uk/2014/12/06/microsoft\\_linux\\_and\\_the\\_cloud/](http://www.theregister.co.uk/2014/12/06/microsoft_linux_and_the_cloud/)

#### **QUESTION 62**

A graphic design company regularly runs out of storage space on its file servers due to the large size of its customer artwork files. The company is considering migrating to cloud computing to solve this problem. Which of the following characteristics of cloud computing is the MOST beneficial reason the company should implement a cloud solution?

- A. Scalability
- B. Security
- C. Variable costs
- D. Hardware independence

**Correct Answer:** A

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

#### **QUESTION 63**

Which of the following will allow an organization to integrate internal identity management services with a cloud provider in order to provide single sign-on across the internal and cloud-hosted environments?

- A. Virtualization
- B. Federation
- C. Role-based authentication
- D. Outsourcing

**Correct Answer:** B

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Azure AD supports three different ways to sign in to applications:

- Federated Single Sign-On enables applications to redirect to Azure AD for user authentication instead of prompting for its own password. This is supported for applications that support protocols such as SAML 2.0, WS-Federation, or OpenID Connect, and is the richest mode of single sign-on.
- Password-based Single Sign-On
- Existing Single Sign-On

References: <https://azure.microsoft.com/en-us/documentation/articles/active-directory-appssoaccess-what-is/>

#### **QUESTION 64**

Which of the following is a potential advantage of using Storage as a Service?

- A. Data is accessible when the Internet is not functioning
- B. In-house IT staff controls all data
- C. Increase in encryption technologies
- D. Decrease in IT management of the platform

**Correct Answer:** D

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

One advantages of SaaS is that it makes it easy to “mobilize” your workforce because users can access SaaS apps and data from any Internet-connected computer or mobile device. You don’t need to worry about developing apps to run on different types of computers and devices because the service provider has already done so. In addition, you don’t need to bring special expertise onboard to manage the security issues inherent in mobile computing. A carefully chosen service provider will ensure the security of your data, regardless of the type of device consuming it.

References: <https://azure.microsoft.com/en-us/overview/what-is-saas/>

**QUESTION 65**

Which of the following describes the difference between SaaS and IaaS?

- A. SaaS defines a standard while IaaS implements the standard.
- B. SaaS enables the software developer while IaaS provides the specifications.
- C. SaaS provides applications while IaaS provides equipment.
- D. SaaS implements security while IaaS provides information.

**Correct Answer: C**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

Software as a service (SaaS) allows users to connect to and use cloud-based apps over the Internet.





References: <https://azure.microsoft.com/en-us/overview/what-is-saas/>

#### QUESTION 66

Locating datacenters close to target markets is the result of which of the following strategic initiatives?

- A. Geoproximity
- B. Geodiversity
- C. Geography
- D. Localization

**Correct Answer:** A

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Azure allows specification of geographical regions or affinity groups. Geographical regions are related to the data centers, like North Central US, South Central US,

Anywhere US, East Asia, North Europe, and so on. The list of options will grow as more data centers are added.

Instead of selecting a region, it is possible to specify an affinity group. Affinity groups are hints to Azure that essentially state that everything within the group should be as close in proximity as Azure will allow. That usually means keeping items within the same data center, which besides having the benefit of geo-location, can sometimes offer performance improvements for communication.

References: <http://greglevenhagen.com/azure-geographical-location-restriction/>

#### **QUESTION 67**

A company regularly doubles the number of employees over the summer by hiring on temporary staff. The company currently pays the same price every month for its email software, equal to the maximum number of employees on staff who have email addresses. The company would like to only pay for the number of active email addresses and is considering migrating to a cloud solution. Which of the following characteristics of cloud computing is the MOST beneficial reason the company should implement a cloud solution?

- A. Scalability
- B. Hardware independence
- C. Variable costs
- D. Security

**Correct Answer:** C

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

#### **QUESTION 68**

Which of the following is a benefit of public cloud computing?

- A. Enhances fixed expenditures for hardware and software
- B. Contributes to the quality of user input data
- C. Reduces OPEX costs for application and databases
- D. Adds flexibility and agility to enterprise architecture

**Correct Answer:** D

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Public cloud computing offers greater flexibility, agility, and scalability.

**QUESTION 69**

Which of the following is the cloud characteristic that speeds up development, deployment and overall time of market?



<https://www.gratisexam.com/>

- A. Rapid elasticity
- B. Cloud bursting
- C. Universal access
- D. Network pooling

**Correct Answer:** A

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Rapid elasticity is a cloud computing term for scalable provisioning, or the ability to provide scalable services. Experts point to this kind of scalable model as one of five fundamental aspects of cloud computing.

Rapid elasticity allows users to automatically request additional space in the cloud or other types of services.

References: <https://www.techopedia.com/definition/29526/rapid-elasticity>

**QUESTION 70**

A company would like to move an application to the cloud which resides on a single physical server in their datacenter. The server has two drives, one of which hosts operating system, and the other hosts the application data. The operating system has been showing errors recently and the application data was corrupted last Friday at 4:00PM. Data is backed up every day at 1:00PM. Which of the following would be the BEST option for migrating this application to the cloud?

- A. Setup a server in the cloud, install an operating system, install the application and copy the data to the cloud server from last Friday's backup.
- B. Setup a server in the cloud, install an operating system, install and configure the application and copy the data to the cloud server from last Thursday's backup.
- C. Clone or P2V the server with both drivers to the cloud platform.
- D. Clone or P2V the server with the application to the cloud platform and copy the operating system to the cloud server.

**Correct Answer:** A

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

Use the latest backup of the application data.

**QUESTION 71**

An existing capability is being migrated into the cloud. Capacity management issues have been noticed in the past and an exercise is being performed to calculate current and future volumes. In which of the following lifecycle phases is this likely to be performed?

- A. Operation
- B. Design
- C. Transition
- D. Strategy

**Correct Answer: C**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

**QUESTION 72**

An entrepreneur has decided to open an e-commerce site to complement their retail store. After researching their options, they decide that a PaaS solution will be sufficient. To reduce upfront cost, the entrepreneur intends to build the site themselves. Which of the following skill-tests will be needed?

- A. Firewall Administration
- B. Web-Server Administration
- C. Security standard development
- D. Application development

**Correct Answer: D**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

Platform as a service (PaaS) is a complete development and deployment environment in the cloud, with resources that enable you to deliver everything from simple

cloud-based apps to sophisticated, cloud-enabled enterprise applications.

PaaS allows you to avoid the expense and complexity of buying and managing software licenses, the underlying application infrastructure and middleware or the development tools and other resources. You manage the applications and services you develop, and the cloud service provider typically manages everything else.

References: <https://azure.microsoft.com/en-us/overview/what-is-paas/>

#### **QUESTION 73**

An organization is planning to host a number of its critical applications in the cloud. Which of the following is the Best way to gain a broad assurance of the cloud provider's security posture?

- A. A review that includes interviewing key security stakeholders and identifying the key controls that they operate.
- B. A review that includes security policies, evidence of the controls, physical site assessments and vulnerability scanning.
- C. A review that includes the right to audit on a yearly basis and review of the security clauses in the contract.
- D. A review that includes security applications, external audits, intrusion detection and firewall policy reviews.

**Correct Answer:** B

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

#### **QUESTION 74**

One major impact that cloud computing has had on the application development process is the need for greater:

- A. security
- B. speed
- C. isolation
- D. standardization

**Correct Answer:** A

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

#### **QUESTION 75**

Why is it important to know the physical location for a governmental cloud based storage solution?

- A. Data stored in other countries could be accessed by the local government.
- B. Data stored in other countries could slow down application response.
- C. Data stored in other countries could impact access latency.
- D. Data stored in other countries could reduce revenue for the originating country.

**Correct Answer:** A

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

With Azure Government all data, applications, and hardware reside in the continental United States.

References: <https://azure.microsoft.com/en-us/overview/clouds/government/>

#### **QUESTION 76**

Which of the following should be measured with a direct cost chargeback method?

- A. Power and cooling consumed
- B. CPU cycles used
- C. Technical staff
- D. Square footage cost of the facility

**Correct Answer:** A

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

#### **CHARGEBACK METHODS**

A range of approaches have been developed for implementing chargeback in an organization, as summarized in the figure below. The degree of complexity, degree of difficulty, and cost to implement decreases from the top of the chart [service-based pricing (SBP)], to the bottom [high-level allocation (HLA)]. HLA is the simplest method; it uses a straight division of IT costs based on a generic metric such as headcount. Slightly more effort to implement is low-level allocation (LLA), which bases consumer costs on something more related to IT activity such as the number of users or servers. Direct cost (DC) more closely resembles a time and materials charge but is often tied to headcount as well.

References: <https://journal.uptimeinstitute.com/it-chargeback-drives-efficiency/>

**QUESTION 77**

An organization is moving web server clusters to a public IaaS cloud while keeping database servers in the company owned datacenter. The organization will continue utilizing the internal service desk to manage the application. Which of the following ITIL processes will plan the move?

- A. Release Management
- B. Incident Management
- C. Problem Management
- D. Change Management

**Correct Answer:** D

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Change Management is an IT service management discipline. The objective of change management in this context is to ensure that standardized methods and procedures are used for efficient and prompt handling of all changes to control IT infrastructure, in order to minimize the number and impact of any related incidents upon service.

References: [https://en.wikipedia.org/wiki/Change\\_management\\_\(ITSM\)](https://en.wikipedia.org/wiki/Change_management_(ITSM))

**QUESTION 78**

HOTSPOT

**Case Study**

This is a case study. Case studies are not timed separately. You can use as much exam time as you would like to complete each case. However, there may be additional case studies and sections on this exam. You must manage your time to ensure that you are able to complete all questions included on this exam in the time provided.

To answer the questions included in a case study, you will need to reference information that is provided in the case study. Case studies might contain exhibits and other resources that provide more information about the scenario that is described in the case study. Each question is independent of the other question on this case study.

At the end of this case study, a review screen will appear. This screen allows you to review your answers and to make changes before you move to the next sections of the exam. After you begin a new section, you cannot return to this section.

To start the case study

To display the first question on this case study, click the Next button. Use the buttons in the left pane to explore the content of the case study before you answer the questions. Clicking these buttons displays information such as business requirements, existing environment, and problem statements. If the case study has an All Information tab, note that the information displayed is identical to the information displayed on the subsequent tabs. When you are ready to answer a question, click the Question button to return to the question.

**Background**

You are a developer for Fabrikam, a company that specializes in payment processing. Fabrikam is developing a solution to process payments for various events, such as music concerts. You develop an ASP.NET MVC website that is hosted in Azure to support an upcoming music concert. The music concert is expected to generate a large volume of ticket sales in a short amount of time.

The website uploads information to an Azure storage queue. A worker role in Azure retrieves information from the queue and generates the concert tickets in a PDF file format after the financial transaction is approved.

You observe a delay between the time the website adds a message to a queue and the time it becomes available to read from the queue. After examining the queue, you determine that no queue messages have a DequeueCount value greater than zero. The website does not throw any errors.

## **Business Requirements**

### **Payments**

The music concert website must be able to submit event payment information for processing. The website must remain responsive while submitting payment information. Customers must be able to add notes about their orders to a free-form control on the website. These notes must be submitted with the payment when the customer submits an order.

Customers often enter notes that exceed 7 KB in size.

### **Technical Requirements**

#### **Payment Submission and Processing**

Event payment information must be sent from the website to a Windows Communication Foundation (WCF) service worker role. The worker role must submit the information to the payment processor in JSON format.

#### **Payment Processing**

You have the following payment processing requirements:

- If the number of messages in a queue goes above or below a specified threshold, worker role instances must be created or deleted as needed. This process must be completed by using the least amount of effort. It must be easy to reconfigure role instance thresholds.
- Payments must be retrieved from the queue in the maximum batch sizes that are allowed by the queue and pulled from the queue for 5 minutes.
- The payment queue must not be re-created when processing payments.
- During single Payment processing, the number of tickets available for an event must be updated. The update operation must be retried for 30 seconds or 5 retry attempts, whichever occurs first. Each retry should pause for at least two seconds and for one second longer than the previous attempt. If the update fails, the payment should be placed in the poison queue.

### **Storage**

You have the following storage requirements:

- Payment information must be stored by using Azure Queue storage. Connection to the Azure storage account has been established in a configured setting



named `StorageConnectionString`, which is configured for the web and worker roles.

- A payment processing queue and a poison payment queue must be used when processing payments.
- Azure Queue message content must be XML-safe and UTF-8 encoded.
- An Azure storage account must be established for diagnostic information in a configured setting named `DiagnosticsStorageConnectionString`, which is configured for both the web and worker roles.

## **Security and Monitoring**

### **Security**

The web role must be secured by using HTTPS.

### **Monitoring**

You must collect diagnostic data for both the web and worker roles by using the Diagnostics module. Diagnostics configuration changes must not require the code of the roles to be rebuilt. The diagnostic data is used for debugging and troubleshooting, measuring performance, monitoring resource usage, traffic analysis and capacity planning, and auditing.

Performance testing must evaluate the roles under normal and stress conditions without incurring changes for running Azure. Memory allocation, function time, and multithreading concurrency issues must be evaluated.

### **Deployment**

You purchase a custom domain name `fabrikamfunding.com` to host the website, web role, and worker roles. You must deploy an HTTPS certificate with the web role, and you must update associated configuration files accordingly.

Web role and worker role instance sizes must be specified as Medium. You must deploy one web role instance named `FabrikamFundingPaymentGenerator`, and worker role instances named `FabrikamFundingPaymentProcessor`.

### **Application Structure**

Relevant portions of the app files are shown below. Line numbers are included for reference only and include a two-character prefix that denotes the specific file to which they belong.

# CustomRetryPolicy.cs

```
CR01 public class CustomRetryPolicy : IRetryPolicy
CR02 {
CR03     int _retryCount = 0;
CR04     readonly TimeSpan _baseInterval= TimeSpan.FromSeconds(1);
CR05     readonly string _poisonPaymentQueueName;
CR06     private readonly CloudQueueClient _queueClient;
CR07     private readonly EventPayment _eventPayment;
CR08     public CustomRetryPolicy(string poisonPaymentQueueName, CloudQueueClient
queueClient, EventPayment eventPayment)
CR09     {
CR10         _poisonPaymentQueueName = poisonPaymentQueueName;
CR11         _queueClient = queueClient;
CR12         _eventPayment = eventPayment;
CR13     }
CR14     public IRetryPolicy CreateInstance()
CR15     {
CR16         return new CustomRetryPolicy(_poisonPaymentQueueName, _queueClient,
_eventPayment);
CR17     }
CR18 }
```

## Event.cs

```
EV01 public class Event : TableEntity
EV02 {
EV03     public int AvailableTickets { get; set; }
EV04 }
```

## EventPayment.cs

```
EP01 [DataContract]
EP02 public class EventPayment
EP03 {
EP04     [DataMember]
EP05     public int EventId { get; set; }
EP06     [DataMember]
EP07     public string Email { get; set; }
EP08     [DataMember]
EP09     public string Notes { get; set; }
EP10     [DataMember]
EP11     public int TicketCount { get; set; }
EP12     [DataMember]
EP13     public DateTime OrderDate { get; set; }
EP14     [DataMember]
EP15     public Guid EventPaymentId { get; set; }
EP16 }
```

## QueueManager.cs

```
QM01 public class QueueManager
QM02 {
QM03     private readonly CloudQueueClient _queueClient;
QM04     private readonly CloudTableClient _tableClient;
QM05     private const string PaymentQueueName = "paymentqueue";
QM06     private const string PoisonPaymentQueueName = "poisonpaymentqueue";
QM07     public QueueManager()
QM08     {
QM09         var storageAccount = CloudStorageAccount.Parse(
QM10             CloudConfigurationManager.GetSetting("StorageConnectionString"));
QM11         _queueClient = storageAccount.CreateCloudQueueClient();
QM12         _tableClient = storageAccount.CreateCloudTableClient();
QM13     }
QM14     public async Task SendMessageAsync(EventPayment eventPayment)
QM15     {
QM16         ...
QM17     }
QM18     public async Task ProcessMessagesAsync()
QM19     {
QM20         ...
QM21     }
QM22     public async Task ProcessPayment(EventPayment eventPayment)
QM23     {
QM24         var events = _tableClient.GetTableReference("events");
QM25         var key = eventPayment.EventId.ToString();
QM26         var operation = await events.ExecuteAsync(TableOperation.Retrieve<Event>(key, key));
QM27         var @event = operation.Result as Event;
QM28         @event.AvailableTickets = @event.AvailableTickets - eventPayment.TicketCount;
QM29         var requestOptions = new TableRequestOptions
QM30         {
QM31             RetryPolicy = new CustomRetryPolicy(
QM32                 PoisonPaymentQueueName,
QM33                 _queueClient,
QM34                 eventPayment),
QM35         };
QM36         var context = new OperationContext
QM37         {
QM38             StartTime = DateTime.Now,
QM39         };
QM40         await events.ExecuteAsync(TableOperation.Replace(@event),
requestOptions, context;
QM41     }
QM42 }
```

You need to implement the `SendMessageAsync` method on the `QueueManager` class.

How should you complete the relevant code? To answer, select the appropriate code segment from each list in the answer area.

**Hot Area:**

## Answer Area

```
public async Task SendMessageAsync(EventPayment eventPayment)
{
    var queue = _queueClient.GetQueueReference(PaymentQueueName);
    await queue.CreateIfNotExistsAsync();

    await queue.ClearAsync();
    queue.EncodeMessage = true;
    var eventPaymentMessage = new XmlSerializer(typeof(EventPayment)).Serialize(eventPayment);
    var eventPaymentMessage = JsonConvert.SerializeObject(eventPayment);

    await queue.PeekMessageAsync();
    queue.Metadata.Add("message", eventPaymentMessage);
    var message = new CloudQueueMessage(eventPaymentMessage);
    var message = CloudConfigurationManager.GetSetting("eventPaymentMessage");

    queue.AddMessage(message);
    queue.DeleteMessage(message);
    await queue.AddMessageAsync(message);
    await queue.DeleteMessageAsync(message);
}
```

Correct Answer:

## Answer Area

```
public async Task SendMessageAsync(EventPayment eventPayment)
{
    var queue = _queueClient.GetQueueReference(PaymentQueueName);
    await queue.CreateIfNotExistsAsync();

    await queue.ClearAsync();
    queue.EncodeMessage = true;
    var eventPaymentMessage = new XmlSerializer(typeof(EventPayment)).Serialize(eventPayment);
    var eventPaymentMessage = JsonConvert.SerializeObject(eventPayment);

    await queue.PeekMessageAsync();
    queue.Metadata.Add("message", eventPaymentMessage);
    var message = new CloudQueueMessage(eventPaymentMessage);
    var message = CloudConfigurationManager.GetSetting("eventPaymentMessage");

    queue.AddMessage(message);
    queue.DeleteMessage(message);
    await queue.AddMessageAsync(message);
    await queue.DeleteMessageAsync(message);
}
```

Section: [none]  
Explanation

## **Explanation/Reference:**

### **QUESTION 79**

#### **DRAG DROP**

#### **Case Study**

This is a case study. Case studies are not timed separately. You can use as much exam time as you would like to complete each case. However, there may be additional case studies and sections on this exam. You must manage your time to ensure that you are able to complete all questions included on this exam in the time provided.

To answer the questions included in a case study, you will need to reference information that is provided in the case study. Case studies might contain exhibits and other resources that provide more information about the scenario that is described in the case study. Each question is independent of the other question on this case study.

At the end of this case study, a review screen will appear. This screen allows you to review your answers and to make changes before you move to the next sections of the exam. After you begin a new section, you cannot return to this section.

To start the case study

To display the first question on this case study, click the Next button. Use the buttons in the left pane to explore the content of the case study before you answer the questions. Clicking these buttons displays information such as business requirements, existing environment, and problem statements. If the case study has an All Information tab, note that the information displayed is identical to the information displayed on the subsequent tabs. When you are ready to answer a question, click the Question button to return to the question.

#### **Background**

You are a developer for Fabrikam, a company that specializes in payment processing. Fabrikam is developing a solution to process payments for various events, such as music concerts. You develop an ASP.NET MVC website that is hosted in Azure to support an upcoming music concert. The music concert is expected to generate a large volume of ticket sales in a short amount of time.

The website uploads information to an Azure storage queue. A worker role in Azure retrieves information from the queue and generates the concert tickets in a PDF file format after the financial transaction is approved.

You observe a delay between the time the website adds a message to a queue and the time it becomes available to read from the queue. After examining the queue, you determine that no queue messages have a DequeueCount value greater than zero. The website does not throw any errors.

#### **Business Requirements**

##### **Payments**

The music concert website must be able to submit event payment information for processing. The website must remain responsive while submitting payment information. Customers must be able to add notes about their orders to a free-form control on the website. These notes must be submitted with the payment when the customer submits an order.

Customers often enter notes that exceed 7 KB in size.



## Technical Requirements

### Payment Submission and Processing

Event payment information must be sent from the website to a Windows Communication Foundation (WCF) service worker role. The worker role must submit the information to the payment processor in JSON format.

### Payment Processing

You have the following payment processing requirements:

- If the number of messages in a queue goes above or below a specified threshold, worker role instances must be created or deleted as needed. This process must be completed by using the least amount of effort. It must be easy to reconfigure role instance thresholds.
- Payments must be retrieved from the queue in the maximum batch sizes that are allowed by the queue and pulled from the queue for 5 minutes.
- The payment queue must not be re-created when processing payments.
- During single Payment processing, the number of tickets available for an event must be updated. The update operation must be retried for 30 seconds or 5 retry attempts, whichever occurs first. Each retry should pause for at least two seconds and for one second longer than the previous attempt. If the update fails, the payment should be placed in the poison queue.

### Storage

You have the following storage requirements:

- Payment information must be stored by using Azure Queue storage. Connection to the Azure storage account has been established in a configured setting named StorageConnectionString, which is configured for the web and worker roles.
- A payment processing queue and a poison payment queue must be used when processing payments.
- Azure Queue message content must be XML-safe and UTF-8 encoded.
- An Azure storage account must be established for diagnostic information in a configured setting named DiagnosticsStorageConnectionString, which is configured for both the web and worker roles.

## Security and Monitoring

### Security

The web role must be secured by using HTTPS.

### Monitoring

You must collect diagnostic data for both the web and worker roles by using the Diagnostics module. Diagnostics configuration changes must not require the code of the roles to be rebuilt. The diagnostic data is used for debugging and troubleshooting, measuring performance, monitoring resource usage, traffic analysis and capacity planning, and auditing.

Performance testing must evaluate the roles under normal and stress conditions without incurring changes for running Azure. Memory allocation, function time, and multithreading concurrency issues must be evaluated.

## Deployment

You purchase a custom domain name `fabrikamfunding.com` to host the website, web role, and worker roles. You must deploy an HTTPS certificate with the web role, and you must update associated configuration files accordingly.

Web role and worker role instance sizes must be specified as Medium. You must deploy one web role instance named `FabrikamFundingPaymentGenerator`, and worker role instances named `FabrikamFundingPaymentProcessor`.

## Application Structure

Relevant portions of the app files are shown below. Line numbers are included for reference only and include a two-character prefix that denotes the specific file to which they belong.

### CustomRetryPolicy.cs

```
CR01 public class CustomRetryPolicy : IRetryPolicy
CR02 {
CR03     int _retryCount = 0;
CR04     readonly TimeSpan _baseInterval= TimeSpan.FromSeconds(1);
CR05     readonly string _poisonPaymentQueueName;
CR06     private readonly CloudQueueClient _queueClient;
CR07     private readonly EventPayment _eventPayment;
CR08     public CustomRetryPolicy(string poisonPaymentQueueName, CloudQueueClient
queueClient, EventPayment eventPayment)
CR09     {
CR10         _poisonPaymentQueueName = poisonPaymentQueueName;
CR11         _queueClient = queueClient;
CR12         _eventPayment = eventPayment;
CR13     }
CR14     public IRetryPolicy CreateInstance()
CR15     {
CR16         return new CustomRetryPolicy(_poisonPaymentQueueName, _queueClient,
_eventPayment);
CR17     }
CR18 }
```

## Event.cs

```
EV01 public class Event : TableEntity
EV02 {
EV03     public int AvailableTickets { get; set; }
EV04 }
```

## EventPayment.cs

```
EP01 [DataContract]
EP02 public class EventPayment
EP03 {
EP04     [DataMember]
EP05     public int EventId { get; set; }
EP06     [DataMember]
EP07     public string Email { get; set; }
EP08     [DataMember]
EP09     public string Notes { get; set; }
EP10     [DataMember]
EP11     public int TicketCount { get; set; }
EP12     [DataMember]
EP13     public DateTime OrderDate { get; set; }
EP14     [DataMember]
EP15     public Guid EventPaymentId { get; set; }
EP16 }
```

## QueueManager.cs

```
QM01 public class QueueManager
QM02 {
QM03     private readonly CloudQueueClient _queueClient;
QM04     private readonly CloudTableClient _tableClient;
QM05     private const string PaymentQueueName = "paymentqueue";
QM06     private const string PoisonPaymentQueueName = "poisonpaymentqueue";
QM07     public QueueManager()
QM08     {
QM09         var storageAccount = CloudStorageAccount.Parse(
QM10             CloudConfigurationManager.GetSetting("StorageConnectionString"));
QM11         _queueClient = storageAccount.CreateCloudQueueClient();
QM12         _tableClient = storageAccount.CreateCloudTableClient();
QM13     }
QM14     public async Task SendMessageAsync(EventPayment eventPayment)
QM15     {
QM16         ...
QM17     }
QM18     public async Task ProcessMessagesAsync()
QM19     {
QM20         ...
QM21     }
QM22     public async Task ProcessPayment(EventPayment eventPayment)
QM23     {
QM24         var events = _tableClient.GetTableReference("events");
QM25         var key = eventPayment.EventId.ToString();
QM26         var operation = await events.ExecuteAsync(TableOperation.Retrieve<Event>(key, key));
QM27         var @event = operation.Result as Event;
QM28         @event.AvailableTickets = @event.AvailableTickets - eventPayment.TicketCount;
QM29         var requestOptions = new TableRequestOptions
QM30         {
QM31             RetryPolicy = new CustomRetryPolicy(
QM32                 PoisonPaymentQueueName,
QM33                 _queueClient,
QM34                 eventPayment),
QM35         };
QM36         var context = new OperationContext
QM37         {
QM38             StartTime = DateTime.Now,
QM39         };
QM40         await events.ExecuteAsync(TableOperation.Replace(@event),
requestOptions, context;
QM41     }
QM42 }
```

You need to implement the ProcessPaymentAsync method in the QueueManager class.

