

Question #1

Topic 1

**HOTSPOT -**

You are implementing a software as a service (SaaS) ASP.NET Core web service that will run as an Azure Web App. The web service will use an on-premises

SQL Server database for storage. The web service also includes a WebJob that processes data updates. Four customers will use the web service.

Each instance of the WebJob processes data for a single customer and must run as a singleton instance.

Each deployment must be tested by using deployment slots prior to serving production data.

Azure costs must be minimized.

Azure resources must be located in an isolated network.

You need to configure the App Service plan for the Web App.

How should you configure the App Service plan? To answer, select the appropriate settings in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area****App service plan setting      Value**

Number of VM instances

2
4
8
16

Pricing tier

Isolated
Standard
Premium
Consumption

**Answer Area****App service plan setting      Value**

Number of VM instances

2
4
8
16

Correct Answer:

Pricing tier

Isolated
Standard
Premium
Consumption

Number of VM instances: 4 -

You are not charged extra for deployment slots.

Pricing tier: Isolated -

The App Service Environment (ASE) is a powerful feature offering of the Azure App Service that gives network isolation and improved scale capabilities. It is essentially a deployment of the Azure App Service into a subnet of a customer's Azure Virtual Network (VNet).

Reference:

<https://azure.microsoft.com/sv-se/blog/announcing-app-service-isolated-more-power-scale-and-ease-of-use/>



Question #2

Topic 1

**DRAG DROP -**

You are a developer for a software as a service (SaaS) company that uses an Azure Function to process orders. The Azure Function currently runs on an Azure

Function app that is triggered by an Azure Storage queue.

You are preparing to migrate the Azure Function to Kubernetes using Kubernetes-based Event Driven Autoscaling (KEDA).

You need to configure Kubernetes Custom Resource Definitions (CRD) for the Azure Function.

Which CRDs should you configure? To answer, drag the appropriate CRD types to the correct locations. Each CRD 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:

**Answer Area**

CRD types	Setting	CRD type
Secret	Azure Function code	
Deployment	Polling interval	
ScaledObject	Azure Storage connection string	
TriggerAuthentication		

**Answer Area**

CRD types	Setting	CRD type
Secret	Azure Function code	
Deployment	Polling interval	
ScaledObject	Azure Storage connection string	
TriggerAuthentication		

Box 1: Deployment -

To deploy Azure Functions to Kubernetes use the func kubernetes deploy command has several attributes that directly control how our app scales, once it is deployed to Kubernetes.

Box 2: ScaledObject -

With --polling-interval, we can control the interval used by KEDA to check Azure Service Bus Queue for messages.

Example of ScaledObject with polling interval

```
apiVersion: keda.k8s.io/v1alpha1
```

```
kind: ScaledObject
```

```
metadata:
```

```
name: transformer-fn
```

```
namespace: tt
```

```
labels:
```

```
deploymentName: transformer-fn
```

```
spec:
```

```
scaleTargetRef:
```

```
deploymentName: transformer-fn
```

```
pollingInterval: 5
```

```
minReplicaCount: 0
```

```
maxReplicaCount: 100
```

Box 3: Secret -

Store connection strings in Kubernetes Secrets.

Example: to create the Secret in our demo Namespace:

```
# create the k8s demo namespace
kubectl create namespace tt

# grab connection string from Azure Service Bus
KEDA_SCALER_CONNECTION_STRING=$(az servicebus queue authorization-rule keys list \
-g $RG_NAME \
--namespace-name $SBN_NAME \
--queue-name inbound \
-n keda-scaler \
--query "primaryConnectionString" \
-o tsv)

# create the kubernetes secret
kubectl create secret generic tt-keda-auth \
--from-literal KedaScaler=$KEDA_SCALER_CONNECTION_STRING \
--namespace tt

Reference:
https://www.thinktecture.com/en/kubernetes/serverless-workloads-with-keda/
```

## Question #3

## HOTSPOT -

You are creating a CLI script that creates an Azure web app and related services in Azure App Service. The web app uses the following variables:

Variable name	Value
\$gitrepo	<a href="https://github.com/Contos/webapp">https://github.com/Contos/webapp</a>
\$webappname	Webapp1103

You need to automatically deploy code from GitHub to the newly created web app.

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

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

```
az group create --location westeurope --name myResourceGroup
```

```
    --name $webappname --resource-group myResourceGroup --sku FREE
```

az webapp  
az appservice plan create  
az webapp deployment  
az group delete

```
az webapp create  
az appservice plan create  
az webapp deployment  
az group delete
```

```
--repo-url $gitrepo --branch master --manual-integration  
git clone $gitrepo  
--plan $webappname
```

```
az webapp  
az appservice plan create  
az webapp deployment  
az group delete
```

```
--resource-group myResourceGroup
```

```
--repo-url $gitrepo --branch master --manual-integration  
git clone $gitrepo  
--plan $webappname
```

**Correct Answer:****Answer Area**

```
az group create --location westeurope --name myResourceGroup
```

```
--name $webappname --resource-group myResourceGroup --sku FREE
```

az webapp
<b>az appservice plan create</b>
az webapp deployment
az group delete

```
--name $webappname --resource-group myResourceGroup
```

<b>az webapp create</b>
az appservice plan create
az webapp deployment
az group delete

```
--repo-url $gitrepo --branch master --manual-integration  
git clone $gitrepo  
--plan $webappname
```

```
source config --name $webappname
```

az webapp
az appservice plan create
<b>az webapp deployment</b>
az group delete

```
--resource-group myResourceGroup
```

Question #4

Topic 1

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 develop a software as a service (SaaS) offering to manage photographs. Users upload photos to a web service which then stores the photos in Azure

Storage Blob storage. The storage account type is General-purpose V2.

When photos are uploaded, they must be processed to produce and save a mobile-friendly version of the image. The process to produce a mobile-friendly version of the image must start in less than one minute.

You need to design the process that starts the photo processing.