Develop the solution by selecting and arranging the required code blocks in the correct order.

NOTE: You will not need all of the code segments.

**Select and Place:**

## Code segments

```
while (true)
{
    var messages =
        await queue.GetMessagesAsyn-
            c(32, TimeSpan.FromMinutes(5),
                null, null);
    foreach (var message in messag-
        es.Where(message => message !=
            null))
    {
```

```
        var eventPayment = JsonConvert.
            DeserializeObject<EventPayment>
                (message.AsString);
```

```
        await queue.DeleteMessageAsync
            (message);
    }
}
```

```
public async Task ProcessPay-
    mentsAsync()
{
    var queue = _queueClient.Get-
        QueueReference (Payment-
            QueueName);
    await queue.CreateAsync();
```

```
public async Task ProcessPay-
    mentsAsync()
{
    var queue = _queueClient.Get-
        QueueReference (Payment-
            QueueName);
```

## Answer Area



**Correct Answer:**

## Code segments

```
var eventPayment = JsonConvert.  
DeserializeObject<EventPayment>  
message.AsString);
```

```
public async Task ProcessPay-  
mentsAsync()  
{  
var queue = _queueClient.Get-  
QueueReference (Payment-  
QueueName);  
await queue.CreateAsync();
```

## Answer Area

```
public async Task ProcessPay-  
mentsAsync()  
{  
var queue = _queueClient.Get-  
QueueReference (Payment-  
QueueName);  
await queue.CreateIfNotEx-  
istsAsync();
```

```
while (true)  
{  
var messages =  
await queue.GetMessagesAsyn-  
c(32, TimeSpan.FromMinutes(5),  
null, null);  
foreach (var message in messag-  
es.Where(message => message !=  
null))  
{
```

```
await ProcessPayment  
(eventPayment);
```

```
await queue.DeleteMessageAsync  
(message);  
}  
}  
}
```





**Section: [none]**

**Explanation**

**Explanation/Reference:**

## **QUESTION 80**

### **Case Study**

This is a case study. Case studies are not timed separately. You can use as much exam time as you would like to complete each case. However, there may be additional case studies and sections on this exam. You must manage your time to ensure that you are able to complete all questions included on this exam in the time provided.

To answer the questions included in a case study, you will need to reference information that is provided in the case study. Case studies might contain exhibits and other resources that provide more information about the scenario that is described in the case study. Each question is independent of the other question on this case study.

At the end of this case study, a review screen will appear. This screen allows you to review your answers and to make changes before you move to the next sections of the exam. After you begin a new section, you cannot return to this section.

To start the case study

To display the first question on this case study, click the Next button. Use the buttons in the left pane to explore the content of the case study before you answer the questions. Clicking these buttons displays information such as business requirements, existing environment, and problem statements. If the case study has an All Information tab, note that the information displayed is identical to the information displayed on the subsequent tabs. When you are ready to answer a question, click the Question button to return to the question.

### **Background**

You are a developer for Fabrikam, a company that specializes in payment processing. Fabrikam is developing a solution to process payments for various events, such as music concerts. You develop an ASP.NET MVC website that is hosted in Azure to support an upcoming music concert. The music concert is expected to generate a large volume of ticket sales in a short amount of time.

The website uploads information to an Azure storage queue. A worker role in Azure retrieves information from the queue and generates the concert tickets in a PDF file format after the financial transaction is approved.

You observe a delay between the time the website adds a message to a queue and the time it becomes available to read from the queue. After examining the queue, you determine that no queue messages have a DequeueCount value greater than zero. The website does not throw any errors.

### **Business Requirements**

#### **Payments**

The music concert website must be able to submit event payment information for processing. The website must remain responsive while submitting payment information. Customers must be able to add notes about their orders to a free-form control on the website. These notes must be submitted with the payment when the customer submits an order.

Customers often enter notes that exceed 7 KB in size.

## **Technical Requirements**

### **Payment Submission and Processing**

Event payment information must be sent from the website to a Windows Communication Foundation (WCF) service worker role. The worker role must submit the information to the payment processor in JSON format.

### **Payment Processing**

You have the following payment processing requirements:

- If the number of messages in a queue goes above or below a specified threshold, worker role instances must be created or deleted as needed. This process must be completed by using the least amount of effort. It must be easy to reconfigure role instance thresholds.
- Payments must be retrieved from the queue in the maximum batch sizes that are allowed by the queue and pulled from the queue for 5 minutes.
- The payment queue must not be re-created when processing payments.
- During single Payment processing, the number of tickets available for an event must be updated. The update operation must be retried for 30 seconds or 5 retry attempts, whichever occurs first. Each retry should pause for at least two seconds and for one second longer than the previous attempt. If the update fails, the payment should be placed in the poison queue.

### **Storage**

You have the following storage requirements:

- Payment information must be stored by using Azure Queue storage. Connection to the Azure storage account has been established in a configured setting named `StorageConnectionString`, which is configured for the web and worker roles.
- A payment processing queue and a poison payment queue must be used when processing payments.
- Azure Queue message content must be XML-safe and UTF-8 encoded.
- An Azure storage account must be established for diagnostic information in a configured setting named `DiagnosticsStorageConnectionString`, which is configured for both the web and worker roles.

## **Security and Monitoring**

### **Security**

The web role must be secured by using HTTPS.

### **Monitoring**

You must collect diagnostic data for both the web and worker roles by using the Diagnostics module. Diagnostics configuration changes must not require the code of the roles to be rebuilt. The diagnostic data is used for debugging and troubleshooting, measuring performance, monitoring resource usage, traffic analysis and capacity planning, and auditing.

Performance testing must evaluate the roles under normal and stress conditions without incurring changes for running Azure. Memory allocation, function time, and multithreading concurrency issues must be evaluated.

## Deployment

You purchase a custom domain name fabrikamfunding.com to host the website, web role, and worker roles. You must deploy an HTTPS certificate with the web role, and you must update associated configuration files accordingly.

Web role and worker role instance sizes must be specified as Medium. You must deploy one web role instance named FabrikamFundingPaymentGenerator, and worker role instances named FabrikamFundingPaymentProcessor.

## Application Structure

Relevant portions of the app files are shown below. Line numbers are included for reference only and include a two-character prefix that denotes the specific file to which they belong.

### CustomRetryPolicy.cs

```
CR01 public class CustomRetryPolicy : IRetryPolicy
CR02 {
CR03     int _retryCount = 0;
CR04     readonly TimeSpan _baseInterval= TimeSpan.FromSeconds(1);
CR05     readonly string _poisonPaymentQueueName;
CR06     private readonly CloudQueueClient _queueClient;
CR07     private readonly EventPayment _eventPayment;
CR08     public CustomRetryPolicy(string poisonPaymentQueueName, CloudQueueClient
queueClient, EventPayment eventPayment)
CR09     {
CR10         _poisonPaymentQueueName = poisonPaymentQueueName;
CR11         _queueClient = queueClient;
CR12         _eventPayment = eventPayment;
CR13     }
CR14     public IRetryPolicy CreateInstance()
CR15     {
CR16         return new CustomRetryPolicy(_poisonPaymentQueueName, _queueClient,
_eventPayment);
CR17     }
CR18 }
```

## Event.cs

```
EV01 public class Event : TableEntity
EV02 {
EV03     public int AvailableTickets { get; set; }
EV04 }
```

## EventPayment.cs

```
EP01 [DataContract]
EP02 public class EventPayment
EP03 {
EP04     [DataMember]
EP05     public int EventId { get; set; }
EP06     [DataMember]
EP07     public string Email { get; set; }
EP08     [DataMember]
EP09     public string Notes { get; set; }
EP10     [DataMember]
EP11     public int TicketCount { get; set; }
EP12     [DataMember]
EP13     public DateTime OrderDate { get; set; }
EP14     [DataMember]
EP15     public Guid EventPaymentId { get; set; }
EP16 }
```

## QueueManager.cs

```
QM01 public class QueueManager
QM02 {
QM03     private readonly CloudQueueClient _queueClient;
QM04     private readonly CloudTableClient _tableClient;
QM05     private const string PaymentQueueName = "paymentqueue";
QM06     private const string PoisonPaymentQueueName = "poisonpaymentqueue";
QM07     public QueueManager()
QM08     {
QM09         var storageAccount = CloudStorageAccount.Parse(
QM10             CloudConfigurationManager.GetSetting("StorageConnectionString"));
QM11         _queueClient = storageAccount.CreateCloudQueueClient();
QM12         _tableClient = storageAccount.CreateCloudTableClient();
QM13     }
QM14     public async Task SendMessageAsync(EventPayment eventPayment)
QM15     {
QM16         ...
QM17     }
QM18     public async Task ProcessMessagesAsync()
QM19     {
QM20         ...
QM21     }
QM22     public async Task ProcessPayment(EventPayment eventPayment)
QM23     {
QM24         var events = _tableClient.GetTableReference("events");
QM25         var key = eventPayment.EventId.ToString();
QM26         var operation = await events.ExecuteAsync(TableOperation.Retrieve<Event>(key, key));
QM27         var @event = operation.Result as Event;
QM28         @event.AvailableTickets = @event.AvailableTickets - eventPayment.TicketCount;
QM29         var requestOptions = new TableRequestOptions
QM30         {
QM31             RetryPolicy = new CustomRetryPolicy(
QM32                 PoisonPaymentQueueName,
QM33                 _queueClient,
QM34                 eventPayment),
QM35         };
QM36         var context = new OperationContext
QM37         {
QM38             StartTime = DateTime.Now,
QM39         };
QM40         await events.ExecuteAsync(TableOperation.Replace(@event),
requestOptions, context;
QM41     }
QM42 }
```

The **SendMessageAsync** method of the **QueueManager** class occasionally throws errors.

You need to correct the errors.

What should you do?

- A. Remove all attributes from the **EventPayment** class.
- B. Encode the **notes** field content by using UTF-32 encoding.
- C. Update the **notes** field to a byte array. Binary encode and decode the **notes** content when sending or receiving an **EventPayment** class.
- D. Update the **SendMessageAsync** method of the **QueueManager** class to store the notes field in BLOB storage. Update the **EventPayment** class to store the BLOB uniform resource identifier (URI). Extract the **notes** BLOB information by using the BLOB URI in the **ProcessMessagesAsync** method of the **QueueManager** class.

**Correct Answer:** D

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

#### QUESTION 81

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.

After you answer a question in this sections, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

You are administering an Azure environment for your company. You plan to deploy virtual machines (VMs) for a mobile application. You have the following requirements:

- Ensure that all VMs use the **Standard D3** size.
- Ensure that at least two of the four servers must be available at all times.
- Ensure that users of the application do not experience downtime or loss of connection.

You need to configure four VMs for application development.

Solution: Create a Virtual Machine Scale Set (VMSS) that has an instance count of 4.

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer:** B

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

### QUESTION 82

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.

After you answer a question in this sections, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

You are administrating an Azure environment for your company. You plan to deploy virtual machines (VMs) for a mobile application. You have the following requirements:

- Ensure that all VMs use the **Standard D3** size.
- Ensure that at least two of the four servers must be available at all times.
- Ensure that users of the application do not experience downtime or loss of connection.

You need to configure four VMs for application development.

Solution: You create an availability set that has two fault domains and two update domains by using the Azure portal. You create four virtual machines and assign the new availability set to each VM.

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer:** A

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

### QUESTION 83

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.

After you answer a question in this sections, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

You are administrating an Azure environment for your company. You plan to deploy virtual machines (VMs) for a mobile application. You have the following requirements:

- Ensure that all VMs use the **Standard D3** size.
- Ensure that at least two of the four servers must be available at all times.
- Ensure that users of the application do not experience downtime or loss of connection.

You need to configure four VMs for application development.

Solution: Create two resource groups by using the Azure portal. Create four VMs. Assign two VMs to the first resource group and two to the second group.

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer:** B

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

#### **QUESTION 84**

**DRAG DROP**

Your team uses a proprietary source control product. You use FTP to manually deploy an Azure Web App.

You must move your source code from the proprietary source control product to a secure on-premises Git versioning system. Instead of deploying the website by using FTP, the website must automatically deploy to Azure each time developers check-in source files.



<https://www.gratisexam.com/>

You need to implement the new deployment strategy.

Which three actions should you perform in sequence? To answer, move the appropriate actions from the list of actions to the answer area and arrange them in the correct order.



Select and Place:

### Actions

In the Azure management portal, configure Web Apps to support deployment from the local Git repository.

Commit the website to the Git repository.

In the Azure portal, configure Web App to support deployment from Microsoft Visual Studio Team Services.

In the Azure portal, configure Web App to support deployment from external repository sources.

Create a local Git repository.

### Answer Area



Correct Answer:

## Actions

In the Azure portal, configure Web App to support deployment from Microsoft Visual Studio Team Services.

In the Azure portal, configure Web App to support deployment from external repository sources.

## Answer Area

In the Azure management portal, configure Web Apps to support deployment from the local Git repository.

Create a local Git repository.

Commit the website to the Git repository.

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

References:

<https://docs.microsoft.com/en-us/aspnet/core/publishing/azure-continuous-deployment>

### QUESTION 85

You are building an ASP.NET Azure Web App that is built from source code on GitHub. Automatic deployment is used for integration testing. The web.config file has settings that are updated during development deployments by using a TransformXml MSBuild task.

The settings in the web.config must be set to specific values during integration testing.

You need to ensure that the web.config is updated when the Web App is deployed to Azure.

Which two actions should you perform? Each correct answer presents part of the solution.

- A. Add the integration settings and values to the ServiceDefinition.csdef and ServiceConfiguration.csfg files.
- B. In Azure, create a new deployment slot namedIntegration.
- C. Create an XML Document Transform (XDT) file namedweb.Integration.configthat converts the values to the integration test values.
- D. In Azure, add a tag with the keyEnvironmentand the valueIntegration.

**Correct Answer:** BC

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

#### **QUESTION 86**

DRAG DROP

You are developing an ASP.NET Software-as-a-Service (SaaS) application that stores database credentials in Azure Key Vault.

When the application encounters an error, information about the error is shown to the end user for support purposes. Information shown includes:

- stack trace
- application version and release date
- configuration values

Database credentials must not be accessible to end users.

You need to ensure that end users cannot view database credentials.

Which three actions should you perform in sequence? To answer, move the appropriate actions from the list of actions to the answer area and arrange them in the correct order.

**Select and Place:**

## Actions

Create a ClientId and Client Secret using the Azure portal

Create a new Azure Web App with the ClientId and Client Secret values in application settings and deploy

Configure Azure Key Vault to use HSM

Create an Azure Web App with the certificate in custom domains and Secure Socket Layer (SSL) and deploy

Create an X509 certificate using by using makecert

Create a new Azure Active Directory (Azure AD) application with the certificate

Create an Azure Web App with the certificate pfx in application settings and deploy

## Answer Area



Correct Answer:

## Actions

- 
- 
- 
- Create an Azure Web App with the certificate in custom domains and Secure Socket Layer (SSL) and deploy
- Create an X509 certificate using by using makecert
- Create a new Azure Active Directory (Azure AD) application with the certificate
- Create an Azure Web App with the certificate pfx in application settings and deploy

## Answer Area

- Configure Azure Key Vault to use HSM
- Create a ClientId and Client Secret using the Azure portal
- ◀ Create a new Azure Web App with the ClientId and Client Secret values in application settings and deploy ▶
- ◀ ▶

Section: [none]

Explanation

Explanation/Reference:

QUESTION 87

You have an application that stores data in Azure Cosmos DB. You have the following class:

```
public static class Repository<T> where T : class
{
    public static async Task<T> SaveItem<T>(string id, T item)
    {
        var documentUri = UriFactory.CreateDocumentUri(DatabaseId, CollectionId, id);
        await client.ReplaceDocumentAsync(documentUri, item);
        var document = await client.ReadDocumentAsync(documentUri);

        return (T) (dynamic) document;
    }

    private static readonly string DatabaseId = ConfigurationManager.AppSettings["database"];
    private static readonly string CollectionId = ConfigurationManager.AppSettings["collection"];
    private static DocumentClient client;
}
```

You need to determine which consistency level to apply to this code.

Which three consistency levels can you use? Each correct answer presents a complete solution.

**NOTE:** Each correct selection is worth one point.

- A. Bounded Staleness
- B. Session
- C. Consistent Prefix
- D. Strong
- E. Eventual

**Correct Answer:** BCE

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

**QUESTION 88**

DRAG DROP

You are using Microsoft Visual Studio to develop an App Service Web App named WebApp.

The app must collect the statistics and details on the application dependencies.

You need to set up, configure, and validate monitoring using Application Insights.

Which three actions should you perform in sequence? To answer, move the appropriate actions from the list of actions to the answer area and arrange them in the correct order.

**Select and Place:**

### Actions

Upload the project and run it to generate log data.

Add Application Insights to the solution. Then, add the **Microsoft.ApplicationInsights.TraceListener** NuGet package to the project.

Start a new instance of Azure PowerShell and run the following Azure PowerShell command:  
**Get-AzureWebSiteLog -Name WebApp -Tail**

In the Azure Portal, browse to the Application Insights resource and open **Search**.

Use the Azure Command-Line interface to run the following command:  
**azure site log tail WebApp**

### Answer Area



**Correct Answer:**

## Actions

Start a new instance of Azure PowerShell and run the following Azure PowerShell command:  
**Get-AzureWebSiteLog -Name WebApp -Tail**

Use the Azure Command-Line interface to run the following command:  
**azure site log tail WebApp**

## Answer Area

Add Application Insights to the solution. Then, add the **Microsoft.ApplicationInsights.TraceListener** NuGet package to the project.



Upload the project and run it to generate log data.



In the Azure Portal, browse to the Application Insights resource and open **Search**.



Section: [none]

Explanation

Explanation/Reference:

Explanation:

References:

<https://docs.microsoft.com/en-us/azure/application-insights/app-insights-asp-net-trace-logs>

### QUESTION 89

You are creating virtual machines (VMs) that are hosted on Azure.



You must be able to change the Remote Desktop access settings for the VMs. You must also be able to change the password for the built-in administrator account on all VMs. You identify the VMAccess VM extensions that have the required capabilities.

You need to enable the VMAccess VM extensions.

Which approach should you use?

- A. Use Azure PowerShell cmdlets to change the name of the availability set to the same name for all of the VMs.
- B. Use the Azure portal to restart each VM.
- C. In the Azure Portal, create an image from a virtual hard disk (VHD) for each VM by running Sysprep.
- D. For each VM, use Azure PowerShell cmdlets to enable the VM Agent and the VMAccess VM extensions.

**Correct Answer:** D

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

#### **QUESTION 90**

You connect to an existing service over the network by using HTTP. The service listens on HTTP port 80. You plan to create a test environment for this existing service by using an Azure virtual machine (VM) that runs Windows Server.

The service must be accessible from the public Internet over HTTP port 8080.

You need to configure the test environment.

Which two actions should you take? Each correct answer presents part of the solution.

- A. Configure a Network Security Group to route traffic from port 8080 to port 80
- B. Configure a Network Security Group to route traffic from port 80 to port 8080.
- C. Ensure that the public IP address is configured as a static IP address.
- D. Configure the Windows Server firewall to allow incoming and outgoing traffic on port 8080.
- E. Configure the Windows Server firewall to allow incoming and outgoing traffic on port 80.

**Correct Answer:** AE

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

**QUESTION 91**

DRAG DROP

You plan to run SQL Server Enterprise Edition by using an Azure virtual machine (VM).

You must configure the VM to run all SQL Server high volume workloads.

You need to optimize SQL Server performance for workloads that run on the new VM.

What should you do? To answer, drag the appropriate optimization technique to the correct configuration option. Each optimization technique may be used once, more than once, or not at all. You may need to drag the split bar between panes or scroll to view content.

**Select and Place:**

## Optimization technique

Keep the VM in the same region and disable geo-replication.

Use Premium Storage and enable read caching for data files and TempDB.

Enable locked pages and instant file initialization for data files.

Use a DS3-series or higher-level VM.

Move the VM to a different region and enable geo-replication.

Enable autogrow and autoshrink.

## Answer Area

### Configuration option

### Optimization technique

VM size

Storage

Disks

I/O

**Correct Answer:**

## Optimization technique

Move the VM to a different region and enable geo-replication.
Enable autogrow and autoshrink.

## Answer Area

### Configuration option

### Optimization technique

VM size

Use a DS3-series or higher-level VM.

Storage

Keep the VM in the same region and disable geo-replication.

Disks

Use Premium Storage and enable read caching for data files and TempDB.

I/O

Enable locked pages and instant file initialization for data files.

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

References:

<https://docs.microsoft.com/en-us/azure/virtual-machines/windows/sql/virtual-machines-windows-sql-performance>

**QUESTION 92****HOTSPOT**

You are administering an Azure environment for your company that requires multiple virtual machines (VMs) for a production application. You have the following requirements:

- Two VMs are required for application data.
- Seven VMs are required for image processing.
- VM sizes should be set to Standard D2.
- Only two image processing servers can be rebooted at a time.

You need to configure an availability set for the image processing VMs.

How many fault domain and update domains should you implement? To answer, configure the appropriate options in the dialog box in the answer area.

NOTE: Each correct selection is worth one point.

**Hot Area:****Answer Area**

Availability set	Value
fault domains	<input type="text"/>
	1
	2
	3
update domains	<input type="text"/>
	2
	3
	4

**Correct Answer:**

## Answer Area

Availability set	Value
fault domains	<div></div>
	1
	2
	3
update domains	<div></div>
	2
	3
	4

Section: [none]

Explanation

Explanation/Reference:

Explanation:

References:

<https://docs.microsoft.com/en-us/azure/virtual-machines/windows/manage-availability>

### QUESTION 93

DRAG DROP

You have six Ubuntu Linux virtual machines (VMs) that run a Hadoop cluster on Azure. All VMs were deployed by using Azure Resource Manager (ARM) templates and Azure PowerShell cmdlets. One of the VMs runs a custom web user interface that allows users to examine the processing jobs within the Hadoop cluster. You are planning a backup strategy for long-term retention and recovery that includes geo-replication.

The backup and recovery solution must be cost effective.

You need to backup all VMs.

Which five actions should you perform in sequence? To answer, move the appropriate actions from the list of actions in the answer area and arrange them in the correct order.

Select and Place:

## Actions

Select the VMs to include in the backup.

Select the appropriate backup policy.

Create a recovery services vault for each VM that has geo-redundant storage replication enabled.

Set the backup goal to **Azure and VM**.

Run and confirm that an initial backup has been completed for all VMs.

Create a backup vault for the VM backups that has geo-redundant storage replication enabled.

Create a recovery services vault for the VM backups that has locally-redundant storage replication enabled.

Create a recovery services vault for the VM backups that has geo-redundant storage replication enabled.

## Answer Area



Correct Answer:

## Actions

Create a recovery services vault for each VM that has geo-redundant storage replication enabled.

Create a backup vault for the VM backups that has geo-redundant storage replication enabled.

Create a recovery services vault for the VM backups that has locally-redundant storage replication enabled.

## Answer Area

Create a recovery services vault for the VM backups that has geo-redundant storage replication enabled.

Set the backup goal to **Azure and VM**.

Select the appropriate backup policy.

Select the VMs to include in the backup.

Run and confirm that an initial backup has been completed for all VMs.





**Section: [none]**

**Explanation**

**Explanation/Reference:**

**QUESTION 94**

HOTSPOT

You administer an Azure environment that contains multiple virtual machines (VMs).

You need to view and retrieve diagnostic logs for all VMs.

Which storage type should you use for each data source? To answer, select the appropriate options in the answer area.

**Hot Area:**

## Answer Area

Data source	Storage type
Azure logs	<div><div></div><div>Blob</div><div>Table</div><div>Queue</div></div>
IIS 7.0 logs	<div><div></div><div>Blob</div><div>Table</div><div>Queue</div></div>
Windows Event logs	<div><div></div><div>Blob</div><div>Table</div><div>Queue</div></div>

Correct Answer:

## Answer Area

Data source	Storage type
Azure logs	<div><div></div><div>Blob</div><div>Table</div><div>Queue</div></div>
IIS 7.0 logs	<div><div></div><div>Blob</div><div>Table</div><div>Queue</div></div>
Windows Event logs	<div><div></div><div>Blob</div><div>Table</div><div>Queue</div></div>

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

References:

<https://docs.microsoft.com/en-us/azure/cloud-services/cloud-services-dotnet-diagnostics-storage>

### QUESTION 95

HOTSPOT

You deploy a cloud service that reads and processes orders from a queue by using a worker role. The service includes a C# class named OrderProcessor.

Your organization is moving all Azure resources to use Azure Resource Manager (ARM) templates. You must migrate the code to Service Fabric. You establish a

new Service Fabric cluster to deploy the updated code. You migrate all settings from the ServiceConfiguration.cscfg to a new Settings.xml file that each Service Fabric instance will use.

You need to update the code for the OrderProcessor class.

How should you complete the code segment? To answer, select the appropriate options in the answer area.

**Hot Area:**

```
namespace WideWorldImportersOrderProcessor
{
    public class OrderProcessor:
    {
        private CloudQueue ordersQueue;
        private CloudBlobContainer ordersBlobContainer;
        private DbContext ordersDatabaseContext;
        protected override
        {
            ConfigurationPackage configPackage = this.Context.
            .GetConfigurationPackageObject ("Config");
            KeyedCollection<string, ConfigurationProperty> parameters =
                configPackage.Settings.Sections ["MyConfigSection"].Parameters;
            string databaseConnectionString = parameters["OrdersDatabaseConnection"]?.Value;
            ordersDatabaseContext = GetOrdersDatabaseContext (databaseConnectionString);
            ordersBlobContainer = GetOrdersBlobStorageContainerReference ();
            ordersQueue = GetOrdersQueueReference ();
            ProcessOrders ();
        }
    }
}
```

**Correct Answer:**

```

namespace WideWorldImportersOrderProcessor
{
    public class OrderProcessor:
    {
        private CloudQueue ordersQueue;
        private CloudBlobContainer ordersBlobContainer;
        private DbContext ordersDatabaseContext;
        protected override
        {
            Task RunAsync(CancellationToken cancellationToken)
            IEnumerable<ServiceInstanceListener> CreateServiceInstanceListeners()

            ConfigurationPackage configPackage = this.Context.
            .GetConfigurationPackageObject ("Config");
            NodeContext
            CodePackageActivationContext

            KeyedCollection<string, ConfigurationProperty> parameters =
            configPackage.Settings.Sections ["MyConfigSection"].Parameters;
            string databaseConnectionString = parameters["OrdersDatabaseConnection"]?.Value;
            ordersDatabaseContext = GetOrdersDatabaseContext (databaseConnectionString);
            ordersBlobContainer = GetOrdersBlobStorageContainerReference ();
            ordersQueue = GetOrdersQueueReference ();
            ProcessOrders ();
        }
    }
}

```

Section: [none]

Explanation

Explanation/Reference:

Explanation:

References:

<https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cloud-services-migration-worker-role-stateless-service>

QUESTION 96

You have an existing virtual network with a gateway that is deployed to Azure. You delete all objects that are deployed to the virtual network.

You use the Azure portal to delete the virtual network, but the deletion fails.

You need to determine the cause of the error.

What should you do first?

- A. Delete any local network settings and DNS servers.
- B. Save all settings.
- C. Delete all point-to-site virtual network connections.
- D. Delete the virtual network gateway.

**Correct Answer:** D

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

#### **QUESTION 97**

You have an on-premises Windows Identity Foundation (WIF) application. A section of the application uses resources that are hosted in Azure. The application uses Azure Active Directory (Azure AD) to control access to the section of the application that accesses Azure resources. You synchronize all user principals to Azure Active Directory.

The application has the following requirements:

- Use Windows integrated credentials for single sign-on (SSO).
- Use Azure Active Directory as an identity provider.

You need to create an endpoint to use for web sign-in to the secured section of the application.

Which endpoint should you use?

- A. SAML-P
- B. OAuth
- C. Azure AD Graph API
- D. WS-Federation

**Correct Answer:** D

**Section:** [none]

## Explanation

### Explanation/Reference:

Explanation:

### QUESTION 98

#### DRAG DROP

You plan to connect a customer's on-premises infrastructure to Azure. You have several connections available.

You have the following requirements:

- All connections must be secure.
- All on-premises solutions must support hybrid functionality.

You need to recommend connectivity solutions.

Which solutions should you recommend? To answer, drag the appropriate connection strategy to the correct connection. Each connection strategy may be used once, more than once, or not at all. You may need to drag the split bar between panes or scroll to view content.

### Select and Place:

#### Connection strategies

Existing WAN network

Windows built-in virtual private network (VPN) client

Industry standard IPsec virtual private network (VPN)

External-facing IPv6 address

#### Answer Area

Secure cross-premises connection	Connection strategy
Site-to-site VPN	Connection strategy
Point-to-site VPN	Connection strategy
Azure ExpressRoute	Connection strategy

### Correct Answer:

## Connection strategies

External-facing IPv6 address

## Answer Area

Secure cross-premises connection	Connection strategy
Site-to-site VPN	Industry standard IPsec virtual private network (VPN)
Point-to-site VPN	Windows built-in virtual private network (VPN) client
Azure ExpressRoute	Existing WAN network

Section: [none]

Explanation

Explanation/Reference:

### QUESTION 99

DRAG DROP

You are developer working on a project that will be deployed to Azure. The project includes a local SQL Server database.

You need to migrate the database to Azure SQL.



<https://www.gratisexam.com/>

How should you complete the code segment? To answer, drag the appropriate code segment to the correct location or locations. Each code segment may be used

<https://www.gratisexam.com/>



once, more than once, or not at all. You may need to drag the split bar between panes or scroll to view content.

Select and Place:

### Code segment

sqlpackage.exe

dtexec.exe

Start-AzureSqlDatabaseImport

Start-AzureSqlDatabaseRestore

db.bacpac

db.mdf

### Answer Area

/a:Export  
/ssn:devsql  
/sdn:proddb  
/tf:C:\Temp\

azure storage blob upload C:\Temp\db.bacpac db db.bacpac  
\$ctx = New-AzureSqlDatabaseServerContext -ServerName \$ServerName -Credential \$credential

-SqlConnectionContext \$ctx  
-StorageContainer db  
-DatabaseName proddb  
-BlobName

Correct Answer:

## Code segment

sqlpackage.exe

dtexec.exe

Start-AzureSqlDatabaseImport

Start-AzureSqlDatabaseRestore

db.bacpac

db.mdf

## Answer Area

sqlpackage.exe

/a:Export  
/ssn:devsql  
/sdn:proddb  
/tf:C:\Temp\

db.bacpac

azure storage blob upload C:\Temp\db.bacpac db db.bacpac  
\$ctx = New-AzureSqlDatabaseServerContext -ServerName \$ServerName -Credential \$credential

Start-AzureSqlDatabaseImport

-SqlConnectionContext \$ctx  
-StorageContainer db  
-DatabaseName proddb  
-BlobName

db.bacpac

Section: [none]

Explanation

Explanation/Reference:

### QUESTION 100

#### Case Study

This is a case study. Case studies are not timed separately. You can use as much exam time as you would like to complete each case. However, there may be additional case studies and sections on this exam. You must manage your time to ensure that you are able to complete all questions included on this exam in the time provided.

To answer the questions included in a case study, you will need to reference information that is provided in the case study. Case studies might contain exhibits and other resources that provide more information about the scenario that is described in the case study. Each question is independent of the other question on this case study.

At the end of this case study, a review screen will appear. This screen allows you to review your answers and to make changes before you move to the next sections of the exam. After you begin a new section, you cannot return to this section.

To start the case study

To display the first question on this case study, click the Next button. Use the buttons in the left pane to explore the content of the case study before you answer the questions. Clicking these buttons displays information such as business requirements, existing environment, and problem statements. If the case study has an All Information tab, note that the information displayed is identical to the information displayed on the subsequent tabs. When you are ready to answer a question, click the Question button to return to the question.

## **Background**

You are a developer for LitWare, Inc., a game development company. You are developing a backend service for an online social gaming platform named GamerData. The game is built around point generators, which are associated with physical landmarks. Players claim point generators which give them a set amount of points per day.

## **Business Requirements**

### **Mobile App**

The game itself runs on various mobile devices and is developed by TailSpin Toys, a company that specializes in mobile game development. The mobile app will periodically make calls to the GamerData service to find the five closest point generators that are located less than the specified distance from the player's current location. If no point generators are found, the search distance increases until one is found.

The mobile app shows all the point generators owned by each player. The mobile app allows for each player to search for claimed point generators by player name. This search does not require exact spelling of names. The details for each claimed generator is shown in the app.

When a player claims a point generator, they should receive an email notification. An Azure Function named EmailPlayer has been developed to email players with details about recently claimed point generators.

## **Sponsors**

The platform allows business to sponsor point generators within a business location.

## **Reports**

A report named Daily Sponsor Report must be generated each day at midnight. The report must contain a section for each sponsor. Each sponsor section must contain two subsections.

The first subsection of the report contains the names of the point generators for that sponsor, ordered by the last time the point generator was claimed. The second subsection contains the current owners for each of the point generators for the sponsor. Generation of reports must not impact the GamerData service.

## Technical Requirements

### GamerData Service

All data for the GamerData service is stored in an Azure DocumentDB instance named GamerData. Business and players interact with the service by using a REST API.

The REST API must:

- Produce valid Swagger API specifications for non-obsolete actions.
- Be optimized for loading specific point generators.
- Follow REST best practices.
- Include appropriate terms of service.

Costs for all Azure services must be minimized.

### Build and Deployment

The GamerData service will be deployed to Azure in a private VNet.

### Security

Sponsors have accounts in an Azure Active Directory (Azure AD) with business-to-consumer (B2C) enabled named litwaregamerdata.onmicrosoft.com managed by Litware, Inc. for both GamerData and LitWare, Inc. services.

Only Litware, Inc. developers and automated testing tools should be able to directly access the GamerData service. All other use of the service must be through Azure API Management. A description of the security practices used during development, available on Microsoft SharePoint, must be available to users of the API under the terms of service.

### Reporting

Azure Search will be used as the source for running reports. The properties of indexes in Azure Search must match the names of the properties in DocumentDB.

### Performance

The Azure DocumentDB must not be used for reporting purposes. All services must perform queries in the data store when possible.

### Application Structure

#### Startup.cs

Relevant portions of the app files are shown below. (Line numbers in the code segments are included for reference only and include a two-character prefix that

denotes the specific file to which they belong.)

```
SP01 public class Startup
SP02 {
SP03     public IConfigurationRoot Configuration { get; }
SP04     public Startup(IHostingEnvironment env)
SP05     {
SP06         var builder = new ConfigurationBuilder().SetBasePath(env.ContentRoot-
Path).AddJsonFile ("appsettings.json");
SP07         Configuration = builder.Build();
SP08     }
SP09     public void ConfigureServices(IServiceCollection services)
SP10     {
SP11         services.AddMvc();
SP12         Services.AddSwaggerGen();
SP13     }
SP14     public void Configure(IApplicationBuilder app, IHostingEnvironment env,
ILoggerFactory loggerFactory)
SP15     {
SP16         app.UseMvc();
SP17         app.UseSwagger();
SP18     }
SP19 }
```

PointController.cs

Relevant portions of the app files are shown below. (Line numbers in the code segments are included for reference only and include a two-character prefix that denotes the specific file to which they belong.)

```

PC01 [Route("api/pointgen")]
PC02     public class PointGeneratorController : Controller
PC03     {
PC04         private static readonly string DatabaseName = "GamerData";
PC05         private static readonly string CollectionName = "PointGenerators";
PC06         private static readonly string EndpointUrl = "...";
PC07         private static readonly string AuthorizationKey = "...";
PC08
PC09         [HttpGet("{name}")]
PC10         public async Task<PointGenerator> Get(string name)
PC11         {
PC12             using (var client = new DocumentClient(new Uri(EndpointUrl),
AuthorizationKey))
PC13             {
PC14                 var response = await client.ReadDocumentAsync(UriFactory.Create-
DocumentUri(DatabaseName, CollectionName, name));
PC15                 return (PointGenerator) (dynamic) response.Resource;
PC16             }
PC17         }
PC18
PC19         [Route("nearby")]
PC20         [HttpGet]
PC21         public IEnumerable<pointGenerator> Nearby(double longitude, double
latitude, long minDistance)
PC22         {
PC23             var location = new Point(longitude, latitude);
PC24             using (var client = new DocumentClient(new Uri(EndpointUrl),
AuthorizationKey))
PC25             {
PC26
PC27             }
PC28         }
PC29
PC30         public async Task<PointGenerator> Update[FromBody] PointGenerator pg)
PC31         {
PC32             using (var client = new DocumentClient(new Uri(EndpointUrl),
AuthorizationKey))
PC33             {
PC34                 var collection = await GetCollection();
PC35                 await client.UpsertDocumentAsync(collection.SelfLink, pg);
PC36                 return pg;
PC37             }
PC38         }
PC39         private static async Task<DocumentCollection> GetCollection()
PC40         {
PC41             ...
PC42         }
PC43     }

```

PointGenerator.cs

Relevant portions of the app files are shown below. (Line numbers in the code segments are included for reference only and include a two-character prefix that denotes the specific file to which they belong.)

```
PG01 public class PointGenerator
PG02 {
PG02
PG04     public string Name { get; set; }
PG05     [JsonProperty("currentOwner")]
PG06     public string CurrentOwner { get; set; }
PG07     [JsonProperty("sponsor")]
PG08     public string Sponsor { get; set; }
PG09     [JsonProperty("dateLastClaimed")]
PG10     public DateTimeOffset DateLastClaimed { get; set; }
PG12     [JsonProperty("location")]
PG12     public Point Location { get; set; }
PG13 }
```

You need to create the Azure Search index.

How should you configure the Azure Search index? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

**Hot Area:**

FIELD NAME	TYPE	RETRIEVABLE	FILTERABLE	SORTABLE
currentOwner	Edm.String	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
sponsor	Edm.String	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
dateLastClaimed	Edm.DateTimeOffset	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

**Correct Answer:**

FIELD NAME	TYPE	RETRIEVABLE	FILTERABLE	SORTABLE
currentOwner	Edm.String	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
sponsor	Edm.String	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
dateLastClaimed	Edm.DateTimeOffset	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

**Section:** [none]

**Explanation**

**Explanation/Reference:**

### QUESTION 101

DRAG DROP

#### Case Study

This is a case study. Case studies are not timed separately. You can use as much exam time as you would like to complete each case. However, there may be additional case studies and sections on this exam. You must manage your time to ensure that you are able to complete all questions included on this exam in the time provided.

To answer the questions included in a case study, you will need to reference information that is provided in the case study. Case studies might contain exhibits and other resources that provide more information about the scenario that is described in the case study. Each question is independent of the other question on this case study.

At the end of this case study, a review screen will appear. This screen allows you to review your answers and to make changes before you move to the next sections of the exam. After you begin a new section, you cannot return to this section.

To start the case study

To display the first question on this case study, click the Next button. Use the buttons in the left pane to explore the content of the case study before you answer the questions. Clicking these buttons displays information such as business requirements, existing environment, and problem statements. If the case study has an All Information tab, note that the information displayed is identical to the information displayed on the subsequent tabs. When you are ready to answer a question, click the Question button to return to the question.

#### Background

You are a developer for LitWare, Inc., a game development company. You are developing a backend service for an online social gaming platform named GamerData. The game is built around point generators, which are associated with physical landmarks. Players claim point generators which give them a set amount of points per day.



## **Business Requirements**

### **Mobile App**

The game itself runs on various mobile devices and is developed by TailSpin Toys, a company that specializes in mobile game development. The mobile app will periodically make calls to the GamerData service to find the five closest point generators that are located less than the specified distance from the player's current location. If no point generators are found, the search distance increases until one is found.

The mobile app shows all the point generators owned by each player. The mobile app allows for each player to search for claimed point generators by player name. This search does not require exact spelling of names. The details for each claimed generator is shown in the app.

When a player claims a point generator, they should receive an email notification. An Azure Function named EmailPlayer has been developed to email players with details about recently claimed point generators.

### **Sponsors**

The platform allows business to sponsor point generators within a business location.

### **Reports**

A report named Daily Sponsor Report must be generated each day at midnight. The report must contain a section for each sponsor. Each sponsor section must contain two subsections.

The first subsection of the report contains the names of the point generators for that sponsor, ordered by the last time the point generator was claimed. The second subsection contains the current owners for each of the point generators for the sponsor. Generation of reports must not impact the GamerData service.

## **Technical Requirements**

### **GamerData Service**

All data for the GamerData service is stored in an Azure DocumentDB instance named GamerData. Business and players interact with the service by using a REST API.

The REST API must:

- Produce valid Swagger API specifications for non-obsolete actions.
- Be optimized for loading specific point generators.
- Follow REST best practices.
- Include appropriate terms of service.

Costs for all Azure services must be minimized.

## **Build and Deployment**

The GamerData service will be deployed to Azure in a private VNet.

## **Security**

Sponsors have accounts in an Azure Active Directory (Azure AD) with business-to-consumer (B2C) enabled named litwaregamerdata.onmicrosoft.com managed by Litware, Inc. for both GamerData and LitWare, Inc. services.

Only Litware, Inc. developers and automated testing tools should be able to directly access the GamerData service. All other use of the service must be through Azure API Management. A description of the security practices used during development, available on Microsoft SharePoint, must be available to users of the API under the terms of service.

## **Reporting**

Azure Search will be used as the source for running reports. The properties of indexes in Azure Search must match the names of the properties in DocumentDB.

## **Performance**

The Azure DocumentDB must not be used for reporting purposes. All services must perform queries in the data store when possible.

## **Application Structure**

Startup.cs

Relevant portions of the app files are shown below. (Line numbers in the code segments are included for reference only and include a two-character prefix that denotes the specific file to which they belong.)

```

SP01 public class Startup
SP02 {
SP03     public IConfigurationRoot Configuration { get; }
SP04     public Startup(IHostingEnvironment env)
SP05     {
SP06         var builder = new ConfigurationBuilder().SetBasePath(env.ContentRoot-
Path).AddJsonFile ("appsettings.json");
SP07         Configuration = builder.Build();
SP08     }
SP09     public void ConfigureServices(IServiceCollection services)
SP10     {
SP11         services.AddMvc();
SP12         Services.AddSwaggerGen();
SP13     }
SP14     public void Configure(IApplicationBuilder app, IHostingEnvironment env,
ILoggerFactory loggerFactory)
SP15     {
SP16         app.UseMvc();
SP17         app.UseSwagger();
SP18     }
SP19 }

```

PointController.cs

Relevant portions of the app files are shown below. (Line numbers in the code segments are included for reference only and include a two-character prefix that denotes the specific file to which they belong.)

```

PC01 [Route("api/pointgen")]
PC02     public class PointGeneratorController : Controller
PC03     {
PC04         private static readonly string DatabaseName = "GamerData";
PC05         private static readonly string CollectionName = "PointGenerators";
PC06         private static readonly string EndpointUrl = "...";
PC07         private static readonly string AuthorizationKey = "...";
PC08
PC09         [HttpGet("{name}")]
PC10         public async Task<PointGenerator> Get(string name)
PC11         {
PC12             using (var client = new DocumentClient(new Uri(EndpointUrl),
AuthorizationKey))
PC13             {
PC14                 var response = await client.ReadDocumentAsync(UriFactory.Create-
DocumentUri(DatabaseName, CollectionName, name));
PC15                 return (PointGenerator) (dynamic) response.Resource;
PC16             }
PC17         }
PC18
PC19         [Route("nearby")]
PC20         [HttpGet]
PC21         public IEnumerable<pointGenerator> Nearby(double longitude, double
latitude, long minDistance)
PC22         {
PC23             var location = new Point(longitude, latitude);
PC24             using (var client = new DocumentClient(new Uri(EndpointUrl),
AuthorizationKey))
PC25             {
PC26
PC27             }
PC28         }
PC29
PC30         public async Task<PointGenerator> Update[FromBody] PointGenerator pg)
PC31         {
PC32             using (var client = new DocumentClient(new Uri(EndpointUrl),
AuthorizationKey))
PC33             {
PC34                 var collection = await GetCollection();
PC35                 await client.UpsertDocumentAsync(collection.SelfLink, pg);
PC36                 return pg;
PC37             }
PC38         }
PC39         private static async Task<DocumentCollection> GetCollection()
PC40         {
PC41             ...
PC42         }
PC43     }

```

PointGenerator.cs

Relevant portions of the app files are shown below. (Line numbers in the code segments are included for reference only and include a two-character prefix that denotes the specific file to which they belong.)

```
PG01 public class PointGenerator
PG02     {
PG02
PG04     public string Name { get; set; }
PG05     [JsonProperty("currentOwner")]
PG06     public string CurrentOwner { get; set; }
PG07     [JsonProperty("sponsor")]
PG08     public string Sponsor { get; set; }
PG09     [JsonProperty("dateLastClaimed")]
PG10     public DateTimeOffset DateLastClaimed { get; set; }
PG12     [JsonProperty("location")]
PG12     public Point Location { get; set; }
PG13 }
```

You need to write a method to return the email address for a given sponsor.

What should you do? To answer, drag the appropriate code segment to the correct location. Each code segment may be used once, more than once, or not at all. You may need to drag the split bar between panes or scroll to view content.

**Select and Place:**

## Code segments

```
https://graph.windows.net/litwaregamerdata
```

```
https://litwaregamerdata.onmicrosoft.com/graph
```

```
https://graph.windows.net/litwareinc
```

```
https://litwareinc.onmicrosoft.com/graph
```

```
/users?api-version=1.6&$filter=displayName eq \' {pg.Sponsor} \'
```

```
/users?api-version=1.6&$filter=displayName eq \' {pg.CurrentOwner} \'
```

```
result.value[0].mail
```

```
result.value[0].sipProxyAddress
```



---

## Answer Area

```
private async Task<string> getEmail(PointGenerator pg, string accessToken)
```

```
{    var url = " " ;
```

```
    url += $" " ;
```

**Correct Answer:**

## Code segments

```
https://litwaregamerdata.onmicrosoft.com/graph
```

```
https://graph.windows.net/litwareinc
```

```
https://litwareinc.onmicrosoft.com/graph
```

```
/users?api-version=1.6&$filter=displayName eq \' {pg.CurrentOwner} \'
```

```
result.value[0].sipProxyAddress
```



## Answer Area

```
private async Task<string> getEmail(PointGenerator pg, string accessToken)
{
    var url = "https://graph.windows.net/litwaregamerdata";
    url += $" /users?api-version=1.6&$filter=displayName eq \' {pg.Sponsor} \'
```



**Section: [none]**

**Explanation**

**Explanation/Reference:**

## **QUESTION 102**

### **Case Study**

This is a case study. Case studies are not timed separately. You can use as much exam time as you would like to complete each case. However, there may be additional case studies and sections on this exam. You must manage your time to ensure that you are able to complete all questions included on this exam in the time provided.

To answer the questions included in a case study, you will need to reference information that is provided in the case study. Case studies might contain exhibits and other resources that provide more information about the scenario that is described in the case study. Each question is independent of the other question on this case study.

At the end of this case study, a review screen will appear. This screen allows you to review your answers and to make changes before you move to the next sections of the exam. After you begin a new section, you cannot return to this section.

To start the case study

To display the first question on this case study, click the Next button. Use the buttons in the left pane to explore the content of the case study before you answer the questions. Clicking these buttons displays information such as business requirements, existing environment, and problem statements. If the case study has an All Information tab, note that the information displayed is identical to the information displayed on the subsequent tabs. When you are ready to answer a question, click the Question button to return to the question.

### **Background**

You are a developer for LitWare, Inc., a game development company. You are developing a backend service for an online social gaming platform named GamerData. The game is built around point generators, which are associated with physical landmarks. Players claim point generators which give them a set amount of points per day.

### **Business Requirements**

#### **Mobile App**

The game itself runs on various mobile devices and is developed by TailSpin Toys, a company that specializes in mobile game development. The mobile app will periodically make calls to the GamerData service to find the five closest point generators that are located less than the specified distance from the player's current location. If no point generators are found, the search distance increases until one is found.

The mobile app shows all the point generators owned by each player. The mobile app allows for each player to search for claimed point generators by player name. This search does not require exact spelling of names. The details for each claimed generator is shown in the app.

When a player claims a point generator, they should receive an email notification. An Azure Function named EmailPlayer has been developed to email players with details about recently claimed point generators.

## **Sponsors**

The platform allows business to sponsor point generators within a business location.

## **Reports**

A report named Daily Sponsor Report must be generated each day at midnight. The report must contain a section for each sponsor. Each sponsor section must contain two subsections.

The first subsection of the report contains the names of the point generators for that sponsor, ordered by the last time the point generator was claimed. The second subsection contains the current owners for each of the point generators for the sponsor. Generation of reports must not impact the GamerData service.

## **Technical Requirements**

### **GamerData Service**

All data for the GamerData service is stored in an Azure DocumentDB instance named GamerData. Business and players interact with the service by using a REST API.

The REST API must:

- Produce valid Swagger API specifications for non-obsolete actions.
- Be optimized for loading specific point generators.
- Follow REST best practices.
- Include appropriate terms of service.

Costs for all Azure services must be minimized.

### **Build and Deployment**

The GamerData service will be deployed to Azure in a private VNet.

### **Security**

Sponsors have accounts in an Azure Active Directory (Azure AD) with business-to-consumer (B2C) enabled named litwaregamerdata.onmicrosoft.com managed by Litware, Inc. for both GamerData and LitWare, Inc. services.

Only Litware, Inc. developers and automated testing tools should be able to directly access the GamerData service. All other use of the service must be through Azure API Management. A description of the security practices used during development, available on Microsoft SharePoint, must be available to users of the API under the terms of service.

### **Reporting**

Azure Search will be used as the source for running reports. The properties of indexes in Azure Search must match the names of the properties in DocumentDB.

## Performance

The Azure DocumentDB must not be used for reporting purposes. All services must perform queries in the data store when possible.

## Application Structure

### Startup.cs

Relevant portions of the app files are shown below. (Line numbers in the code segments are included for reference only and include a two-character prefix that denotes the specific file to which they belong.)

```
SP01 public class Startup
SP02 {
SP03     public IConfigurationRoot Configuration { get; }
SP04     public Startup(IHostingEnvironment env)
SP05     {
SP06         var builder = new ConfigurationBuilder().SetBasePath(env.ContentRoot-
SP07             Path).AddJsonFile("appsettings.json");
SP08         Configuration = builder.Build();
SP09     }
SP10     public void ConfigureServices(IServiceCollection services)
SP11     {
SP12         services.AddMvc();
SP13         Services.AddSwaggerGen();
SP14     }
SP15     public void Configure(IApplicationBuilder app, IHostingEnvironment env,
SP16         ILoggerFactory loggerFactory)
SP17     {
SP18         app.UseMvc();
SP19         app.UseSwagger();
SP20     }
SP21 }
```

### PointController.cs

Relevant portions of the app files are shown below. (Line numbers in the code segments are included for reference only and include a two-character prefix that denotes the specific file to which they belong.)



```

PC01 [Route("api/pointgen")]
PC02     public class PointGeneratorController : Controller
PC03     {
PC04         private static readonly string DatabaseName = "GamerData";
PC05         private static readonly string CollectionName = "PointGenerators";
PC06         private static readonly string EndpointUrl = "...";
PC07         private static readonly string AuthorizationKey = "...";
PC08
PC09         [HttpGet("{name}")]
PC10         public async Task<PointGenerator> Get(string name)
PC11         {
PC12             using (var client = new DocumentClient(new Uri(EndpointUrl),
AuthorizationKey))
PC13             {
PC14                 var response = await client.ReadDocumentAsync(UriFactory.Create-
DocumentUri(DatabaseName, CollectionName, name));
PC15                 return (PointGenerator) (dynamic) response.Resource;
PC16             }
PC17         }
PC18
PC19         [Route("nearby")]
PC20         [HttpGet]
PC21         public IEnumerable<pointGenerator> Nearby(double longitude, double
latitude, long minDistance)
PC22         {
PC23             var location = new Point(longitude, latitude);
PC24             using (var client = new DocumentClient(new Uri(EndpointUrl),
AuthorizationKey))
PC25             {
PC26
PC27             }
PC28         }
PC29
PC30         public async Task<PointGenerator> Update[FromBody] PointGenerator pg)
PC31         {
PC32             using (var client = new DocumentClient(new Uri(EndpointUrl),
AuthorizationKey))
PC33             {
PC34                 var collection = await GetCollection();
PC35                 await client.UpsertDocumentAsync(collection.SelfLink, pg);
PC36                 return pg;
PC37             }
PC38         }
PC39         private static async Task<DocumentCollection> GetCollection()
PC40         {
PC41             ...
PC42         }
PC43     }

```

PointGenerator.cs

Relevant portions of the app files are shown below. (Line numbers in the code segments are included for reference only and include a two-character prefix that denotes the specific file to which they belong.)

```
PG01 public class PointGenerator
PG02 {
PG02
PG04     public string Name { get; set; }
PG05     [JsonProperty("currentOwner")]
PG06     public string CurrentOwner { get; set; }
PG07     [JsonProperty("sponsor")]
PG08     public string Sponsor { get; set; }
PG09     [JsonProperty("dateLastClaimed")]
PG10     public DateTimeOffset DateLastClaimed { get; set; }
PG12     [JsonProperty("location")]
PG12     public Point Location { get; set; }
PG13 }
```

You need to add a routing constraint.

Which code segment should you add at line PC29?

- A. [HttpDelete]
- B. [HttpPost]
- C. [HttpOptions]
- D. [HttpsHead]

**Correct Answer:** B

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

**QUESTION 103**

DRAG DROP

**Case Study**

This is a case study. Case studies are not timed separately. You can use as much exam time as you would like to complete each case. However, there may be additional case studies and sections on this exam. You must manage your time to ensure that you are able to complete all questions included on this exam in the time provided.

To answer the questions included in a case study, you will need to reference information that is provided in the case study. Case studies might contain exhibits and other resources that provide more information about the scenario that is described in the case study. Each question is independent of the other question on this case study.

At the end of this case study, a review screen will appear. This screen allows you to review your answers and to make changes before you move to the next sections of the exam. After you begin a new section, you cannot return to this section.

To start the case study

To display the first question on this case study, click the Next button. Use the buttons in the left pane to explore the content of the case study before you answer the questions. Clicking these buttons displays information such as business requirements, existing environment, and problem statements. If the case study has an All Information tab, note that the information displayed is identical to the information displayed on the subsequent tabs. When you are ready to answer a question, click the Question button to return to the question.

## **Background**

You are a developer for LitWare, Inc., a game development company. You are developing a backend service for an online social gaming platform named GamerData. The game is built around point generators, which are associated with physical landmarks. Players claim point generators which give them a set amount of points per day.

## **Business Requirements**

### **Mobile App**

The game itself runs on various mobile devices and is developed by TailSpin Toys, a company that specializes in mobile game development. The mobile app will periodically make calls to the GamerData service to find the five closest point generators that are located less than the specified distance from the player's current location. If no point generators are found, the search distance increases until one is found.

The mobile app shows all the point generators owned by each player. The mobile app allows for each player to search for claimed point generators by player name. This search does not require exact spelling of names. The details for each claimed generator is shown in the app.

When a player claims a point generator, they should receive an email notification. An Azure Function named EmailPlayer has been developed to email players with details about recently claimed point generators.

### **Sponsors**

The platform allows business to sponsor point generators within a business location.

### **Reports**

A report named Daily Sponsor Report must be generated each day at midnight. The report must contain a section for each sponsor. Each sponsor section must contain two subsections.

The first subsection of the report contains the names of the point generators for that sponsor, ordered by the last time the point generator was claimed. The second subsection contains the current owners for each of the point generators for the sponsor. Generation of reports must not impact the GamerData service.

## **Technical Requirements**

### **GamerData Service**

All data for the GamerData service is stored in an Azure DocumentDB instance named GamerData. Business and players interact with the service by using a REST API.

The REST API must:

- Produce valid Swagger API specifications for non-obsolete actions.
- Be optimized for loading specific point generators.
- Follow REST best practices.
- Include appropriate terms of service.

Costs for all Azure services must be minimized.

### **Build and Deployment**

The GamerData service will be deployed to Azure in a private VNet.

### **Security**

Sponsors have accounts in an Azure Active Directory (Azure AD) with business-to-consumer (B2C) enabled named litwaregamerdata.onmicrosoft.com managed by Litware, Inc. for both GamerData and LitWare, Inc. services.

Only Litware, Inc. developers and automated testing tools should be able to directly access the GamerData service. All other use of the service must be through Azure API Management. A description of the security practices used during development, available on Microsoft SharePoint, must be available to users of the API under the terms of service.

### **Reporting**

Azure Search will be used as the source for running reports. The properties of indexes in Azure Search must match the names of the properties in DocumentDB.

### **Performance**

The Azure DocumentDB must not be used for reporting purposes. All services must perform queries in the data store when possible.

### **Application Structure**



## Startup.cs

Relevant portions of the app files are shown below. (Line numbers in the code segments are included for reference only and include a two-character prefix that denotes the specific file to which they belong.)

```
SP01 public class Startup
SP02 {
SP03     public IConfigurationRoot Configuration { get; }
SP04     public Startup(IHostingEnvironment env)
SP05     {
SP06         var builder = new ConfigurationBuilder().SetBasePath(env.ContentRoot-
SP07             Path).AddJsonFile ("appsettings.json");
SP08         Configuration = builder.Build();
SP09     }
SP10     public void ConfigureServices(IServiceCollection services)
SP11     {
SP12         services.AddMvc();
SP13         Services.AddSwaggerGen();
SP14     }
SP15     public void Configure(IApplicationBuilder app, IHostingEnvironment env,
SP16         ILoggerFactory loggerFactory)
SP17     {
SP18         app.UseMvc();
SP19         app.UseSwagger();
SP20     }
SP21 }
```

## PointController.cs

Relevant portions of the app files are shown below. (Line numbers in the code segments are included for reference only and include a two-character prefix that denotes the specific file to which they belong.)

```

PC01 [Route("api/pointgen")]
PC02     public class PointGeneratorController : Controller
PC03     {
PC04         private static readonly string DatabaseName = "GamerData";
PC05         private static readonly string CollectionName = "PointGenerators";
PC06         private static readonly string EndpointUrl = "...";
PC07         private static readonly string AuthorizationKey = "...";
PC08
PC09         [HttpGet("{name}")]
PC10         public async Task<PointGenerator> Get(string name)
PC11         {
PC12             using (var client = new DocumentClient(new Uri(EndpointUrl),
AuthorizationKey))
PC13             {
PC14                 var response = await client.ReadDocumentAsync(UriFactory.Create-
DocumentUri(DatabaseName, CollectionName, name));
PC15                 return (PointGenerator) (dynamic) response.Resource;
PC16             }
PC17         }
PC18
PC19         [Route("nearby")]
PC20         [HttpGet]
PC21         public IEnumerable<pointGenerator> Nearby(double longitude, double
latitude, long minDistance)
PC22         {
PC23             var location = new Point(longitude, latitude);
PC24             using (var client = new DocumentClient(new Uri(EndpointUrl),
AuthorizationKey))
PC25             {
PC26
PC27             }
PC28         }
PC29
PC30         public async Task<PointGenerator> Update[FromBody] PointGenerator pg)
PC31         {
PC32             using (var client = new DocumentClient(new Uri(EndpointUrl),
AuthorizationKey))
PC33             {
PC34                 var collection = await GetCollection();
PC35                 await client.UpsertDocumentAsync(collection.SelfLink, pg);
PC36                 return pg;
PC37             }
PC38         }
PC39         private static async Task<DocumentCollection> GetCollection()
PC40         {
PC41             ...
PC42         }
PC43     }

```

PointGenerator.cs

Relevant portions of the app files are shown below. (Line numbers in the code segments are included for reference only and include a two-character prefix that denotes the specific file to which they belong.)

```
PG01 public class PointGenerator
PG02     {
PG02
PG04     public string Name { get; set; }
PG05     [JsonProperty("currentOwner")]
PG06     public string CurrentOwner { get; set; }
PG07     [JsonProperty("sponsor")]
PG08     public string Sponsor { get; set; }
PG09     [JsonProperty("dateLastClaimed")]
PG10     public DateTimeOffset DateLastClaimed { get; set; }
PG12     [JsonProperty("location")]
PG12     public Point Location { get; set; }
PG13 }
```

You need to ensure that sponsors can interact with the GamerData service by using the same credentials as they use for other LitWare, Inc. services.

Which three actions should you perform in sequence? To answer, move the appropriate actions from the list of actions to the answer area and arrange them in the correct order.

**Select and Place:**

## Actions

Create a new Azure AD named litware.onmicrosoft.com.

Enable OAuth 2.0 user authorization in Azure AD.

Configure an API Management OAuth 2.0 authorization server.

Configure an Azure AD OAuth 2.0 authorization server.

Enable OAuth 2.0 user authorization in API Management.

Register the API Management developer portal as an Azure AD application in litware.onmicrosoft.com.

Register the API Management developer portal as an Azure AD application in litware2.onmicrosoft.com.

Register the API Management developer portal as an Azure AD application in litwaregamedata.onmicrosoft.com.

## Answer Area



**Correct Answer:**

## Actions

Create a new Azure AD named litware.onmicrosoft.com.

Configure an API Management OAuth 2.0 authorization server.

Enable OAuth 2.0 user authorization in API Management.

Register the API Management developer portal as an Azure AD application in litware.onmicrosoft.com.

Register the API Management developer portal as an Azure AD application in litware2.onmicrosoft.com.

## Answer Area

Enable OAuth 2.0 user authorization in Azure AD.

Register the API Management developer portal as an Azure AD application in litwaregamedata.onmicrosoft.com.

Configure an Azure AD OAuth 2.0 authorization server.



Section: [none]  
Explanation

## **Explanation/Reference:**

### **QUESTION 104**

#### **Case Study**

This is a case study. Case studies are not timed separately. You can use as much exam time as you would like to complete each case. However, there may be additional case studies and sections on this exam. You must manage your time to ensure that you are able to complete all questions included on this exam in the time provided.

To answer the questions included in a case study, you will need to reference information that is provided in the case study. Case studies might contain exhibits and other resources that provide more information about the scenario that is described in the case study. Each question is independent of the other question on this case study.

At the end of this case study, a review screen will appear. This screen allows you to review your answers and to make changes before you move to the next sections of the exam. After you begin a new section, you cannot return to this section.

To start the case study

To display the first question on this case study, click the Next button. Use the buttons in the left pane to explore the content of the case study before you answer the questions. Clicking these buttons displays information such as business requirements, existing environment, and problem statements. If the case study has an All Information tab, note that the information displayed is identical to the information displayed on the subsequent tabs. When you are ready to answer a question, click the Question button to return to the question.

#### **Background**

You are a developer for LitWare, Inc., a game development company. You are developing a backend service for an online social gaming platform named GamerData. The game is built around point generators, which are associated with physical landmarks. Players claim point generators which give them a set amount of points per day.

#### **Business Requirements**

##### **Mobile App**

The game itself runs on various mobile devices and is developed by TailSpin Toys, a company that specializes in mobile game development. The mobile app will periodically make calls to the GamerData service to find the five closest point generators that are located less than the specified distance from the player's current location. If no point generators are found, the search distance increases until one is found.

The mobile app shows all the point generators owned by each player. The mobile app allows for each player to search for claimed point generators by player name. This search does not require exact spelling of names. The details for each claimed generator is shown in the app.

When a player claims a point generator, they should receive an email notification. An Azure Function named EmailPlayer has been developed to email players with details about recently claimed point generators.

##### **Sponsors**

The platform allows business to sponsor point generators within a business location.

## **Reports**

A report named Daily Sponsor Report must be generated each day at midnight. The report must contain a section for each sponsor. Each sponsor section must contain two subsections.

The first subsection of the report contains the names of the point generators for that sponsor, ordered by the last time the point generator was claimed. The second subsection contains the current owners for each of the point generators for the sponsor. Generation of reports must not impact the GamerData service.

## **Technical Requirements**

### **GamerData Service**

All data for the GamerData service is stored in an Azure DocumentDB instance named GamerData. Business and players interact with the service by using a REST API.

The REST API must:

- Produce valid Swagger API specifications for non-obsolete actions.
- Be optimized for loading specific point generators.
- Follow REST best practices.
- Include appropriate terms of service.

Costs for all Azure services must be minimized.

### **Build and Deployment**

The GamerData service will be deployed to Azure in a private VNet.

### **Security**

Sponsors have accounts in an Azure Active Directory (Azure AD) with business-to-consumer (B2C) enabled named litwaregamerdata.onmicrosoft.com managed by Litware, Inc. for both GamerData and LitWare, Inc. services.

Only Litware, Inc. developers and automated testing tools should be able to directly access the GamerData service. All other use of the service must be through Azure API Management. A description of the security practices used during development, available on Microsoft SharePoint, must be available to users of the API under the terms of service.

### **Reporting**

Azure Search will be used as the source for running reports. The properties of indexes in Azure Search must match the names of the properties in DocumentDB.

### **Performance**



The Azure DocumentDB must not be used for reporting purposes. All services must perform queries in the data store when possible.

## Application Structure

### Startup.cs

Relevant portions of the app files are shown below. (Line numbers in the code segments are included for reference only and include a two-character prefix that denotes the specific file to which they belong.)

```
SP01 public class Startup
SP02 {
SP03     public IConfigurationRoot Configuration { get; }
SP04     public Startup(IHostingEnvironment env)
SP05     {
SP06         var builder = new ConfigurationBuilder().SetBasePath(env.ContentRoot-
SP07             Path).AddJsonFile ("appsettings.json");
SP08         Configuration = builder.Build();
SP09     }
SP10     public void ConfigureServices(IServiceCollection services)
SP11     {
SP12         services.AddMvc();
SP13         Services.AddSwaggerGen();
SP14     }
SP15     public void Configure(IApplicationBuilder app, IHostingEnvironment env,
SP16         ILoggerFactory loggerFactory)
SP17     {
SP18         app.UseMvc();
SP19         app.UseSwagger();
SP20     }
SP21 }
```

### PointController.cs

Relevant portions of the app files are shown below. (Line numbers in the code segments are included for reference only and include a two-character prefix that denotes the specific file to which they belong.)



```

PC01 [Route("api/pointgen")]
PC02     public class PointGeneratorController : Controller
PC03     {
PC04         private static readonly string DatabaseName = "GamerData";
PC05         private static readonly string CollectionName = "PointGenerators";
PC06         private static readonly string EndpointUrl = "...";
PC07         private static readonly string AuthorizationKey = "...";
PC08
PC09         [HttpGet("{name}")]
PC10         public async Task<PointGenerator> Get(string name)
PC11         {
PC12             using (var client = new DocumentClient(new Uri(EndpointUrl),
AuthorizationKey))
PC13             {
PC14                 var response = await client.ReadDocumentAsync(UriFactory.Create-
DocumentUri(DatabaseName, CollectionName, name));
PC15                 return (PointGenerator) (dynamic) response.Resource;
PC16             }
PC17         }
PC18
PC19         [Route("nearby")]
PC20         [HttpGet]
PC21         public IEnumerable<pointGenerator> Nearby(double longitude, double
latitude, long minDistance)
PC22         {
PC23             var location = new Point(longitude, latitude);
PC24             using (var client = new DocumentClient(new Uri(EndpointUrl),
AuthorizationKey))
PC25             {
PC26
PC27             }
PC28         }
PC29
PC30         public async Task<PointGenerator> Update[FromBody] PointGenerator pg)
PC31         {
PC32             using (var client = new DocumentClient(new Uri(EndpointUrl),
AuthorizationKey))
PC33             {
PC34                 var collection = await GetCollection();
PC35                 await client.UpsertDocumentAsync(collection.SelfLink, pg);
PC36                 return pg;
PC37             }
PC38         }
PC39         private static async Task<DocumentCollection> GetCollection()
PC40         {
PC41             ...
PC42         }
PC43     }

```

PointGenerator.cs

Relevant portions of the app files are shown below. (Line numbers in the code segments are included for reference only and include a two-character prefix that denotes the specific file to which they belong.)

```
PG01 public class PointGenerator
PG02     {
PG02
PG04     public string Name { get; set; }
PG05     [JsonProperty("currentOwner")]
PG06     public string CurrentOwner { get; set; }
PG07     [JsonProperty("sponsor")]
PG08     public string Sponsor { get; set; }
PG09     [JsonProperty("dateLastClaimed")]
PG10     public DateTimeOffset DateLastClaimed { get; set; }
PG12     [JsonProperty("location")]
PG12     public Point Location { get; set; }
PG13 }
```

You need to write an Azure Search Query to return data for the first subsection of the Daily Sponsor Report.

Which query string should you use?

- A. facets=currentOwner&sort=dateLastClaimed
- B. \$filter=sponsor&sort=dateLastClaimed
- C. search=currentOwner&sort=dateLastClaimed
- D. group=sponsor&sort=dateLastClaimed
- E. facets=sponsor&sort=dateLastClaimed

**Correct Answer: C**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

**QUESTION 105**

**Case Study**

This is a case study. Case studies are not timed separately. You can use as much exam time as you would like to complete each case. However, there may be additional case studies and sections on this exam. You must manage your time to ensure that you are able to complete all questions included on this exam in the time provided.

To answer the questions included in a case study, you will need to reference information that is provided in the case study. Case studies might contain exhibits and other resources that provide more information about the scenario that is described in the case study. Each question is independent of the other question on this case study.

At the end of this case study, a review screen will appear. This screen allows you to review your answers and to make changes before you move to the next sections of the exam. After you begin a new section, you cannot return to this section.

To start the case study

To display the first question on this case study, click the Next button. Use the buttons in the left pane to explore the content of the case study before you answer the questions. Clicking these buttons displays information such as business requirements, existing environment, and problem statements. If the case study has an All Information tab, note that the information displayed is identical to the information displayed on the subsequent tabs. When you are ready to answer a question, click the Question button to return to the question.

## **Background**

You are a developer for LitWare, Inc., a game development company. You are developing a backend service for an online social gaming platform named GamerData. The game is built around point generators, which are associated with physical landmarks. Players claim point generators which give them a set amount of points per day.

## **Business Requirements**

### **Mobile App**

The game itself runs on various mobile devices and is developed by TailSpin Toys, a company that specializes in mobile game development. The mobile app will periodically make calls to the GamerData service to find the five closest point generators that are located less than the specified distance from the player's current location. If no point generators are found, the search distance increases until one is found.

The mobile app shows all the point generators owned by each player. The mobile app allows for each player to search for claimed point generators by player name. This search does not require exact spelling of names. The details for each claimed generator is shown in the app.

When a player claims a point generator, they should receive an email notification. An Azure Function named EmailPlayer has been developed to email players with details about recently claimed point generators.

### **Sponsors**

The platform allows business to sponsor point generators within a business location.

### **Reports**

A report named Daily Sponsor Report must be generated each day at midnight. The report must contain a section for each sponsor. Each sponsor section must contain two subsections.

The first subsection of the report contains the names of the point generators for that sponsor, ordered by the last time the point generator was claimed. The second subsection contains the current owners for each of the point generators for the sponsor. Generation of reports must not impact the GamerData service.

## **Technical Requirements**

### **GamerData Service**

All data for the GamerData service is stored in an Azure DocumentDB instance named GamerData. Business and players interact with the service by using a REST API.

The REST API must:

- Produce valid Swagger API specifications for non-obsolete actions.
- Be optimized for loading specific point generators.
- Follow REST best practices.
- Include appropriate terms of service.

Costs for all Azure services must be minimized.

### **Build and Deployment**

The GamerData service will be deployed to Azure in a private VNet.

### **Security**

Sponsors have accounts in an Azure Active Directory (Azure AD) with business-to-consumer (B2C) enabled named litwaregamerdata.onmicrosoft.com managed by Litware, Inc. for both GamerData and LitWare, Inc. services.

Only Litware, Inc. developers and automated testing tools should be able to directly access the GamerData service. All other use of the service must be through Azure API Management. A description of the security practices used during development, available on Microsoft SharePoint, must be available to users of the API under the terms of service.

### **Reporting**

Azure Search will be used as the source for running reports. The properties of indexes in Azure Search must match the names of the properties in DocumentDB.

### **Performance**

The Azure DocumentDB must not be used for reporting purposes. All services must perform queries in the data store when possible.

### **Application Structure**

## Startup.cs

Relevant portions of the app files are shown below. (Line numbers in the code segments are included for reference only and include a two-character prefix that denotes the specific file to which they belong.)

```
SP01 public class Startup
SP02 {
SP03     public IConfigurationRoot Configuration { get; }
SP04     public Startup(IHostingEnvironment env)
SP05     {
SP06         var builder = new ConfigurationBuilder().SetBasePath(env.ContentRoot-
SP07             Path).AddJsonFile ("appsettings.json");
SP08         Configuration = builder.Build();
SP09     }
SP10     public void ConfigureServices(IServiceCollection services)
SP11     {
SP12         services.AddMvc();
SP13         Services.AddSwaggerGen();
SP14     }
SP15     public void Configure(IApplicationBuilder app, IHostingEnvironment env,
SP16         ILoggerFactory loggerFactory)
SP17     {
SP18         app.UseMvc();
SP19         app.UseSwagger();
SP20     }
SP21 }
```

## PointController.cs

Relevant portions of the app files are shown below. (Line numbers in the code segments are included for reference only and include a two-character prefix that denotes the specific file to which they belong.)

```

PC01 [Route("api/pointgen")]
PC02     public class PointGeneratorController : Controller
PC03     {
PC04         private static readonly string DatabaseName = "GamerData";
PC05         private static readonly string CollectionName = "PointGenerators";
PC06         private static readonly string EndpointUrl = "...";
PC07         private static readonly string AuthorizationKey = "...";
PC08
PC09         [HttpGet("{name}")]
PC10         public async Task<PointGenerator> Get(string name)
PC11         {
PC12             using (var client = new DocumentClient(new Uri(EndpointUrl),
AuthorizationKey))
PC13             {
PC14                 var response = await client.ReadDocumentAsync(UriFactory.Create-
DocumentUri(DatabaseName, CollectionName, name));
PC15                 return (PointGenerator) (dynamic) response.Resource;
PC16             }
PC17         }
PC18
PC19         [Route("nearby")]
PC20         [HttpGet]
PC21         public IEnumerable<pointGenerator> Nearby(double longitude, double
latitude, long minDistance)
PC22         {
PC23             var location = new Point(longitude, latitude);
PC24             using (var client = new DocumentClient(new Uri(EndpointUrl),
AuthorizationKey))
PC25             {
PC26
PC27             }
PC28         }
PC29
PC30         public async Task<PointGenerator> Update[FromBody] PointGenerator pg)
PC31         {
PC32             using (var client = new DocumentClient(new Uri(EndpointUrl),
AuthorizationKey))
PC33             {
PC34                 var collection = await GetCollection();
PC35                 await client.UpsertDocumentAsync(collection.SelfLink, pg);
PC36                 return pg;
PC37             }
PC38         }
PC39         private static async Task<DocumentCollection> GetCollection()
PC40         {
PC41             ...
PC42         }
PC43     }

```

PointGenerator.cs

Relevant portions of the app files are shown below. (Line numbers in the code segments are included for reference only and include a two-character prefix that denotes the specific file to which they belong.)

```
PG01 public class PointGenerator
PG02 {
PG02
PG04     public string Name { get; set; }
PG05     [JsonProperty("currentOwner")]
PG06     public string CurrentOwner { get; set; }
PG07     [JsonProperty("sponsor")]
PG08     public string Sponsor { get; set; }
PG09     [JsonProperty("dateLastClaimed")]
PG10     public DateTimeOffset DateLastClaimed { get; set; }
PG12     [JsonProperty("location")]
PG12     public Point Location { get; set; }
PG13 }
```

You need to trigger the EmailPlayer Azure Function when a point generator is claimed.

What are two possible ways to achieve this goal? Each correct answer presents a complete solution.

- A. Insert code after line PC35 to create a Queue trigger and send a queue message.
- B. Create a trigger based on the primary data store.
- C. Create a trigger based on the reporting data store.
- D. Insert code after line PC14 to create a Service Bus trigger and send a message.

**Correct Answer:** D

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

**QUESTION 106**

**HOTSPOT**

**Case Study**



This is a case study. Case studies are not timed separately. You can use as much exam time as you would like to complete each case. However, there may be additional case studies and sections on this exam. You must manage your time to ensure that you are able to complete all questions included on this exam in the time provided.

To answer the questions included in a case study, you will need to reference information that is provided in the case study. Case studies might contain exhibits and other resources that provide more information about the scenario that is described in the case study. Each question is independent of the other question on this case study.

At the end of this case study, a review screen will appear. This screen allows you to review your answers and to make changes before you move to the next sections of the exam. After you begin a new section, you cannot return to this section.

To start the case study

To display the first question on this case study, click the Next button. Use the buttons in the left pane to explore the content of the case study before you answer the questions. Clicking these buttons displays information such as business requirements, existing environment, and problem statements. If the case study has an All Information tab, note that the information displayed is identical to the information displayed on the subsequent tabs. When you are ready to answer a question, click the Question button to return to the question.

## **Background**

You are a developer for LitWare, Inc., a game development company. You are developing a backend service for an online social gaming platform named GamerData. The game is built around point generators, which are associated with physical landmarks. Players claim point generators which give them a set amount of points per day.

## **Business Requirements**

### **Mobile App**

The game itself runs on various mobile devices and is developed by TailSpin Toys, a company that specializes in mobile game development. The mobile app will periodically make calls to the GamerData service to find the five closest point generators that are located less than the specified distance from the player's current location. If no point generators are found, the search distance increases until one is found.

The mobile app shows all the point generators owned by each player. The mobile app allows for each player to search for claimed point generators by player name. This search does not require exact spelling of names. The details for each claimed generator is shown in the app.

When a player claims a point generator, they should receive an email notification. An Azure Function named EmailPlayer has been developed to email players with details about recently claimed point generators.

### **Sponsors**

The platform allows business to sponsor point generators within a business location.

### **Reports**

A report named Daily Sponsor Report must be generated each day at midnight. The report must contain a section for each sponsor. Each sponsor section must contain two subsections.



The first subsection of the report contains the names of the point generators for that sponsor, ordered by the last time the point generator was claimed. The second subsection contains the current owners for each of the point generators for the sponsor. Generation of reports must not impact the GamerData service.

## **Technical Requirements**

### **GamerData Service**

All data for the GamerData service is stored in an Azure DocumentDB instance named GamerData. Business and players interact with the service by using a REST API.

The REST API must:

- Produce valid Swagger API specifications for non-obsolete actions.
- Be optimized for loading specific point generators.
- Follow REST best practices.
- Include appropriate terms of service.

Costs for all Azure services must be minimized.

### **Build and Deployment**

The GamerData service will be deployed to Azure in a private VNet.

### **Security**

Sponsors have accounts in an Azure Active Directory (Azure AD) with business-to-consumer (B2C) enabled named litwaregamerdata.onmicrosoft.com managed by Litware, Inc. for both GamerData and LitWare, Inc. services.

Only Litware, Inc. developers and automated testing tools should be able to directly access the GamerData service. All other use of the service must be through Azure API Management. A description of the security practices used during development, available on Microsoft SharePoint, must be available to users of the API under the terms of service.

### **Reporting**

Azure Search will be used as the source for running reports. The properties of indexes in Azure Search must match the names of the properties in DocumentDB.

### **Performance**

The Azure DocumentDB must not be used for reporting purposes. All services must perform queries in the data store when possible.

### **Application Structure**

## Startup.cs

Relevant portions of the app files are shown below. (Line numbers in the code segments are included for reference only and include a two-character prefix that denotes the specific file to which they belong.)

```
SP01 public class Startup
SP02 {
SP03     public IConfigurationRoot Configuration { get; }
SP04     public Startup(IHostingEnvironment env)
SP05     {
SP06         var builder = new ConfigurationBuilder().SetBasePath(env.ContentRoot-
SP07             Path).AddJsonFile ("appsettings.json");
SP08         Configuration = builder.Build();
SP09     }
SP10     public void ConfigureServices(IServiceCollection services)
SP11     {
SP12         services.AddMvc();
SP13         Services.AddSwaggerGen();
SP14     }
SP15     public void Configure(IApplicationBuilder app, IHostingEnvironment env,
SP16         ILoggerFactory loggerFactory)
SP17     {
SP18         app.UseMvc();
SP19         app.UseSwagger();
SP20     }
SP21 }
```

## PointController.cs

Relevant portions of the app files are shown below. (Line numbers in the code segments are included for reference only and include a two-character prefix that denotes the specific file to which they belong.)

```

PC01 [Route("api/pointgen")]
PC02     public class PointGeneratorController : Controller
PC03     {
PC04         private static readonly string DatabaseName = "GamerData";
PC05         private static readonly string CollectionName = "PointGenerators";
PC06         private static readonly string EndpointUrl = "...";
PC07         private static readonly string AuthorizationKey = "...";
PC08
PC09         [HttpGet("{name}")]
PC10         public async Task<PointGenerator> Get(string name)
PC11         {
PC12             using (var client = new DocumentClient(new Uri(EndpointUrl),
AuthorizationKey))
PC13             {
PC14                 var response = await client.ReadDocumentAsync(UriFactory.Create-
DocumentUri(DatabaseName, CollectionName, name));
PC15                 return (PointGenerator) (dynamic) response.Resource;
PC16             }
PC17         }
PC18
PC19         [Route("nearby")]
PC20         [HttpGet]
PC21         public IEnumerable<pointGenerator> Nearby(double longitude, double
latitude, long minDistance)
PC22         {
PC23             var location = new Point(longitude, latitude);
PC24             using (var client = new DocumentClient(new Uri(EndpointUrl),
AuthorizationKey))
PC25             {
PC26
PC27             }
PC28         }
PC29
PC30         public async Task<PointGenerator> Update[FromBody] PointGenerator pg)
PC31         {
PC32             using (var client = new DocumentClient(new Uri(EndpointUrl),
AuthorizationKey))
PC33             {
PC34                 var collection = await GetCollection();
PC35                 await client.UpsertDocumentAsync(collection.SelfLink, pg);
PC36                 return pg;
PC37             }
PC38         }
PC39         private static async Task<DocumentCollection> GetCollection()
PC40         {
PC41             ...
PC42         }
PC43     }

```

PointGenerator.cs

Relevant portions of the app files are shown below. (Line numbers in the code segments are included for reference only and include a two-character prefix that denotes the specific file to which they belong.)

```
PG01 public class PointGenerator
PG02     {
PG02
PG04         public string Name { get; set; }
PG05         [JsonProperty("currentOwner")]
PG06         public string CurrentOwner { get; set; }
PG07         [JsonProperty("sponsor")]
PG08         public string Sponsor { get; set; }
PG09         [JsonProperty("dateLastClaimed")]
PG10         public DateTimeOffset DateLastClaimed { get; set; }
PG12         [JsonProperty("location")]
PG12         public Point Location { get; set; }
PG13     }
```

You need to build a Swagger specification for creating the GamerData managed API.

What should you do? To answer, select the appropriate options in the answer area.

**Hot Area:**

```

"paths": {
  "/api/pointgen/{name}": {
    "get": {
      "parameters": [
        {
          "name": "name", "in": "
          ▼", "required": true,
          "type": "string"
        }
      ],
    },
    "/api/pointgen/nearby": {
      "get": {
        "parameters": [
          {
            "name": "longitude", "in": "
            ▼", "required":
            true, "type": "number", "format": "double"
          },
          {
            "name": "latitude", "in": "
            ▼", "required": true,
            "type": "number", "format": "double"
          },
          {
            "name": "minDistance", "in": "
            ▼", "required": true,

```

**Correct Answer:**

```

"paths": {
  "/api/pointgen/{name}": {
    "get": {
      "parameters": [
        {
          "name": "name", "in": "
", "required": true,
          "type": "string"
        }
      ],
    },
    "/api/pointgen/nearby": {
      "get": {
        "parameters": [
          {
            "name": "longitude", "in": "
", "required":
            true, "type": "number", "format": "double"
          },
          {
            "name": "latitude", "in": "
", "required": true,
            "type": "number", "format": "double"
          },
          {
            "name": "minDistance", "in": "
", "required": true,

```

**Section: [none]**

**Explanation**

**Explanation/Reference:**

## **QUESTION 107**

### **Case Study**

This is a case study. Case studies are not timed separately. You can use as much exam time as you would like to complete each case. However, there may be additional case studies and sections on this exam. You must manage your time to ensure that you are able to complete all questions included on this exam in the time provided.

To answer the questions included in a case study, you will need to reference information that is provided in the case study. Case studies might contain exhibits and other resources that provide more information about the scenario that is described in the case study. Each question is independent of the other question on this case study.

At the end of this case study, a review screen will appear. This screen allows you to review your answers and to make changes before you move to the next sections of the exam. After you begin a new section, you cannot return to this section.

To start the case study

To display the first question on this case study, click the Next button. Use the buttons in the left pane to explore the content of the case study before you answer the questions. Clicking these buttons displays information such as business requirements, existing environment, and problem statements. If the case study has an All Information tab, note that the information displayed is identical to the information displayed on the subsequent tabs. When you are ready to answer a question, click the Question button to return to the question.

### **Background**

You are a developer for LitWare, Inc., a game development company. You are developing a backend service for an online social gaming platform named GamerData. The game is built around point generators, which are associated with physical landmarks. Players claim point generators which give them a set amount of points per day.

### **Business Requirements**

#### **Mobile App**

The game itself runs on various mobile devices and is developed by TailSpin Toys, a company that specializes in mobile game development. The mobile app will periodically make calls to the GamerData service to find the five closest point generators that are located less than the specified distance from the player's current location. If no point generators are found, the search distance increases until one is found.

The mobile app shows all the point generators owned by each player. The mobile app allows for each player to search for claimed point generators by player name. This search does not require exact spelling of names. The details for each claimed generator is shown in the app.

When a player claims a point generator, they should receive an email notification. An Azure Function named EmailPlayer has been developed to email players with details about recently claimed point generators.



## **Sponsors**

The platform allows business to sponsor point generators within a business location.

## **Reports**

A report named Daily Sponsor Report must be generated each day at midnight. The report must contain a section for each sponsor. Each sponsor section must contain two subsections.

The first subsection of the report contains the names of the point generators for that sponsor, ordered by the last time the point generator was claimed. The second subsection contains the current owners for each of the point generators for the sponsor. Generation of reports must not impact the GamerData service.

## **Technical Requirements**

### **GamerData Service**

All data for the GamerData service is stored in an Azure DocumentDB instance named GamerData. Business and players interact with the service by using a REST API.

The REST API must:

- Produce valid Swagger API specifications for non-obsolete actions.
- Be optimized for loading specific point generators.
- Follow REST best practices.
- Include appropriate terms of service.

Costs for all Azure services must be minimized.

### **Build and Deployment**

The GamerData service will be deployed to Azure in a private VNet.

### **Security**

Sponsors have accounts in an Azure Active Directory (Azure AD) with business-to-consumer (B2C) enabled named litwaregamerdata.onmicrosoft.com managed by Litware, Inc. for both GamerData and LitWare, Inc. services.

Only Litware, Inc. developers and automated testing tools should be able to directly access the GamerData service. All other use of the service must be through Azure API Management. A description of the security practices used during development, available on Microsoft SharePoint, must be available to users of the API under the terms of service.

### **Reporting**

Azure Search will be used as the source for running reports. The properties of indexes in Azure Search must match the names of the properties in DocumentDB.

## Performance

The Azure DocumentDB must not be used for reporting purposes. All services must perform queries in the data store when possible.

## Application Structure

### Startup.cs

Relevant portions of the app files are shown below. (Line numbers in the code segments are included for reference only and include a two-character prefix that denotes the specific file to which they belong.)

```
SP01 public class Startup
SP02 {
SP03     public IConfigurationRoot Configuration { get; }
SP04     public Startup(IHostingEnvironment env)
SP05     {
SP06         var builder = new ConfigurationBuilder().SetBasePath(env.ContentRoot-
Path).AddJsonFile ("appsettings.json");
SP07         Configuration = builder.Build();
SP08     }
SP09     public void ConfigureServices(IServiceCollection services)
SP10     {
SP11         services.AddMvc();
SP12         Services.AddSwaggerGen();
SP13     }
SP14     public void Configure(IApplicationBuilder app, IHostingEnvironment env,
ILoggerFactory loggerFactory)
SP15     {
SP16         app.UseMvc();
SP17         app.UseSwagger();
SP18     }
SP19 }
```

### PointController.cs

Relevant portions of the app files are shown below. (Line numbers in the code segments are included for reference only and include a two-character prefix that denotes the specific file to which they belong.)



```

PC01 [Route("api/pointgen")]
PC02     public class PointGeneratorController : Controller
PC03     {
PC04         private static readonly string DatabaseName = "GamerData";
PC05         private static readonly string CollectionName = "PointGenerators";
PC06         private static readonly string EndpointUrl = "...";
PC07         private static readonly string AuthorizationKey = "...";
PC08
PC09         [HttpGet("{name}")]
PC10         public async Task<PointGenerator> Get(string name)
PC11         {
PC12             using (var client = new DocumentClient(new Uri(EndpointUrl),
AuthorizationKey))
PC13             {
PC14                 var response = await client.ReadDocumentAsync(UriFactory.Create-
DocumentUri(DatabaseName, CollectionName, name));
PC15                 return (PointGenerator) (dynamic) response.Resource;
PC16             }
PC17         }
PC18
PC19         [Route("nearby")]
PC20         [HttpGet]
PC21         public IEnumerable<pointGenerator> Nearby(double longitude, double
latitude, long minDistance)
PC22         {
PC23             var location = new Point(longitude, latitude);
PC24             using (var client = new DocumentClient(new Uri(EndpointUrl),
AuthorizationKey))
PC25             {
PC26
PC27             }
PC28         }
PC29
PC30         public async Task<PointGenerator> Update[FromBody] PointGenerator pg)
PC31         {
PC32             using (var client = new DocumentClient(new Uri(EndpointUrl),
AuthorizationKey))
PC33             {
PC34                 var collection = await GetCollection();
PC35                 await client.UpsertDocumentAsync(collection.SelfLink, pg);
PC36                 return pg;
PC37             }
PC38         }
PC39         private static async Task<DocumentCollection> GetCollection()
PC40         {
PC41             ...
PC42         }
PC43     }

```

PointGenerator.cs

Relevant portions of the app files are shown below. (Line numbers in the code segments are included for reference only and include a two-character prefix that denotes the specific file to which they belong.)

```
PG01 public class PointGenerator
PG02     {
PG02
PG04     public string Name { get; set; }
PG05     [JsonProperty("currentOwner")]
PG06     public string CurrentOwner { get; set; }
PG07     [JsonProperty("sponsor")]
PG08     public string Sponsor { get; set; }
PG09     [JsonProperty("dateLastClaimed")]
PG10     public DateTimeOffset DateLastClaimed { get; set; }
PG12     [JsonProperty("location")]
PG12     public Point Location { get; set; }
PG13 }
```

You need to decrease the amount of time it takes to query point generators by configuring API management caching.

In the Azure portal, which value should you use for the Vary by Query string parameters setting?

- A. name
- B. longitude;latitude;minDistance
- C. longitude;latitude;dateLastClaimed
- D. Id

**Correct Answer:** B

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

### QUESTION 108

#### Background:

You are a developer for ProseWare Inc., a software-as-a-service (SaaS) company that provides a comment system that websites use to allow for end users to post

comments associated with a webpage or topic on a customer's website.

### **Business requirements**

#### **Moderation:**

The moderation of comments is a feature of the software, and usually involves the editing of a comment.

Only users who have accounts in a group in Azure Active Directory (Azure AD) have the ability moderate. External users can also become moderators, but only by explicit invitation.

Any moderation action must include the name of the moderator.

#### **Comment navigation:**

Each comment is identified by a unique string consisting of a random string of characters.

Within the body of a comment, internal links to other comment threads can be specified using the link format: “/<parent comment id> / <child comment id>”

#### **Comment search:**

Comments can be searched using Azure Search. Searches must do the following:

Searching for email addresses must match email addresses in comments.

Searching must work for the client's language.

Internal links to other comments using the link format should be searched.

#### **Content screening:**

Comment content is screened for inappropriate language, length, and topic using content analysis. Content must be screened, but can appear prior to be screened.

#### **Mobile App:**

The moderation functionality can be accessed using a Universal Windows Platform (UWP) app named ProsewareApp. The app includes functionality that notifies moderators when changes are made to a comment they modified.

#### **Export:**

Customers can perform an export of all comments to a customer supplied Microsoft OneDrive folder on demand. The export functionality is implemented as an Azure Logic App, and it must be able to be triggered by the customer from their local network.

#### **Interaction agents:**

Interaction agents are parts of the system that interact with comment threads. The main purpose is to modify a comment's body based on the contents of the comment. For example, one of the agents is WikiAgent, which adds links to Wikipedia articles when it sees text in the comment body that exactly matches a Wikipedia article title. Interaction Agents are implemented in Service Fabric.

Interaction agents must meet the following requirements:

- Only successfully process each comment once
- Any errors encountered during the processing of a comment should be retried
- Must run on systems that allow for custom applications to be installed

- Must run in a VNet or private network space
- Must be run on a system that can scale up and down based on demand
- A single user's usage of Interaction Agents must not impact other users' usage of Interaction Agents

**Technical requirements****Authentication:**

ProseWare Inc. allows for user authentication through Azure AD and Twitter.

**Storage:**

The application runs as a Web App on Azure. Comments are stored in an Azure DocumentDB database named "Proseware".

**Performance:**

The product includes a service level agreement (SLA) for individual method performance. All data retrieval methods must return within 100ms 99% of the time.

**API:**

The ProseWare Inc. API is made available to public callers using an Azure API App. Azure AD and Twitter are the Authentication Providers.

**Application structure**

CommentController.cs:

```

CC01 [Route("api/[controller]")]
CC02 public class CommentController : Controller
CC03 {
CC04     private IDatabase _redis;
CC05     private DataStore _dataStore;
CC06     private CloudQueue _queue
CC07
CC08     public CommentController ()
CC09     {
CC10         _queue = CloudStorageAccount.Parse(" ").Create-
CloudQueueClient().GetQueueReference ("commentQueue");
CC11         _redis = ConnectionMultiplexer.Connect("...").GetData-
base();
CC12         _dataStore = new DataStore();
CC13     }
CC14
CC15     [HttpGet("{commentId}")]
CC16     public async Task<Comment> Get(string commentId)
CC17     {
CC18         var cached = await _redis.StringGetAsync(commentId);
CC19         if (cached.HasValue)
CC20         {
CC21             return JsonConvert.DeserializeObject<Com-
ment>(cached.ToString());
CC22         }
CC23         return await _dataStore.LoadAsync(commentId);
CC24     }
CC25
CC26     [HttpGet]
CC27     public IEnumerable<Comment> GetChildComments(string com-
mentId)

```



```

CC28 {
CC29     IEnumerable<Comment> result = null;
CC30
CC31     if (result == null)
CC32     {
CC33         result = _dataStore.LoadThread(commentId);
CC34     }
CC35     return results;
CC36 }
CC37
CC38 [HttpPost]
CC39 public async Task<IActionResult> New([FromBody]Comment
comment)
CC40 {
CC41     await Save(comment);
CC42     return Ok();
CC43 }
CC44
CC45 [HttpPost]
CC46 public async Task<IActionResult> Reply(string inRe-
sponseTo, [FromBody]Comment comment)
CC47 {
CC48     comment.InResponseTo = inResponseTo;
CC49     await Save(comment);
CC50     return View();
CC51 }
CC52
CC53 private static Comment Convert(string json)
CC54 {
CC55     return JsonConvert.DeserializeObject<Comment>(json);
CC56 }
CC57
CC58 private async Task Save(Comment comment, string moderator-
Name = null)
CC59 {
CC60     comment.Moderator = moderatorName;
CC61     var json = JsonConvert.SerializeObject(comment);
CC62     _redis.StringSet(comment.Id, json);
CC62
CC63
CC64     await _queue.AddMessageAsync(new CloudQueueMessage(com-
ment.Id));
CC65     _dataStore.Save(comment);
CC66 }

```

cleaner.csx:

```
CL01 #r "Newtonsoft.Json"
CL02
CL03 using System;
CL04 using Newtonsoft.Json;
CL05 using Newtonsoft.Json.Linq;
CL06 public static void Run(string commentId, object result,
TraceWriter log)
CL07 {
CL08     dynamic comment = JObject.Parse(item);
CL09     ...
CL10     result = comment;
CL11 }
```

ICommentAgent.cs:

```
CA01 public interface ICommentAgent: IActor
CA02     {
CA03     Task<string> ModifyCommentText(string id, string body,
string title);
CA04     }
```

WikiAgent.cs:

```

WA01 [StatePersistence(StatePersistence.Persisted)]
WA02 internal class WikiAgent : Agent, ICommentAgent
WA03 {
WA04 public WikiAgent(ActorService, ActorId id) : base(service,
id) {}
WA05 public async Task<string> ModifyCommentText(string id,
string body, string title)
WA06 {
WA07     try
WA08     {
WA09
WA10         var newBody = scanForLinks(body);
WA11
WA12         return newBody;
WA13     }
WA14     catch
WA15     {
WA16
WA17         throw
WA18     }
WA19 }
WA20 }

```

Comment.cs:

```

CO01 public class Comment
CO02 {
CO03     public string Id {get; set;}
CO04     public string UserId {get; set;}
CO05     public string InResponseTo {get; set;}
CO06     public string Title {get; set;}
CO07     public DateTimeOffset Date {get; set;}
CO08     public string Body {get; set;}
CO09     public string Moderator {get; internal; set;}
CO10 }

```

DataStore.cs:

```

DS01 public class DataStore
DS02 {
DS03     private const string EndpointUrl = "https:
//proseware.documents.azure.com:443/";
DS04     private const string PrimaryKey = "";
DS05     private const string db = "Proseware";
DS06     private const string col = "Comments";
DS07     private DocumentClient client;
DS08
DS09     public DataStore()
DS10     {
DS11         client = new DocumentClient(new Uri(EndpointUrl), Pri-
maryKey);
DS12     }
DS13
DS14     public async Task<Comment> LoadAsync(string commentId)
DS15     {
DS16         var uri = UriFactory.CreateDocumentCollectionUri(db,
col);
DS17         return await client.ReadDocumentAsync<Comment>(UriFac-
tory.CreateDocumentUri(db, col, commentId));
DS18     }
DS19
DS20     public async void Save(Comment comment)
DS21     {
DS22         var uri = UriFactory.CreateDocumentCollectionUri(db,
col, comment.Id);
DS23         await client.UpsertDocumentAsync(uri, comment);
DS24     }
DS25     public IEnumerable<Comment> LoadThread(string commentId)
DS26     {
DS27         var uri = UriFactory.CreateDocumentCollectionUri(db,
col);
DS28         return client.CreateDocumentQuery<Comment>(uri).Where(f
=> f.Id == commentId);
DS29     }
}

```

MainPage.xaml.cs:

```
MP01 public sealed partial class MainPage : Page
MP02 {
MP03     public MainPage()
MP04     {
MP05         InitializeComponent();
MP06     }
MP07
MP08     private async void StartNotify()
MP09     {
MP10     }
MP11
MP12     private void UpdateUI()
MP13     {
MP14     }
MP15 }
```

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.

After you answer a question in this section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

You need to implement the infrastructure for the Interaction Agents.

Solution: Create an Azure Container Service cluster and create a container for running Service Fabric.

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer: A**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

## **QUESTION 109**

### **Background:**

You are a developer for ProseWare Inc., a software-as-a-service (SaaS) company that provides a comment system that websites use to allow for end users to post comments associated with a webpage or topic on a customer's website.

### **Business requirements**

#### **Moderation:**

The moderation of comments is a feature of the software, and usually involves the editing of a comment.

Only users who have accounts in a group in Azure Active Directory (Azure AD) have the ability moderate. External users can also become moderators, but only by explicit invitation.

Any moderation action must include the name of the moderator.

#### **Comment navigation:**

Each comment is identified by a unique string consisting of a random string of characters.

Within the body of a comment, internal links to other comment threads can be specified using the link format: “/<parent comment id> / <child comment id>”

#### **Comment search:**

Comments can be searched using Azure Search. Searches must do the following:

- Searching for email addresses must match email addresses in comments.
- Searching must work for the client's language.
- Internal links to other comments using the link format should be searched.

#### **Content screening:**

Comment content is screened for inappropriate language, length, and topic using content analysis. Content must be screened, but can appear prior to be screened.

#### **Mobile App:**

The moderation functionality can be accessed using a Universal Windows Platform (UWP) app named ProsewareApp. The app includes functionality that notifies moderators when changes are made to a comment they modified.

#### **Export:**

Customers can perform an export of all comments to a customer supplied Microsoft OneDrive folder on demand. The export functionality is implemented as an Azure Logic App, and it must be able to be triggered by the customer from their local network.

#### **Interaction agents:**

Interaction agents are parts of the system that interact with comment threads. The main purpose is to modify a comment's body based on the contents of the comment. For example, one of the agents is WikiAgent, which adds links to Wikipedia articles when it sees text in the comment body that exactly matches a Wikipedia article title. Interaction Agents are implemented in Service Fabric.

Interaction agents must meet the following requirements:

- Only successfully process each comment once
- Any errors encountered during the processing of a comment should be retried
- Must run on systems that allow for custom applications to be installed
- Must run in a VNet or private network space
- Must be run on a system that can scale up and down based on demand
- A single user's usage of Interaction Agents must not impact other users' usage of Interaction Agents

### **Technical requirements**

#### **Authentication:**

ProseWare Inc. allows for user authentication through Azure AD and Twitter.

#### **Storage:**

The application runs as a Web App on Azure. Comments are stored in an Azure DocumentDB database named "Proseware".

#### **Performance:**

The product includes a service level agreement (SLA) for individual method performance. All data retrieval methods must return within 100ms 99% of the time.

#### **API:**

The ProseWare Inc. API is made available to public callers using an Azure API App. Azure AD and Twitter are the Authentication Providers.

### **Application structure**

CommentController.cs:

```

CC01 [Route("api/[controller]")]
CC02 public class CommentController : Controller
CC03 {
CC04     private IDatabase _redis;
CC05     private DataStore _dataStore;
CC06     private CloudQueue _queue
CC07
CC08     public CommentController ()
CC09     {
CC10         _queue = CloudStorageAccount.Parse(" ").Create-
CloudQueueClient().GetQueueReference ("commentQueue");
CC11         _redis = ConnectionMultiplexer.Connect("...").GetData-
base();
CC12         _dataStore = new DataStore();
CC13     }
CC14
CC15     [HttpGet("{commentId}")]
CC16     public async Task<Comment> Get(string commentId)
CC17     {
CC18         var cached = await _redis.StringGetAsync(commentId);
CC19         if (cached.HasValue)
CC20         {
CC21             return JsonConvert.DeserializeObject<Com-
ment>(cached.ToString());
CC22         }
CC23         return await _dataStore.LoadAsync(commentId);
CC24     }
CC25
CC26     [HttpGet]
CC27     public IEnumerable<Comment> GetChildComments(string com-
mentId)

```



```

CC28 {
CC29     IEnumerable<Comment> result = null;
CC30
CC31     if (result == null)
CC32     {
CC33         result = _dataStore.LoadThread(commentId);
CC34     }
CC35     return results;
CC36 }
CC37
CC38 [HttpPost]
CC39 public async Task<IActionResult> New([FromBody]Comment
comment)
CC40 {
CC41     await Save(comment);
CC42     return Ok();
CC43 }
CC44
CC45 [HttpPost]
CC46 public async Task<IActionResult> Reply(string inRe-
sponseTo, [FromBody]Comment comment)
CC47 {
CC48     comment.InResponseTo = inResponseTo;
CC49     await Save(comment);
CC50     return View();
CC51 }
CC52
CC53 private static Comment Convert(string json)
CC54 {
CC55     return JsonConvert.DeserializeObject<Comment>(json);
CC56 }
CC57
CC58 private async Task Save(Comment comment, string moderator-
Name = null)
CC59 {
CC60     comment.Moderator = moderatorName;
CC61     var json = JsonConvert.SerializeObject(comment);
CC62     _redis.StringSet(comment.Id, json);
CC62
CC63
CC64     await _queue.AddMessageAsync(new CloudQueueMessage(com-
ment.Id));
CC65     _dataStore.Save(comment);
CC66 }

```

cleaner.csx:

```
CL01 #r "Newtonsoft.Json"
CL02
CL03 using System;
CL04 using Newtonsoft.Json;
CL05 using Newtonsoft.Json.Linq;
CL06 public static void Run(string commentId, object result,
TraceWriter log)
CL07 {
CL08     dynamic comment = JObject.Parse(item);
CL09     ...
CL10     result = comment;
CL11 }
```

ICommentAgent.cs:

```
CA01 public interface ICommentAgent: IActor
CA02     {
CA03     Task<string> ModifyCommentText(string id, string body,
string title);
CA04     }
```

WikiAgent.cs:

```

WA01 [StatePersistence(StatePersistence.Persisted)]
WA02 internal class WikiAgent : Agent, ICommentAgent
WA03 {
WA04 public WikiAgent(ActorService, ActorId id) : base(service,
id) {}
WA05 public async Task<string> ModifyCommentText(string id,
string body, string title)
WA06 {
WA07     try
WA08     {
WA09
WA10         var newBody = scanForLinks(body);
WA11
WA12         return newBody;
WA13     }
WA14     catch
WA15     {
WA16
WA17         throw
WA18     }
WA19 }
WA20 }

```

Comment.cs:

```

CO01 public class Comment
CO02 {
CO03     public string Id {get; set;}
CO04     public string UserId {get; set;}
CO05     public string InResponseTo {get; set;}
CO06     public string Title {get; set;}
CO07     public DateTimeOffset Date {get; set;}
CO08     public string Body {get; set;}
CO09     public string Moderator {get; internal; set;}
CO10 }

```

DataStore.cs:

```

DS01 public class DataStore
DS02 {
DS03     private const string EndpointUrl = "https:
//proseware.documents.azure.com:443/";
DS04     private const string PrimaryKey = "";
DS05     private const string db = "Proseware";
DS06     private const string col = "Comments";
DS07     private DocumentClient client;
DS08
DS09     public DataStore()
DS10     {
DS11         client = new DocumentClient(new Uri(EndpointUrl), Pri-
maryKey);
DS12     }
DS13
DS14     public async Task<Comment> LoadAsync(string commentId)
DS15     {
DS16         var uri = UriFactory.CreateDocumentCollectionUri(db,
col);
DS17         return await client.ReadDocumentAsync<Comment>(UriFac-
tory.CreateDocumentUri(db, col, commentId));
DS18     }
DS19
DS20     public async void Save(Comment comment)
DS21     {
DS22         var uri = UriFactory.CreateDocumentCollectionUri(db,
col, comment.Id);
DS23         await client.UpsertDocumentAsync(uri, comment);
DS24     }
DS25     public IEnumerable<Comment> LoadThread(string commentId)
DS26     {
DS27         var uri = UriFactory.CreateDocumentCollectionUri(db,
col);
DS28         return client.CreateDocumentQuery<Comment>(uri).Where(f
=> f.Id == commentId);
DS29     }
}

```

MainPage.xaml.cs:

```
MP01 public sealed partial class MainPage : Page
MP02 {
MP03     public MainPage()
MP04     {
MP05         InitializeComponent();
MP06     }
MP07
MP08     private async void StartNotify()
MP09     {
MP10     }
MP11
MP12     private void UpdateUI()
MP13     {
MP14     }
MP15 }
```

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.

After you answer a question in this section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

You need to implement the infrastructure for the Interaction Agents.

Solution: Create a set of Azure virtual machines (VMs) using Azure Resource Manager (ARM) templates, and use Chef to install the Service Fabric runtime.

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer: B**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

## **QUESTION 110**

### **Background:**

You are a developer for ProseWare Inc., a software-as-a-service (SaaS) company that provides a comment system that websites use to allow for end users to post comments associated with a webpage or topic on a customer's website.

### **Business requirements**

#### **Moderation:**

The moderation of comments is a feature of the software, and usually involves the editing of a comment.

Only users who have accounts in a group in Azure Active Directory (Azure AD) have the ability moderate. External users can also become moderators, but only by explicit invitation.

Any moderation action must include the name of the moderator.

#### **Comment navigation:**

Each comment is identified by a unique string consisting of a random string of characters.

Within the body of a comment, internal links to other comment threads can be specified using the link format: “/<parent comment id> / <child comment id>”

#### **Comment search:**

Comments can be searched using Azure Search. Searches must do the following:

- Searching for email addresses must match email addresses in comments.
- Searching must work for the client's language.
- Internal links to other comments using the link format should be searched.

#### **Content screening:**

Comment content is screened for inappropriate language, length, and topic using content analysis. Content must be screened, but can appear prior to be screened.

#### **Mobile App:**

The moderation functionality can be accessed using a Universal Windows Platform (UWP) app named ProsewareApp. The app includes functionality that notifies moderators when changes are made to a comment they modified.

#### **Export:**

Customers can perform an export of all comments to a customer supplied Microsoft OneDrive folder on demand. The export functionality is implemented as an Azure Logic App, and it must be able to be triggered by the customer from their local network.

#### **Interaction agents:**

Interaction agents are parts of the system that interact with comment threads. The main purpose is to modify a comment's body based on the contents of the comment. For example, one of the agents is WikiAgent, which adds links to Wikipedia articles when it sees text in the comment body that exactly matches a Wikipedia article title. Interaction Agents are implemented in Service Fabric.

Interaction agents must meet the following requirements:

- Only successfully process each comment once
- Any errors encountered during the processing of a comment should be retried
- Must run on systems that allow for custom applications to be installed
- Must run in a VNet or private network space
- Must be run on a system that can scale up and down based on demand
- A single user's usage of Interaction Agents must not impact other users' usage of Interaction Agents

### **Technical requirements**

#### **Authentication:**

ProseWare Inc. allows for user authentication through Azure AD and Twitter.

#### **Storage:**

The application runs as a Web App on Azure. Comments are stored in an Azure DocumentDB database named "Proseware".

#### **Performance:**

The product includes a service level agreement (SLA) for individual method performance. All data retrieval methods must return within 100ms 99% of the time.

#### **API:**

The ProseWare Inc. API is made available to public callers using an Azure API App. Azure AD and Twitter are the Authentication Providers.

### **Application structure**

CommentController.cs:

```

CC01 [Route("api/[controller]")]
CC02 public class CommentController : Controller
CC03 {
CC04     private IDatabase _redis;
CC05     private DataStore _dataStore;
CC06     private CloudQueue _queue
CC07
CC08     public CommentController ()
CC09     {
CC10         _queue = CloudStorageAccount.Parse(" ").Create-
CloudQueueClient().GetQueueReference ("commentQueue");
CC11         _redis = ConnectionMultiplexer.Connect("...").GetData-
base();
CC12         _dataStore = new DataStore();
CC13     }
CC14
CC15     [HttpGet("{commentId}")]
CC16     public async Task<Comment> Get(string commentId)
CC17     {
CC18         var cached = await _redis.StringGetAsync(commentId);
CC19         if (cached.HasValue)
CC20         {
CC21             return JsonConvert.DeserializeObject<Com-
ment>(cached.ToString());
CC22         }
CC23         return await _dataStore.LoadAsync(commentId);
CC24     }
CC25
CC26     [HttpGet]
CC27     public IEnumerable<Comment> GetChildComments(string com-
mentId)

```



```

CC28 {
CC29     IEnumerable<Comment> result = null;
CC30
CC31     if (result == null)
CC32     {
CC33         result = _dataStore.LoadThread(commentId);
CC34     }
CC35     return results;
CC36 }
CC37
CC38 [HttpPost]
CC39 public async Task<IActionResult> New([FromBody]Comment
comment)
CC40 {
CC41     await Save(comment);
CC42     return Ok();
CC43 }
CC44
CC45 [HttpPost]
CC46 public async Task<IActionResult> Reply(string inRe-
sponseTo, [FromBody]Comment comment)
CC47 {
CC48     comment.InResponseTo = inResponseTo;
CC49     await Save(comment);
CC50     return View();
CC51 }
CC52
CC53 private static Comment Convert(string json)
CC54 {
CC55     return JsonConvert.DeserializeObject<Comment>(json);
CC56 }
CC57
CC58 private async Task Save(Comment comment, string moderator-
Name = null)
CC59 {
CC60     comment.Moderator = moderatorName;
CC61     var json = JsonConvert.SerializeObject(comment);
CC62     _redis.StringSet(comment.Id, json);
CC62
CC63
CC64     await _queue.AddMessageAsync(new CloudQueueMessage(com-
ment.Id));
CC65     _dataStore.Save(comment);
CC66 }

```

cleaner.csx:

```
CL01 #r "Newtonsoft.Json"
CL02
CL03 using System;
CL04 using Newtonsoft.Json;
CL05 using Newtonsoft.Json.Linq;
CL06 public static void Run(string commentId, object result,
TraceWriter log)
CL07 {
CL08     dynamic comment = JObject.Parse(item);
CL09     ...
CL10     result = comment;
CL11 }
```

ICommentAgent.cs:

```
CA01 public interface ICommentAgent: IActor
CA02     {
CA03     Task<string> ModifyCommentText(string id, string body,
string title);
CA04     }
```

WikiAgent.cs:

```

WA01 [StatePersistence(StatePersistence.Persisted)]
WA02 internal class WikiAgent : Agent, ICommentAgent
WA03 {
WA04 public WikiAgent(ActorService, ActorId id) : base(service,
id) {}
WA05 public async Task<string> ModifyCommentText(string id,
string body, string title)
WA06 {
WA07     try
WA08     {
WA09
WA10         var newBody = scanForLinks(body);
WA11
WA12         return newBody;
WA13     }
WA14     catch
WA15     {
WA16
WA17         throw
WA18     }
WA19 }
WA20 }

```

Comment.cs:

```

CO01 public class Comment
CO02 {
CO03     public string Id {get; set;}
CO04     public string UserId {get; set;}
CO05     public string InResponseTo {get; set;}
CO06     public string Title {get; set;}
CO07     public DateTimeOffset Date {get; set;}
CO08     public string Body {get; set;}
CO09     public string Moderator {get; internal; set;}
CO10 }

```

DataStore.cs:

```

DS01 public class DataStore
DS02 {
DS03     private const string EndpointUrl = "https:
//proseware.documents.azure.com:443/";
DS04     private const string PrimaryKey = "";
DS05     private const string db = "Proseware";
DS06     private const string col = "Comments";
DS07     private DocumentClient client;
DS08
DS09     public DataStore()
DS10     {
DS11         client = new DocumentClient(new Uri(EndpointUrl), Pri-
maryKey);
DS12     }
DS13
DS14     public async Task<Comment> LoadAsync(string commentId)
DS15     {
DS16         var uri = UriFactory.CreateDocumentCollectionUri(db,
col);
DS17         return await client.ReadDocumentAsync<Comment>(UriFac-
tory.CreateDocumentUri(db, col, commentId));
DS18     }
DS19
DS20     public async void Save(Comment comment)
DS21     {
DS22         var uri = UriFactory.CreateDocumentCollectionUri(db,
col, comment.Id);
DS23         await client.UpsertDocumentAsync(uri, comment);
DS24     }
DS25     public IEnumerable<Comment> LoadThread(string commentId)
DS26     {
DS27         var uri = UriFactory.CreateDocumentCollectionUri(db,
col);
DS28         return client.CreateDocumentQuery<Comment>(uri).Where(f
=> f.Id == commentId);
DS29     }
}

```

MainPage.xaml.cs:

```
MP01 public sealed partial class MainPage : Page
MP02 {
MP03     public MainPage()
MP04     {
MP05         InitializeComponent();
MP06     }
MP07
MP08     private async void StartNotify()
MP09     {
MP10     }
MP11
MP12     private void UpdateUI()
MP13     {
MP14     }
MP15 }
```

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.

After you answer a question in this section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

You need to implement the infrastructure for the Interaction Agents.

Solution: Create an Azure virtual machine (VM) scale set and use Azure Desired State Configuration (DSC) extension handler to install Service Fabric runtime.

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer:** B

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

## **QUESTION 111**

### **Background:**

You are a developer for ProseWare Inc., a software-as-a-service (SaaS) company that provides a comment system that websites use to allow for end users to post comments associated with a webpage or topic on a customer's website.

### **Business requirements**

#### **Moderation:**

The moderation of comments is a feature of the software, and usually involves the editing of a comment.

Only users who have accounts in a group in Azure Active Directory (Azure AD) have the ability moderate. External users can also become moderators, but only by explicit invitation.

Any moderation action must include the name of the moderator.

#### **Comment navigation:**

Each comment is identified by a unique string consisting of a random string of characters.

Within the body of a comment, internal links to other comment threads can be specified using the link format: “/<parent comment id> / <child comment id>”

#### **Comment search:**

Comments can be searched using Azure Search. Searches must do the following:

- Searching for email addresses must match email addresses in comments.
- Searching must work for the client's language.
- Internal links to other comments using the link format should be searched.

#### **Content screening:**

Comment content is screened for inappropriate language, length, and topic using content analysis. Content must be screened, but can appear prior to be screened.

#### **Mobile App:**

The moderation functionality can be accessed using a Universal Windows Platform (UWP) app named ProsewareApp. The app includes functionality that notifies moderators when changes are made to a comment they modified.

#### **Export:**

Customers can perform an export of all comments to a customer supplied Microsoft OneDrive folder on demand. The export functionality is implemented as an Azure Logic App, and it must be able to be triggered by the customer from their local network.

#### **Interaction agents:**

Interaction agents are parts of the system that interact with comment threads. The main purpose is to modify a comment's body based on the contents of the comment. For example, one of the agents is WikiAgent, which adds links to Wikipedia articles when it sees text in the comment body that exactly matches a Wikipedia article title. Interaction Agents are implemented in Service Fabric.

Interaction agents must meet the following requirements:

- Only successfully process each comment once
- Any errors encountered during the processing of a comment should be retried
- Must run on systems that allow for custom applications to be installed
- Must run in a VNet or private network space
- Must be run on a system that can scale up and down based on demand
- A single user's usage of Interaction Agents must not impact other users' usage of Interaction Agents

### **Technical requirements**

#### **Authentication:**

ProseWare Inc. allows for user authentication through Azure AD and Twitter.

#### **Storage:**

The application runs as a Web App on Azure. Comments are stored in an Azure DocumentDB database named "Proseware".

#### **Performance:**

The product includes a service level agreement (SLA) for individual method performance. All data retrieval methods must return within 100ms 99% of the time.

#### **API:**

The ProseWare Inc. API is made available to public callers using an Azure API App. Azure AD and Twitter are the Authentication Providers.

### **Application structure**

CommentController.cs:

```

CC01 [Route("api/[controller]")]
CC02 public class CommentController : Controller
CC03 {
CC04     private IDatabase _redis;
CC05     private DataStore _dataStore;
CC06     private CloudQueue _queue
CC07
CC08     public CommentController ()
CC09     {
CC10         _queue = CloudStorageAccount.Parse(" ").Create-
CloudQueueClient().GetQueueReference ("commentQueue");
CC11         _redis = ConnectionMultiplexer.Connect("...").GetData-
base();
CC12         _dataStore = new DataStore();
CC13     }
CC14
CC15     [HttpGet("{commentId}")]
CC16     public async Task<Comment> Get(string commentId)
CC17     {
CC18         var cached = await _redis.StringGetAsync(commentId);
CC19         if (cached.HasValue)
CC20         {
CC21             return JsonConvert.DeserializeObject<Com-
ment>(cached.ToString());
CC22         }
CC23         return await _dataStore.LoadAsync(commentId);
CC24     }
CC25
CC26     [HttpGet]
CC27     public IEnumerable<Comment> GetChildComments(string com-
mentId)

```



```

CC28 {
CC29     IEnumerable<Comment> result = null;
CC30
CC31     if (result == null)
CC32     {
CC33         result = _dataStore.LoadThread(commentId);
CC34     }
CC35     return results;
CC36 }
CC37
CC38 [HttpPost]
CC39 public async Task<IActionResult> New([FromBody]Comment
comment)
CC40 {
CC41     await Save(comment);
CC42     return Ok();
CC43 }
CC44
CC45 [HttpPost]
CC46 public async Task<IActionResult> Reply(string inRe-
sponseTo, [FromBody]Comment comment)
CC47 {
CC48     comment.InResponseTo = inResponseTo;
CC49     await Save(comment);
CC50     return View();
CC51 }
CC52
CC53 private static Comment Convert(string json)
CC54 {
CC55     return JsonConvert.DeserializeObject<Comment>(json);
CC56 }
CC57
CC58 private async Task Save(Comment comment, string moderator-
Name = null)
CC59 {
CC60     comment.Moderator = moderatorName;
CC61     var json = JsonConvert.SerializeObject(comment);
CC62     _redis.StringSet(comment.Id, json);
CC62
CC63
CC64     await _queue.AddMessageAsync(new CloudQueueMessage(com-
ment.Id));
CC65     _dataStore.Save(comment);
CC66 }

```

cleaner.csx:

```
CL01 #r "Newtonsoft.Json"
CL02
CL03 using System;
CL04 using Newtonsoft.Json;
CL05 using Newtonsoft.Json.Linq;
CL06 public static void Run(string commentId, object result,
TraceWriter log)
CL07 {
CL08     dynamic comment = JObject.Parse(item);
CL09     ...
CL10     result = comment;
CL11 }
```

ICommentAgent.cs:

```
CA01 public interface ICommentAgent: IActor
CA02     {
CA03     Task<string> ModifyCommentText(string id, string body,
string title);
CA04     }
```

WikiAgent.cs:

```

WA01 [StatePersistence(StatePersistence.Persisted)]
WA02 internal class WikiAgent : Agent, ICommentAgent
WA03 {
WA04 public WikiAgent(ActorService, ActorId id) : base(service,
id) {}
WA05 public async Task<string> ModifyCommentText(string id,
string body, string title)
WA06 {
WA07     try
WA08     {
WA09         var newBody = scanForLinks(body);
WA10
WA11         return newBody;
WA12     }
WA13     catch
WA14     {
WA15
WA16         throw
WA17     }
WA18 }
WA19 }
WA20 }

```

Comment.cs:

```

CO01 public class Comment
CO02 {
CO03     public string Id {get; set;}
CO04     public string UserId {get; set;}
CO05     public string InResponseTo {get; set;}
CO06     public string Title {get; set;}
CO07     public DateTimeOffset Date {get; set;}
CO08     public string Body {get; set;}
CO09     public string Moderator {get; internal; set;}
CO10 }

```

DataStore.cs:

```

DS01 public class DataStore
DS02 {
DS03     private const string EndpointUrl = "https:
//proseware.documents.azure.com:443/";
DS04     private const string PrimaryKey = "";
DS05     private const string db = "Proseware";
DS06     private const string col = "Comments";
DS07     private DocumentClient client;
DS08
DS09     public DataStore()
DS10     {
DS11         client = new DocumentClient(new Uri(EndpointUrl), Pri-
maryKey);
DS12     }
DS13
DS14     public async Task<Comment> LoadAsync(string commentId)
DS15     {
DS16         var uri = UriFactory.CreateDocumentCollectionUri(db,
col);
DS17         return await client.ReadDocumentAsync<Comment>(UriFac-
tory.CreateDocumentUri(db, col, commentId));
DS18     }
DS19
DS20     public async void Save(Comment comment)
DS21     {
DS22         var uri = UriFactory.CreateDocumentCollectionUri(db,
col, comment.Id);
DS23         await client.UpsertDocumentAsync(uri, comment);
DS24     }
DS25     public IEnumerable<Comment> LoadThread(string commentId)
DS26     {
DS27         var uri = UriFactory.CreateDocumentCollectionUri(db,
col);
DS28         return client.CreateDocumentQuery<Comment>(uri).Where(f
=> f.Id == commentId);
DS29     }
}

```

MainPage.xaml.cs:

```
MP01 public sealed partial class MainPage : Page
MP02 {
MP03     public MainPage()
MP04     {
MP05         InitializeComponent();
MP06     }
MP07
MP08     private async void StartNotify()
MP09     {
MP10     }
MP11
MP12     private void UpdateUI()
MP13     {
MP14     }
MP15 }
```

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.

After you answer a question in this section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

You need to implement the infrastructure for the Interaction Agents.

Solution: Create a Service Fabric cluster with Bronze durability and reliability tiers.

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer: A**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

## QUESTION 112

### Background:

You are a developer for ProseWare Inc., a software-as-a-service (SaaS) company that provides a comment system that websites use to allow for end users to post comments associated with a webpage or topic on a customer's website.

### Business requirements

#### Moderation:

The moderation of comments is a feature of the software, and usually involves the editing of a comment.

Only users who have accounts in a group in Azure Active Directory (Azure AD) have the ability moderate. External users can also become moderators, but only by explicit invitation.

Any moderation action must include the name of the moderator.



<https://www.gratisexam.com/>

#### Comment navigation:

Each comment is identified by a unique string consisting of a random string of characters.

Within the body of a comment, internal links to other comment threads can be specified using the link format: “/<parent comment id> / <child comment id>”

#### Comment search:

Comments can be searched using Azure Search. Searches must do the following:

- Searching for email addresses must match email addresses in comments.
- Searching must work for the client's language.
- Internal links to other comments using the link format should be searched.

#### Content screening:

Comment content is screened for inappropriate language, length, and topic using content analysis. Content must be screened, but can appear prior to be screened.

#### Mobile App:

The moderation functionality can be accessed using a Universal Windows Platform (UWP) app named ProsewareApp. The app includes functionality that notifies moderators when changes are made to a comment they modified.

#### Export:

Customers can perform an export of all comments to a customer supplied Microsoft OneDrive folder on demand. The export functionality is implemented as an

<https://www.gratisexam.com/>

Azure Logic App, and it must be able to be triggered by the customer from their local network.

**Interaction agents:**

Interaction agents are parts of the system that interact with comment threads. The main purpose is to modify a comment's body based on the contents of the comment. For example, one of the agents is WikiAgent, which adds links to Wikipedia articles when it sees text in the comment body that exactly matches a Wikipedia article title. Interaction Agents are implemented in Service Fabric.

Interaction agents must meet the following requirements:

- Only successfully process each comment once
- Any errors encountered during the processing of a comment should be retried
- Must run on systems that allow for custom applications to be installed
- Must run in a VNet or private network space
- Must be run on a system that can scale up and down based on demand
- A single user's usage of Interaction Agents must not impact other users' usage of Interaction Agents

**Technical requirements****Authentication:**

ProseWare Inc. allows for user authentication through Azure AD and Twitter.

**Storage:**

The application runs as a Web App on Azure. Comments are stored in an Azure DocumentDB database named "Proseware".

**Performance:**

The product includes a service level agreement (SLA) for individual method performance. All data retrieval methods must return within 100ms 99% of the time.

**API:**

The ProseWare Inc. API is made available to public callers using an Azure API App. Azure AD and Twitter are the Authentication Providers.

**Application structure**

CommentController.cs:

```

CC01 [Route("api/[controller]")]
CC02 public class CommentController : Controller
CC03 {
CC04     private IDatabase _redis;
CC05     private DataStore _dataStore;
CC06     private CloudQueue _queue
CC07
CC08     public CommentController ()
CC09     {
CC10         _queue = CloudStorageAccount.Parse(" ").Create-
CloudQueueClient().GetQueueReference ("commentQueue");
CC11         _redis = ConnectionMultiplexer.Connect("...").GetData-
base();
CC12         _dataStore = new DataStore();
CC13     }
CC14
CC15     [HttpGet("{commentId}")]
CC16     public async Task<Comment> Get(string commentId)
CC17     {
CC18         var cached = await _redis.StringGetAsync(commentId);
CC19         if (cached.HasValue)
CC20         {
CC21             return JsonConvert.DeserializeObject<Com-
ment>(cached.ToString());
CC22         }
CC23         return await _dataStore.LoadAsync(commentId);
CC24     }
CC25
CC26     [HttpGet]
CC27     public IEnumerable<Comment> GetChildComments(string com-
mentId)

```



```

CC28 {
CC29     IEnumerable<Comment> result = null;
CC30
CC31     if (result == null)
CC32     {
CC33         result = _dataStore.LoadThread(commentId);
CC34     }
CC35     return results;
CC36 }
CC37
CC38 [HttpPost]
CC39 public async Task<IActionResult> New([FromBody]Comment
comment)
CC40 {
CC41     await Save(comment);
CC42     return Ok();
CC43 }
CC44
CC45 [HttpPost]
CC46 public async Task<IActionResult> Reply(string inRe-
sponseTo, [FromBody]Comment comment)
CC47 {
CC48     comment.InResponseTo = inResponseTo;
CC49     await Save(comment);
CC50     return View();
CC51 }
CC52
CC53 private static Comment Convert(string json)
CC54 {
CC55     return JsonConvert.DeserializeObject<Comment>(json);
CC56 }
CC57
CC58 private async Task Save(Comment comment, string moderator-
Name = null)
CC59 {
CC60     comment.Moderator = moderatorName;
CC61     var json = JsonConvert.SerializeObject(comment);

```

**cleaner.csx:**

```
CL01 #r "Newtonsoft.Json"
CL02
CL03 using System;
CL04 using Newtonsoft.Json;
CL05 using Newtonsoft.Json.Linq;
CL06 public static void Run(string commentId, object result,
TraceWriter log)
CL07 {
CL08     dynamic comment = JObject.Parse(item);
CL09     ...
CL10     result = comment;
CL11 }
```

**ICommentAgent.cs:**

```
CA01 public interface ICommentAgent: IActor
CA02     {
CA03     Task<string> ModifyCommentText(string id, string body,
string title);
CA04     }
```

**WikiAgent.cs:**

```
WA01 [StatePersistence(StatePersistence.Persisted)]
WA02 internal class WikiAgent : Agent, ICommentAgent
WA03 {
WA04 public WikiAgent(ActorService, ActorId id) : base(service,
id) {}
WA05 public async Task<string> ModifyCommentText(string id,
string body, string title)
WA06 {
WA07     try
WA08     {
WA09
WA10         var newBody = scanForLinks(body);
WA11
WA12         return newBody;
WA13     }
WA14     catch
WA15     {
WA16
WA17         throw
WA18     }
WA19 }
WA20 }
```

**Comment.cs:**

```
CO01 public class Comment
CO02 {
CO03     public string Id {get; set;}
CO04     public string UserId {get; set;}
CO05     public string InResponseTo {get; set;}
CO06     public string Title {get; set;}
CO07     public DateTimeOffset Date {get; set;}
CO08     public string Body {get; set;}
CO09     public string Moderator {get; internal; set;}
CO10 }
```

**DataStore.cs:**

```
DS01 public class DataStore
DS02 {
DS03     private const string EndpointUrl = "https:
//proseware.documents.azure.com:443/";
DS04     private const string PrimaryKey = "";
DS05     private const string db = "Proseware";
DS06     private const string col = "Comments";
DS07     private DocumentClient client;
DS08
DS09     public DataStore()
DS10     {
DS11         client = new DocumentClient(new Uri(EndpointUrl), Pri-
maryKey);
DS12     }
DS13
DS14     public async Task<Comment> LoadAsync(string commentId)
DS15     {
DS16         var uri = UriFactory.CreateDocumentCollectionUri(db,
col);
DS17         return await client.ReadDocumentAsync<Comment>(UriFac-
tory.CreateDocumentUri(db, col, commentId));
DS18     }
DS19
DS20     public async void Save(Comment comment)
DS21     {
DS22         var uri = UriFactory.CreateDocumentCollectionUri(db,
col, comment.Id);
DS23         await client.UpsertDocumentAsync(uri, comment);
DS24     }
DS25     public IEnumerable<Comment> LoadThread(string commentId)
DS26     {
DS27         var uri = UriFactory.CreateDocumentCollectionUri(db,
col);
DS28         return client.CreateDocumentQuery<Comment>(uri).Where(f
=> f.Id == commentId);
DS29     }
```

**MainPage.xaml.cs:**

```
MP01 public sealed partial class MainPage : Page
MP02 {
MP03     public MainPage()
MP04     {
MP05         InitializeComponent();
MP06     }
MP07
MP08     private async void StartNotify()
MP09     {
MP10     }
MP11
MP12     private void UpdateUI()
MP13     {
MP14     }
MP15 }
```

**Note:** This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.

**After you answer a question in this section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.**

You need to ensure that moderators can be added to the system.

Which authentication approach should you use?

- A. Microsoft Office 365 directory
- B. Azure AD self-service signup
- C. Azure AD Organizational Units (OU)
- D. Active Directory Federation

**Correct Answer:** C

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

### QUESTION 113

DRAG DROP

You need to implement the StartNotify method in MainPage.xaml.cs to enable the receiving of notifications.

How should you complete the code? To answer, drag the appropriate code segments to the correct locations. Each code segment may be used once, more than once, or not at all. You may need to drag the split bar between panes or scroll to view content.

NOTE: Each correct selection is worth point.

**Select and Place:**



#### Code segments

PushNotificationChannelManager

Queue

NotificationHub

Registration

moderationnotify

commentQueue

#### Answer area

var a = await

.CreatePushNotificationChannelForApplicationAsync();

var b = new  ("  ", "...");

var result = await b.RegisterNativeAsync(a.Uri);

if (result.Registratiold != null)

{  
    UpdateUI();

}

**Correct Answer:**



### Code segments

Queue

Registration

commentQueue

### Answer area

```
var a = await
```

```
PushNotificationChannelManager.CreatePushNotificationChannelForApplicationAsync();
```

```
var b = new NotificationHub("moderationnotify", "...");
```

```
var result = await b.RegisterNativeAsync(a.Uri);
```

```
if (result.Registratiold != null)
```

```
{
```

```
    UpdateUI();
```

```
}
```

Section: [none]

Explanation

Explanation/Reference:

### QUESTION 114

DRAG DROP

You need to add code at line CC63 to ensure that the Interaction Agent is invoked.

How should you complete the code? To answer, drag the appropriate code segments to the correct locations. Each code segment may be used once, more than once, or not at all. You may need to drag the split bar between panes or scroll to view content.

NOTE: Each correct selection is worth point.

Select and Place:



### Code fragments

`ActorId.CreateRandom()`

`new ActorId(comment.UserId)`

`new ActorId(comment.Id)`

`ActorProxy`

`ServiceProxy`

`ActorServiceProxy`

### Answer area

`var actorId =`   `;`

`var actor =`    
`.Create<ICommentAgent>(actorId, "...");`

`await actor.ModifyCommentText(comment.Id, comment.Body,`  
`comment.Title);`

Correct Answer:

Code fragments

`new ActorId(comment.UserId)`

`new ActorId(comment.Id)`

`ServiceProxy`

`ActorServiceProxy`

Answer area

`var actorId = ActorId.CreateRandom();`  
`var actor = ActorProxy`  
`.Create<ICommentAgent>(actorId, "...");`  
`await actor.ModifyCommentText(comment.Id, comment.Body,`  
`comment.Title);`

Section: [none]

Explanation

Explanation/Reference:

#### QUESTION 115

HOTSPOT

You need to implement exception handling for the ModifyCommentText method in the WikiAgent class.

How should you complete the code? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth point.

Hot Area:

## Answer Area

try

{

	▼
await StateManager.RemoveStateAsync(id);	
if (await StateManager.ContainsStateAsync(id)) return body;	
await StateManager.GetOrAddStateAsync(id, id),	
if (await StateManager.GetStateAsync<string>(id) != null) return body;	

var newBody = ...

	▼
await StateManager.SaveStateAsync();	
await StateManager.ClearCacheAsync();	
await StateManager.AddStateAsync(id, id);	
await StateManager.RemoveStateAsync(id);	

return newBody;

}

catch

{

	▼
await StateManager.SaveStateAsync();	
await StateManager.ClearCacheAsync();	
await StateManager.RemoveStateAsync(id);	
await StateManager.TryRemoveStateAsync(id);	

throw;

}

Correct Answer:

## Answer Area

try

{

	▼
await StateManager.RemoveStateAsync(id);	
if (await StateManager.ContainsStateAsync(id)) return body;	
await StateManager.GetOrAddStateAsync(id, id);	
if (await StateManager.GetStateAsync<string>(id) != null) return body;	

var newBody = ...

	▼
await StateManager.SaveStateAsync();	
await StateManager.ClearCacheAsync();	
await StateManager.AddStateAsync(id, id);	
await StateManager.RemoveStateAsync(id);	

return newBody;

}

catch

{

	▼
await StateManager.SaveStateAsync();	
await StateManager.ClearCacheAsync();	
await StateManager.RemoveStateAsync(id);	
await StateManager.TryRemoveStateAsync(id);	

throw;

}

Section: [none]  
Explanation

**Explanation/Reference:**

**QUESTION 116**

**DRAG DROP**

You need to add code to CommentController.cs to enable moderation of comments.

How should you complete the code? To answer, drag the appropriate code segments to the correct locations. Each code segment may be used once, more than once, or not at all. You may need to drag the split bar between panes or scroll to view content.

NOTE: Each correct selection is worth point.

**Select and Place:**

**Code segments**

Authorize

AllowAnonymous

AutoValidateAntiforgeryToken

Headers ["X-MS-CLIENT-PRINCIPAL-NAME"].First()

Headers ["Proxy-Authorization"].First()

Headers ["X-Forwarded-For"].First()

**Answer area**

[HttpPost]

[

Code segment

]

public async Task<IActionResult> Moderate([FromBody] Comment comm

{

var moderatorName = Request.

Code segment

;

await Save(comment, moderatorName);

return Ok();

}

**Correct Answer:**

### Code segments

AllowAnonymous

AutoValidateAntiforgeryToken

Headers ["Proxy-Authorization"].First()

Headers ["X-Forwarded-For"].First()

### Answer area

```
[HttpPost]
[Authorize]
public async Task<IActionResult> Moderate([FromBody] Comment comm
{
    var moderatorName = Request.
        Headers ["X-MS-CLIENT-PRINCIPAL-NAME"].First() ;
    await Save(comment, moderatorName);
    return Ok();
}
```

Section: [none]

Explanation

Explanation/Reference:

### QUESTION 117

#### CASE STUDY

##### Background:

You are developer for Fabrikam, a company that specializes in payment processing. Fabrikam is developing a solution to process payments for various events, such as music concerts. You develop an ASP.NET MVC website that is hosted in Azure to support an upcoming music concert. The music concert is expected to generate a large volume of ticket sales in a short amount of time.

The website uploads information to an Azure storage queue. A worker role in Azure retrieves information from the queue and generates the concert tickets in a PDF file format after the financial transaction is approved.

You observe a delay between the time the website adds a message to a queue and the time it becomes available to read from the queue. After examining the queue, you determine that no queue messages have a DequeueCount value greater than zero. The website does not throw any errors.

## **Business Requirements**

### **Payments:**

The music concert website must be able to submit event payment information for processing. The website must remain responsive while submitting payment information. Customers must be able to add notes about their orders to a free-form control on the website. These notes must be submitted with the payment when the customer submits an order.

Customers often enter notes that exceed 7 KB in size.

## **Technical Requirements**

### **Payment Submission and Processing:**

Event payment information must be sent from the website to a Windows Communication Foundation (WCF) service worker role. The worker role must submit the information to the payment processor in JSON format.

### **Payment Processing**

You have the following payment processing requirements:

- If the number of messages in a queue goes above or below a specified threshold, worker role instances must be created or deleted as needed. This process must be completed by using the least amount of effort. It must be easy to reconfigure role instance thresholds.
- Payments must be retrieved from the queue in the maximum batch sizes that are allowed by the queue and pulled from the queue for 5 minutes.
- The payment queue must not be re-created when processing payments.
- During single Payment processing, the number of tickets available for an event must be updated. The update operation must be retried for 30 seconds or 5 retry attempts, whichever occurs first. Each retry should pause for at least two seconds and for one second longer than the previous attempt. If the update fails, the payment should be placed in the poison queue.

### **Storage:**

You have the following storage requirements:

- Payment information must be stored by using Azure Queue storage. Connection to the Azure storage account has been established in a configured setting named `StorageConnectionString`, which is configured for the web and worker roles.
- A payment processing queue and a poison payment queue must be used when processing payments.
- Azure Queue message content must be XML-safe and UTF-8 encoded.
- An Azure storage account must be established for diagnostic information in a configured setting named `DiagnosticsStorageConnectionString`, which is configured for both the web and worker roles.

## **Security and Monitoring**

### **Security**

The web role must be secured by using HTTPS.

### **Monitoring**

You must collect diagnostic data for both the web and worker roles by using the Diagnostics module. Diagnostics configuration changes must not require the code of the roles to be rebuilt. The diagnostic data is used for debugging and troubleshooting, measuring performance, monitoring resource usage, traffic analysis and capacity planning, and auditing.



Performance testing must evaluate the roles under normal and stress conditions without incurring changes for running Azure. Memory allocation, function time, and multithreading concurrency issues must be evaluated.

**Deployment:**

You purchase a custom domain name fabrikamfunding.com to host the website, web role, and worker roles. You must deploy an HTTPS certificate with the web role, and you must update associated configuration files accordingly.

Web role and worker role instance sizes must be specified as Medium. You must deploy one web role instance named FabrikamFundingPaymentGenerator, and worker role instances named FabrikamFundingPaymentProcessor.

**Application Structure:**

Relevant portions of the app files are shown below. Line numbers are included for reference only and include a two-character prefix that denotes the specific file to which they belong.

**CustomRetryPolicy.cs**

```
CR01 public class CustomRetryPolicy : IRetryPolicy
CR02 {
CR03     int_retryCount = 0;
CR04     readonly TimeSpan _baseInterval= TimeSpan.FromSeconds(1);
CR05     readonly string _poisonPaymentQueueName;
CR06     private readonly CloudQueueClient _queueClient;
CR07     private readonly EventPayment _eventPayment;
CR08     public CustomRetryPolicy(string poisonPaymentQueueName, CloudQueueClient
queueClient, EventPayment eventPayment)
CR09     {
CR10         _poisonPaymentQueueName = poisonPaymentQueueName;
CR11         _queueClient = queueClient;
CR12         _eventPayment = eventPayment;
CR13     }
CR14     public IRetryPolicy CreateInstance()
CR15     {
CR16         return new CustomRetryPolicy(_poisonPaymentQueueName, _queueClient,
_eventPayment);
CR17     }
CR18 }
```



## Event.cs

```
EV01 public class Event : TableEntity
EV02 {
EV03     public int AvailableTickets { get; set; }
EV04 }
```

## EventPayment.cs

```
EP01 [DataContract]
EP02 public class EventPayment
EP03 {
EP04     [DataMember]
EP05     public int EventId { get; set; }
EP06     [DataMember]
EP07     public string Email { get; set; }
EP08     [DataMember]
EP09     public string Notes { get; set; }
EP10     [DataMember]
EP11     public int TicketCount { get; set; }
EP12     [DataMember]
EP13     public DateTime OrderDate { get; set; }
EP14     [DataMember]
EP15     public Guid EventPaymentId { get; set; }
EP16 }
```

## QueueManager.cs

```
QM01 public class QueueManager
QM02 {
QM03     private readonly CloudQueueClient _queueClient;
QM04     private readonly CloudTableClient _tableClient;
QM05     private const string PaymentQueueName = "paymentqueue";
QM06     private const string PoisonPaymentQueueName = "poisonpaymentqueue";
QM07     public QueueManager()
QM08     {
QM09         var storageAccount = CloudStorageAccount.Parse(
QM10             CloudConfigurationManager.GetSetting("StorageConnectionString"));
QM11         _queueClient = storageAccount.CreateCloudQueueClient();
QM12         _tableClient = storageAccount.CreateCloudTableClient();
QM13     }
QM14     public async Task SendMessageAsync(EventPayment eventPayment)
QM15     {
QM16         ...
QM17     }
QM18     public async Task ProcessMessagesAsync()
QM19     {
QM20         ...
QM21     }
QM22     private async Task ProcessPayment(EventPayment eventPayment)
QM23     {
QM23         var events = _tableClient.GetTableReference("events");
QM23         var key = eventPayment.EventId.ToString();
QM23         var operation = await
QM23             events.ExecuteAsync(TableOperation.Retrieve<Event>(key, key));
QM23         var @event = operation.Result as Event;
QM23         @event.AvailableTickets = @event.AvailableTickets - eventPay-
ment.TicketCount;
QM23         var requestOptions = new TableRequestOptions
QM23         {
QM23             RetryPolicy = new CustomRetryPolicy(
QM23                 PoisonPaymentQueueName,
QM23                 _queueClient,
QM23                 eventPayment),
QM23         };
QM23         var context = new OperationContext
QM23         {
QM23             StartTime = DateTime.Now,
QM23         };
QM23         await events.ExecuteAsync(TableOperation.Replace(@event),
requestOptions, context);
QM40     }
QM41 }
```

You need to diagnose the source of the performance issues when preparing concert tickets.

Which two actions should you perform? Each correct answer presents a complete solution.

NOTE: Each correct selection is worth one point.

- A. Examine the Storage Logging logs for any queue operations that have higher than expected latency.
- B. Examine the Storage Client Library logs to determine whether there is a decrease in the total number of requests for storage operations.
- C. Examine the diagnostic message logs for the worker role to determine whether the worker role is failing to process messages.
- D. Examine the Storage Client Library logs to determine whether there are repeated retries for storage operations.

**Correct Answer:** AD

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

References:

<https://docs.microsoft.com/en-us/azure/storage/storage-monitoring-diagnosing-troubleshooting#you-are-experiencing-unexpected-delays-in-message-delivery>

### **QUESTION 118**

**DRAG DROP**

You are developing an ASP.NET Web App that makes a large number of calls to Azure Blob storage.

You observe that the app suffers from Azure Blob storage throttling.

You need to resolve throttling failures when loading data from Azure Blob storage.

What should you do? To answer, drag the appropriate code segment to the correct location. Each code segment may be used once, more than once, or not at all. You may need to drag the split bar between panes or scroll to view content

NOTE: Each correct selection is worth one point.

**Select and Place:**

### Answer Area

```
var sasBlobUri = "...";
var waitMillisecond = 1000;
while (true) {
    using (var client = new System.Net.Http.HttpClient())
    {
        var response = await client.GetAsync(sasBlobUri);
        if (response.IsSuccessStatusCode)
        {
            return await response.Content.ReadAsByteArrayAsync();
        }
        else
        {
            var statusCode = (int)response.StatusCode;
            if (statusCode == 
                || statusCode ==  )
            {
                waitMillisecond = 
                await Task.Delay(waitMillisecond)
            }
            else
            {
                response.EnsureSuccessStatusCode();
            }
        }
    }
}
```

**Correct Answer:**

400

403

waitMillisecond + 2;

### Answer Area

```
var sasBlobUri = "...";
var waitMillisecond = 1000;
while (true) {
    using (var client = new System.Net.Http.HttpClient())
    {
        var response = await client.GetAsync(sasBlobUri);
        if (response.IsSuccessStatusCode)
        {
            return await response.Content.ReadAsByteArrayAsync();
        }
        else
        {
            var statusCode = (int)response.StatusCode;

            if (statusCode == 500
                || statusCode == 503)
            {
                waitMillisecond = waitMillisecond * 2;

                await Task.Delay(waitMillisecond)
            }
            else
            {
                response.EnsureSuccessStatusCode();
            }
        }
    }
}
```

**Section: [none]**

**Explanation**

**Explanation/Reference:**

## **QUESTION 119**

### **CASE STUDY**

#### **Background:**

You are developer for Fabrikam, a company that specializes in payment processing. Fabrikam is developing a solution to process payments for various events, such as music concerts. You develop an ASP.NET MVC website that is hosted in Azure to support an upcoming music concert. The music concert is expected to generate a large volume of ticket sales in a short amount of time.

The website uploads information to an Azure storage queue. A worker role in Azure retrieves information from the queue and generates the concert tickets in a PDF file format after the financial transaction is approved.

You observe a delay between the time the website adds a message to a queue and the time it becomes available to read from the queue. After examining the queue, you determine that no queue messages have a DequeueCount value greater than zero. The website does not throw any errors.

#### **Business Requirements**

##### **Payments:**

The music concert website must be able to submit event payment information for processing. The website must remain responsive while submitting payment information. Customers must be able to add notes about their orders to a free-form control on the website. These notes must be submitted with the payment when the customer submits an order.

Customers often enter notes that exceed 7 KB in size.

#### **Technical Requirements**

##### **Payment Submission and Processing:**

Event payment information must be sent from the website to a Windows Communication Foundation (WCF) service worker role. The worker role must submit the information to the payment processor in JSON format.

##### **Payment Processing**

You have the following payment processing requirements:

- If the number of messages in a queue goes above or below a specified threshold, worker role instances must be created or deleted as needed. This process must be completed by using the least amount of effort. It must be easy to reconfigure role instance thresholds.
- Payments must be retrieved from the queue in the maximum batch sizes that are allowed by the queue and pulled from the queue for 5 minutes.
- The payment queue must not be re-created when processing payments.
- During single Payment processing, the number of tickets available for an event must be updated. The update operation must be retried for 30 seconds or 5 retry attempts, whichever occurs first. Each retry should pause for at least two seconds and for one second longer than the previous attempt. If the update fails, the payment should be placed in the poison queue.

**Storage:**

You have the following storage requirements:

- Payment information must be stored by using Azure Queue storage. Connection to the Azure storage account has been established in a configured setting named `StorageConnectionString`, which is configured for the web and worker roles.
- A payment processing queue and a poison payment queue must be used when processing payments.
- Azure Queue message content must be XML-safe and UTF-8 encoded.
- An Azure storage account must be established for diagnostic information in a configured setting named `DiagnosticsStorageConnectionString`, which is configured for both the web and worker roles.

**Security and Monitoring****Security**

The web role must be secured by using HTTPS.

**Monitoring**

You must collect diagnostic data for both the web and worker roles by using the Diagnostics module. Diagnostics configuration changes must not require the code of the roles to be rebuilt. The diagnostic data is used for debugging and troubleshooting, measuring performance, monitoring resource usage, traffic analysis and capacity planning, and auditing.

Performance testing must evaluate the roles under normal and stress conditions without incurring changes for running Azure. Memory allocation, function time, and multithreading concurrency issues must be evaluated.

**Deployment:**

You purchase a custom domain name `fabrikamfunding.com` to host the website, web role, and worker roles. You must deploy an HTTPS certificate with the web role, and you must update associated configuration files accordingly.

Web role and worker role instance sizes must be specified as Medium. You must deploy one web role instance named `FabrikamFundingPaymentGenerator`, and worker role instances named `FabrikamFundingPaymentProcessor`.

**Application Structure:**

Relevant portions of the app files are shown below. Line numbers are included for reference only and include a two-character prefix that denotes the specific file to which they belong.



# CustomRetryPolicy.cs

```
CR01 public class CustomRetryPolicy : IRetryPolicy
CR02 {
CR03     int _retryCount = 0;
CR04     readonly TimeSpan _baseInterval= TimeSpan.FromSeconds(1);
CR05     readonly string _poisonPaymentQueueName;
CR06     private readonly CloudQueueClient _queueClient;
CR07     private readonly EventPayment _eventPayment;
CR08     public CustomRetryPolicy(string poisonPaymentQueueName, CloudQueueClient
queueClient, EventPayment eventPayment)
CR09     {
CR10         _poisonPaymentQueueName = poisonPaymentQueueName;
CR11         _queueClient = queueClient;
CR12         _eventPayment = eventPayment;
CR13     }
CR14     public IRetryPolicy CreateInstance()
CR15     {
CR16         return new CustomRetryPolicy(_poisonPaymentQueueName, _queueClient,
_eventPayment);
CR17     }
CR18 }
```

## Event.cs

```
EV01 public class Event : TableEntity
EV02 {
EV03     public int AvailableTickets { get; set; }
EV04 }
```

## EventPayment.cs

```
EP01 [DataContract]
EP02 public class EventPayment
EP03 {
EP04     [DataMember]
EP05     public int EventId { get; set; }
EP06     [DataMember]
EP07     public string Email { get; set; }
EP08     [DataMember]
EP09     public string Notes { get; set; }
EP10     [DataMember]
EP11     public int TicketCount { get; set; }
EP12     [DataMember]
EP13     public DateTime OrderDate { get; set; }
EP14     [DataMember]
EP15     public Guid EventPaymentId { get; set; }
EP16 }
```

## QueueManager.cs

```
QM01 public class QueueManager
QM02 {
QM03     private readonly CloudQueueClient _queueClient;
QM04     private readonly CloudTableClient _tableClient;
QM05     private const string PaymentQueueName = "paymentqueue";
QM06     private const string PoisonPaymentQueueName = "poisonpaymentqueue";
QM07     public QueueManager()
QM08     {
QM09         var storageAccount = CloudStorageAccount.Parse(
QM10             CloudConfigurationManager.GetSetting("StorageConnectionString"));
QM11         _queueClient = storageAccount.CreateCloudQueueClient();
QM12         _tableClient = storageAccount.CreateCloudTableClient();
QM13     }
QM14     public async Task SendMessageAsync(EventPayment eventPayment)
QM15     {
QM16         ...
QM17     }
QM18     public async Task ProcessMessagesAsync()
QM19     {
QM20         ...
QM21     }
QM22     private async Task ProcessPayment(EventPayment eventPayment)
QM23     {
QM23         var events = _tableClient.GetTableReference("events");
QM23         var key = eventPayment.EventId.ToString();
QM23         var operation = await
QM23             events.ExecuteAsync(TableOperation.Retrieve<Event>(key, key));
QM23         var @event = operation.Result as Event;
QM23         @event.AvailableTickets = @event.AvailableTickets - eventPay-
ment.TicketCount;
QM23         var requestOptions = new TableRequestOptions
QM23         {
QM23             RetryPolicy = new CustomRetryPolicy(
QM23                 PoisonPaymentQueueName,
QM23                 _queueClient,
QM23                 eventPayment),
QM23         };
QM23         var context = new OperationContext
QM23         {
QM23             StartTime = DateTime.Now,
QM23         };
QM23         await events.ExecuteAsync(TableOperation.Replace(@event),
requestOptions, context;
QM40     }
QM41 }
```

The SendMessageAsync method of the QueueManager class occasionally throws errors.

You need to correct the errors.

What should you do?

- A. Update the QueueManager to use the Put Message operation of the Queue Service REST API. Use HTTP compression for all calls made to the REST API.
- B. Encode the notesfield content by using UTF-32 encoding.
- C. UpdateSendMessageAsyncmethod of the QueueManagerclass to store the notesfield in BLOB storage. Update the EventPaymentclass to store the BLOB uniform resource identifier (URI). Extract the notes BLOB information by using the BLOB URI in the ProcessMessagesAsyncmethod of the QueueManagerclass.
- D. Update the notesfield to a byte array. Binary encode and decode the notescontent when sending or receiving an EventPaymentclass.

**Correct Answer:** C

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

#### QUESTION 120

##### HOTSPOT

You are developing a multitenant application that uses Azure Search services. You have the following tenants:

Tenant	Requirement
TenantA	The workload and data for this tenant must be isolated from other tenants.
TenantB	The data for this tenant must be isolated from other tenants, but TenantB can share its workload with other tenants.

You must minimize costs associated with implementing any solution. The cost model must be predictable.

You need to design the search experience for the application.

Which Azure Search pattern should you use for each tenant? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

Answer area

Tenant	Pattern
TenantA	<div><div></div><div>▼</div></div> <div><div>index-per-tenant</div><div>service per tenant</div><div>mixed model</div></div>
TenantB	<div><div></div><div>▼</div></div> <div><div>index-per-tenant</div><div>service per tenant</div><div>mixed model</div></div>

Correct Answer:

## Answer area

Tenant	Pattern
TenantA	<div><div></div><div><div>index-per-tenant</div><div>service per tenant</div><div>mixed model</div></div></div>
TenantB	<div><div></div><div><div>index-per-tenant</div><div>service per tenant</div><div>mixed model</div></div></div>

Section: [none]

Explanation

Explanation/Reference:

### QUESTION 121

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.

After you answer a question in this section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

You deploy a Virtual Machine Scale Set (VMSS) named CorpWebVMSS to Azure by using Azure PowerShell and set the instance count to 1. The VMSS includes a storage account, load balancer, public IP address, and six Standard\_A1 Windows virtual machines (VMs) that run Internet Information Services (IIS). All components are deployed to a resource group named CorpWebRG.

You must increase the instance count to support the increased load on IIS.

You need to manually scale out the number of VMs in the scale set to 5.

Solution: You run the following command by using the Azure Command-Line Interface (CLI):

azure vmss scale -g CorpWebRG -n CorpWebVMSS -c 5

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer: A**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

#### **QUESTION 122**

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.

After you answer a question in this section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

You deploy a Virtual Machine Scale Set (VMSS) named CorpWebVMSS to Azure by using Azure PowerShell and set the instance count to 1. The VMSS includes a storage account, load balancer, public IP address, and six Standard\_A1 Windows virtual machines (VMs) that run Internet Information Services (IIS). All components are deployed to a resource group named CorpWebRG.

You must increase the instance count to support the increased load on IIS.

You need to manually scale out the number of VMs in the scale set to 5.

Solution: You deploy the following JSON template by using Azure PowerShell:

```

{
  "$schema": "http://schema.management.azure.com/schemas/2015-01-01-preview/de-
ploymentTemplate.json",
  "contentVersion": "1.0.0.0",
  "resources": [
    {
      "type": "Microsoft.Compute/virtualMachineScaleSets",
      "apiVersion": "2016-03-30",
      "name": "CorpWebVMSS",
      "location": "[resourceGroup().location]",
      "sku": {
        "name": "Standard_A1",
        "tier": "Standard",
        "capacity": "5"
      }
    }
  ]
}

```

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer:** A

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

References:

<https://docs.microsoft.com/en-us/azure/virtual-machine-scale-sets/virtual-machine-scale-sets-autoscale-overview>

### QUESTION 123

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.



After you answer a question in this section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

You deploy a Virtual Machine Scale Set (VMSS) named CorpWebVMSS to Azure by using Azure PowerShell and set the instance count to 1. The VMSS includes a storage account, load balancer, public IP address. and six Standard\_A1 Windows virtual machines (VMs) that run Internet Information Services (IIS). All components are deployed to a resource group named CorpWebRG.

You must increase the instance count to support the increased load on IIS.

You need to manually scale out the number of VMs in the scale set to 5.

Solution: You run the following Azure PowerShell commands:

```
$vmss = Get-AzureRmVmss -ResourceGroupName CorpWebRG -VMScalesSetName CorpWebVMSS  
$vmss.Sku.Capacity = 5  
Update-AzureRmVmss -ResourceGroupName CorpWebRG -Name CorpWebVMSS -VirtualMachineScaleSet $vmss
```

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer: B**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

#### **QUESTION 124**

You manage API management policies in Azure.

You attempt to add a policy that is marked as unavailable.

You need to ensure that you can add the desired policy.

What should you do?

- A. Modify the API Management policy definition.
- B. Enable custom caching for the API Management service.
- C. Modify the scope of the API policy.

D. Integrate the API Management service with the Azure Event Hub service.

**Correct Answer:** C

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

**QUESTION 125**

DRAG DROP

Contoso has an Azure DocumentDB database that contains contact information for customers.

You have a collection named Companies. The collection includes the following data:

```

{
  "id": "ContosoCompany",
  "name": "Contoso",
  "contacts": [
    {
      "giveName": "Lola",
      "surName": "Jacobsen",
      "regions": [
        { "regionName": "West" },
        { "regionName": "South" }
      ]
    },
    {
      "giveName": "David",
      "surName": "Jones",
      "regions": [
        { "regionName": "North" },
        { "regionName": "South" }
      ]
    }
  ],
  "address": { "state": "co", "city": "Denver" }
}

```

You plan to collect the following information for contacts that are located in the South region only:

- Company name
- Given name
- Surname

You need to create the query.

Which three Transact-SQL segments should you use to develop the solution? To answer, move the appropriate Transact-SQL segments from the list of Transact-SQL segments to the answer area and arrange them in the correct order.

Select and Place:

Transact-SQL segments

FROM Contoso c

SELECT Name, givenName, surName

SELECT contoso.Name, c.contacts.givenName,  
c.contacts.surName

WHERE regionName = 'South'

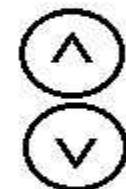
WHERE c.contacts.regions.regionName = 'South'

FROM Companies c

SELECT c.Name, c.contacts.givenName,  
c.contacts.surName



Answer area



Correct Answer:

### Transact-SQL segments

SELECT contoso.Name, c.contacts.givenName, c.contacts.surName
WHERE regionName = 'South'
FROM Companies c
SELECT c.Name, c.contacts.givenName, c.contacts.surName



### Answer area

SELECT Name, givenName, surName

FROM Contoso c

WHERE c.contacts.regions.regionName = 'South'



Section: [none]

Explanation

Explanation/Reference:

#### QUESTION 126

You manage an on-premises server that runs Windows Server 2016. The server has a disk that contains 4 terabytes (TB) of data and thousands of files. None of the individual files are larger than 1 TB. You plan to create a virtual machine (VM) in Azure to process the workload currently handled by the on-premises server.

You need to create a storage location for the data.

What should you do?

- A. Create premium storage account. Use a D-series VM.
- B. Configure a StorSimple virtual array. Configure the VM to use the array with the SMB protocol.
- C. Add a new table storage account. Update the VM workload to use the table storage.

D. Add a singlefile share to the VM. In the VM operating system, assign a drive letter.

**Correct Answer:** D

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

**QUESTION 127**

**DRAG DROP**

You have six Ubuntu Linux virtual machines (VMS) that run a Hadoop cluster on Azure. One of the VMs hosts a custom web user interface that allows users to examine the processing jobs within the Hadoop Cluster.

You need to select the appropriate Azure Storage type for each Azure VM scenario.



<https://www.gratisexam.com/>

Which Azure Storage types should you use? To answer, drag the appropriate Azure Storage type to the correct target. Each Azure Storage type may be used once, more than once, or not at all. You may need to drag the split bar between panes or scroll to view content.

NOTE: Each correct selection is worth one point.

**Select and Place:**

### Azure Storage types

Azure Files

Azure Blobs

Azure Disks

### Answer area

#### Scenario

Provide a Server Message Block (SMB) interface in addition to a REST interface to access files from the VM.

Use REST APIs to store unstructured data for random access and streaming for the VM.

Provide persistent storage attached to the VM.

Mount the share from Ubuntu Linux and access the share by using file system APIs.

Snapshot the VM storage to create point in time read-only backups.

#### Storage Type

Storage Type

Storage Type

Storage Type

Storage Type

Storage Type

Correct Answer:

### Azure Storage types

Azure Files

Azure Blobs

Azure Disks

### Answer area

#### Scenario

Provide a Server Message Block (SMB) interface in addition to a REST interface to access files from the VM.

Use REST APIs to store unstructured data for random access and streaming for the VM.

Provide persistent storage attached to the VM.

Mount the share from Ubuntu Linux and access the share by using file system APIs.

Snapshot the VM storage to create point in time read-only backups.

#### Storage Type

Azure Files

Azure Blobs

Azure Disks

Azure Files

Azure Disks

Section: [none]

Explanation

Explanation/Reference:

#### QUESTION 128

You administer an Azure-based solution that performs image processing. You have four Standard D3 Azure Resource Manager (ARM) virtual machines (VMs). All VMs are deployed in a Virtual Machine Scale Set (VMSS).

The servers must scale up or down as the workload increases or decreases.

You need to configure auto-scaling to scale the VMSS when the server workload is above 95 percent or below 5 percent.

What should you do?

A. Navigate to the VM's Size panel and increase the instance count.



- B. Navigate to the VMSS Metric panel and add a new alert for the CPU Percentage Metric. Configure the alert to notify Via email.
- C. Navigate to the VM's Metric panel and enable diagnostics for basic metrics.
- D. Navigate to the VMSS Metric panel and add a new alert for the CPU Percentage Metric. Configure the alert to notify via webhook.

**Correct Answer:** D

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

References:

<https://docs.microsoft.com/en-us/azure/virtual-machine-scale-sets/virtual-machine-scale-sets-vertical-scale-reprovision>

### **QUESTION 129**

#### **HOTSPOT**

You deploy an application that uses a secure data storage solution to Azure. You use Redis Cache and select the Premium tier.

You have the following requirements:

- Create point-in-time snapshots of the dataset at specific intervals.
- Limit specific clients from using the cache.
- Use primary/replica cache pairs.

You need to configure the environment.

Which feature should you implement for each requirement? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

**Hot Area:**

## Answer area

Requirement	Feature
Create point-in-time snapshots of the dataset at specific intervals.	<div><div></div><div>Redis cluster</div><div>Redis persistence</div><div>Azure Virtual Network</div></div>
Limit specific clients from using the cache.	<div><div></div><div>Redis cluster</div><div>Redis persistence</div><div>Azure Virtual Network</div></div>
Use primary/replica cache pairs.	<div><div></div><div>Redis cluster</div><div>Redis persistence</div><div>Azure Virtual Network</div></div>

Correct Answer:

## Answer area

Requirement	Feature
Create point-in-time snapshots of the dataset at specific intervals.	<div><div></div><div>Redis cluster</div><div>Redis persistence</div><div>Azure Virtual Network</div></div>
Limit specific clients from using the cache.	<div><div></div><div>Redis cluster</div><div>Redis persistence</div><div>Azure Virtual Network</div></div>
Use primary/replica cache pairs.	<div><div></div><div>Redis cluster</div><div>Redis persistence</div><div>Azure Virtual Network</div></div>

Section: [none]

Explanation

Explanation/Reference:

### QUESTION 130

**Note:** This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.

After you answer a question in this sections, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

You are developing a new Azure Logic App. The Logic App requires a custom action to evaluate data from an internal, proprietary system. You create a custom ASP.NET Web API to retrieve data from the system and update the Logic App to use the API.

The Logic App generates a timeout error when it requests data from the API.

You need to eliminate the timeout error and allow the Logic App to retrieve data by using the API.

What should you do?

- A. Update the API to immediately return an HTTP '102 PROCESSING' response when a request is received and an HTTP '205 RESET CONTENT' response when the data is returned from the system.
- B. Update the Logic App to use a new HTTPWebhook trigger to call out to the API's newly-created subscribe and unsubscribe methods.
- C. Update the API to immediately return an HTTP '202 ACCEPTED' response when a request is received and an '200 OK' response when the data is returned from the system.
- D. Update the Logic App adding a wait action to include the interval object's unit and count properties set to valid values.

**Correct Answer: C**

**Section: [none]**

**Explanation**

**Explanation/Reference:**

Explanation:

#### **QUESTION 131**

##### **HOTSPOT**

You plan to migrate an Azure Web App named Contoso from an App Service plan named AppServicePlan1 to another App Service plan. You create a resource group named ContosoGroup.

You create the following Azure PowerShell script. Line numbers are included for reference only.

```
01 $AppServicePlan = @{"serverfarm" = "AppServicePlan2"}
02 Set-AzureResource -name Contoso -ResourceGroupName ContosoGroup -ResourceType
Microsoft.Web/sites ~
    -apiversion 2014-04-01 -PropertyObject $AppServicePlan
03 Get-AzureResource -name Contoso -ResourceGroupName ContosoGroup -ResourceType
Microsoft.Web/sites ~
    -apiversion 2014-04-01
```

For each of the following statements, select Yes if the statement is true. Otherwise, select No.

**Hot Area:**

### Answer Area

	Yes	No
The command in line 01 defines a variable that stores a hash table.	<input type="radio"/>	<input type="radio"/>
The command in line 02 assigns the Web App to the <b>ContosoGroup</b> resource group.	<input type="radio"/>	<input type="radio"/>
The command in line 02 assigns the Web App to a hosting plan named <b>webhostingplan2</b> .	<input type="radio"/>	<input type="radio"/>

Correct Answer:

### Answer Area

	Yes	No
The command in line 01 defines a variable that stores a hash table.	<input checked="" type="radio"/>	<input type="radio"/>
The command in line 02 assigns the Web App to the <b>ContosoGroup</b> resource group.	<input type="radio"/>	<input checked="" type="radio"/>
The command in line 02 assigns the Web App to a hosting plan named <b>webhostingplan2</b> .	<input type="radio"/>	<input checked="" type="radio"/>

Section: [none]

Explanation

Explanation/Reference:

### QUESTION 132

You develop an Azure App Service Mobile App.

The Azure App Service must use Twitter as an authentication provider. You start by registering your application with Twitter.

You need to update your app's authentication and authorization in the Azure Portal.

Which two values should you provide? Each correct answer presents part of the solution.

NOTE: Each correct selection is worth one point.

- A. API Key
- B. Azure Active Directory (Azure AD) Bearer Token
- C. JSON Web Token (JWT)
- D. API Secret
- E. Mobile App gateway URL

**Correct Answer:** AD

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

### **QUESTION 133**

#### **DRAG DROP**

You are developing a business-to-business (B2B) solution by using an Azure Logic App. You plan to use the Enterprise Integration Pack to allow the exchange of the X12 industry standard message format within your Logic App workflow. You start by creating a new Azure Resource Manager (ARM) resource group and Azure App Service plan.

You need to create the B2B solution.

Which four actions should you perform in sequence? To answer, move the appropriate actions from the list of actions to the answer area and arrange them in the correct order.

**Select and Place:**

## Actions

Call the function from the Logic App by using an HTTP trigger.

Create an integration account and the Logic App in the Azure Portal.

Create the Azure Function app.

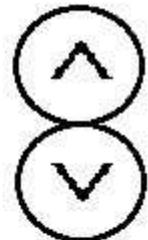
Add a new function in the Azure Function app.

Add partners, schemas, certificates, maps, and agreements.

In your Logic App, use the partners, schemas, certificates, maps, and agreements.

Link the Logic App to the integration account.

## Answer area



**Correct Answer:**

## Actions

Call the function from the Logic App by using an HTTP trigger.

Create the Azure Function app.

Add a new function in the Azure Function app.

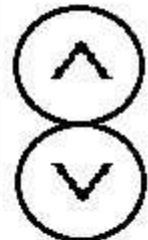
## Answer area

Create an integration account and the Logic App in the Azure Portal.

Add partners, schemas, certificates, maps, and agreements.

Link the Logic App to the integration account.

In your Logic App, use the partners, schemas, certificates, maps, and agreements.



Section: [none]

Explanation

Explanation/Reference:

### QUESTION 134

You are developing an Azure-hosted application that processes request messages for multiple office locations. You create an Azure Service Bus topic named Requests. The topic has a maximum size of 5 gigabytes (GB) and a default message time to live (TTL) of 5 minutes. You also create subscriptions named PriorityRequest and StandardRequest and include appropriate logic to route the messages.



Users report that the application has not processed messages from PriorityRequest in several days.

You need to retrieve the number of messages in the PriorityRequest subscription.

Which metric Should you use?

- A. Subscription Length
- B. Subscription Incoming Requests
- C. Topic Incoming Messages
- D. Topic Size

**Correct Answer:** D

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

### **QUESTION 135**

#### **Background**

Contoso, Ltd. is developing a patient monitoring solution for a hospital. The solution consists of an Azure Web App and a set of mobile applications that health care providers use to monitor patients remotely.

Monitoring devices that run the embedded version of Windows will be attached to patients. The devices will collect information from patients and will transmit real-time continuous data to a service that runs on Azure. The service collects and distributes data. The data that the service provides must be accessible by the website and by the mobile applications.

#### **Business Requirements**

##### **Patients**

All patient data must be stored securely on Azure. Data security must meet or exceed Health Insurance Portability and Accountability Act of 1996 (HIPAA) standards in the United States and must meet or exceed ISO/ICE 27002 data security standards in the rest of the world.

##### **Contractors**

Third-party contractors will develop the mobile applications. All contractors must develop the applications by using virtual machines (VMs) that are hosted on Azure. Only authorized contractors and authorized IP addresses are permitted to access the VMs. The contractors can use Near Field Communication (NFC) tags to launch Remote Desktop (RD) connections to the VMs from NFC-enabled devices. For testing purposes, contractors must be able to run multiple instances of mobile applications within the VMs.

##### **Data Collection and Distribution Service**

The service must monitor the patient data and send out alerts to health care providers when specific conditions are detected. The service must send the alerts to

mobile applications and to the website in real time so that doctors, nurses, and caregivers can attend to the patient. Partner organizations and diagnostic laboratories must be able to securely access the data and the website from remote locations.

### **Current Issues**

A partner that is testing a prototype of the website reports that after signing in to the website, the partner is redirected to the settings page instead of to the home page.

The data from the patient devices is slow to appear on the website and does not always appear. All patient devices online have active connections to the data collection service.

### **Technical Requirements**

#### **Contractors**

All contractors will use virtual machines that are initially configured as size A3. Contractors must sign in to the assigned VM by using IP addresses from a list of preapproved addresses.

#### **Data Collection and Distribution Service**

- The service runs Node.js in a worker role.
- The service must use at least 2048-bit encryption and must use port 8888.
- All patient information must be encrypted and stored by using a NoSQL data store.
- Data must be stored and retrieved securely by using RESTful endpoints.
- Data must NOT be stored within a virtual machine.

All deployed services must send an alert email to [watchguard@contoso.com](mailto:watchguard@contoso.com) when any of the following conditions is met:

- The CPU Percentage metric is at or above 85 percent for at least 10 minutes.
- The Network In metric is at or above 2 KB for at least 10 minutes.
- The Network Out metric is at or above 2 KB for at least 10 minutes.
- The Disk Write metric is at or above 1 KB/sec for at least 30 minutes.
- The Disk Read metric is at or above 1 KB/sec for at least 30 minutes.

#### **Website and Mobile Devices**

The website must be secure and must be accessible only within the hospital's physical grounds. All mobile applications and websites must be responsive. All websites must produce error logs that can be viewed remotely.

#### **Virtual Machines**

- All Azure instances must be deployed and tested on staging instances before they are deployed to production instances.
- All deployed instances must scale up to the next available CPU instance at a CPU usage threshold of 90 percent and scale down when the usage is below 10 percent.

### **Application Structure**

Relevant portions of the application files are shown in the following code segments. Line numbers in the code segments are included for reference only and include a two-character prefix that denotes the specific file to which they belong.

### ControllerFile.cs:

```
CF01 using System;
CF02 using System.Collections.Generic;
CF03 using System.Linq;
CF04 using System.Web;
CF05 using System.Web.Mvc;
CF06 namespace WebApplication1.Controllers
CF07 {
CF08     public class HomeController : Controller
CF09     {
CF10         public ActionResult Index()
CF11         {
CF12             ViewBag.Message = "Welcome to Contoso Patient Monitor.";
CF13
CF14             return View();
CF15         }
CF16         ...
CF17     }
CF18 }
```

## Web.config

```
WC01 <?xml version="1.0" encoding="utf-8"?>
WC02 <configuration>
WC03   <appSettings>
WC04     <add key="webpages:Version" value="3.0.0.0" />
WC05     <add key="webpages:Enabled" value="false" />
WC06     <add key="ClientValidationEnabled" value="true" />
WC07     <add key="UnobtrusiveJavaScriptEnabled" value="true" />
WC08
WC09   </appSettings>
WC10   <system.web>
WC11     <authentication mode="None" />
WC12     <compilation debug="true" targetFramework="4.5" />
WC13     <httpRuntime targetFramework="4.5" />
WC14
WC15   </system.web>
WC16 </configuration>
```

The Web App does not receive alerts quickly enough.

There is a lengthy delay between the time an alert is sent and when it is received by the Web App.

You need to resolve the issue.

What should you do?

- A. Increase the amount of swap memory for the VM instance.
- B. Enable automatic scaling for the Web App.
- C. Decrease the instance count for the worker role.
- D. Enable automatic scaling for the worker role.
- E. Set the monitoring level to Verbose for the worker role.

**Correct Answer: C**

**Section: [none]**

## Explanation

### Explanation/Reference:

Explanation:

From scenario: The data collection service runs Node.js in a worker role.

All deployed instances must scale up to the next available CPU instance at a CPU usage threshold of 90 percent and scale down when the usage is below 10 percent.

### QUESTION 136

**Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.**

**After you answer a question in this sections, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.**

You develop an enterprise application that will be used only by the employees of a company. The application is not Internet-facing. You deploy instances of the application to Azure datacenters on two continents.

You must implement a load balancing solution that meets the following requirements:

- Provide network-level distribution of traffic across all instances of the application.
- Support HTTP and HTTPS protocols.
- Manage all inbound and outbound connections.

Any back-end virtual machine (VM) must be able to service requests from the same user or client session.

Solution: You implement Traffic Manager.

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer: B**

**Section: [none]**

### Explanation

### Explanation/Reference:

Explanation:

A Traffic Manager works at the DNS level. It uses DNS responses to direct end-user traffic to globally distributed endpoints. Clients then connect to those endpoints directly.

An application manager, which works at the Application level (Layer 7), is also required.

References: <https://docs.microsoft.com/en-us/azure/application-gateway/application-gateway-introduction>

### QUESTION 137

**Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.**

**After you answer a question in this sections, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.**

You develop an enterprise application that will be used only by the employees of a company. The application is not Internet-facing. You deploy instances of the application to Azure datacenters on two continents.

You must implement a load balancing solution that meets the following requirements:

- Provide network-level distribution of traffic across all instances of the application.
- Support HTTP and HTTPS protocols.
- Manage all inbound and outbound connections.

Any back-end virtual machine (VM) must be able to service requests from the same user or client session.

Solution: You implement Traffic Manager and Application Gateway.

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer: A**

**Section: [none]**

**Explanation**

#### **Explanation/Reference:**

Explanation:

Application Gateway works at the application layer (Layer 7 in the OSI network reference stack). It acts as a reverse-proxy service, terminating the client connection and forwarding requests to back-end endpoints. It supports the HTTP, HTTPS, and WebSockets protocols.

Application Gateway is useful for applications that require requests from the same user/client session to reach the same back-end virtual machine. Examples of these applications would be shopping cart applications and web mail servers.

Traffic Manager works at the DNS level. It uses DNS responses to direct end-user traffic to globally distributed endpoints. Clients then connect to those endpoints directly.

Microsoft Azure Traffic Manager allows you to control the distribution of user traffic for service endpoints in different datacenters

References: <https://docs.microsoft.com/en-us/azure/traffic-manager/traffic-manager-overview>

**QUESTION 138**

**Note:** This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.

After you answer a question in this sections, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

You develop an enterprise application that will be used only by the employees of a company. The application is not Internet-facing. You deploy instances of the application to Azure datacenters on two continents.

You must implement a load balancing solution that meets the following requirements:

- Provide network-level distribution of traffic across all instances of the application.
- Support HTTP and HTTPS protocols.
- Manage all inbound and outbound connections.

Any back-end virtual machine (VM) must be able to service requests from the same user or client session.

Solution: You implement Application Gateway.

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer:** B

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Traffic Manager is also needed. It allows you to control the distribution of user traffic for service endpoints in different datacenters

References: <https://docs.microsoft.com/en-us/azure/traffic-manager/traffic-manager-overview>

**QUESTION 139**

You administer an Azure environment that includes six Azure Resource Manager (ARM) virtual machines (VMs) that support development. The development team uses Azure SQL databases and Azure Queues for application storage. All Azure resources are grouped within a single subscription and resource group. You need to reduce the recurring monthly Azure costs without degrading server performance. You must minimize the administrative effort involved.

What should you do?

- A. Configure an auto-shutdown schedule for each VM by using the Azure Portal.
- B. Update the development environment to use Azure Table storage.

- C. Create an Azure Automation runbook that compresses unused virtual hard disk (VHD) files daily.
- D. Create an Azure PowerShell script that backs up and deprovisions all Azure SQL databases daily.

**Correct Answer:** B

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Table storage is cheaper than SQL Databases.

It's tempting to go with answer A. However, this would degrade server performance (although shutting the servers down when not in use would save money).

There is another version of this question that does not include shutting down the servers as an answer option.

#### QUESTION 140

**Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.**

**After you answer a question in this sections, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.**

You have a web app that is deployed to Azure.

You need to download a compressed collection of the diagnostic logs.

What should you use?

- A. Azure PowerShell
- B. File Transfer Protocol (FTP)
- C. Application Insights
- D. Microsoft Visual Studio

**Correct Answer:** A

**Section:** [none]

**Explanation**

**Explanation/Reference:**

Explanation:

Diagnostic information stored to the web app file system can be accessed directly using FTP. It can also be downloaded as a Zip archive using Azure PowerShell or the Azure Command-Line Interface.

References: <https://docs.microsoft.com/en-us/azure/app-service/web-sites-enable-diagnostic-log#download>



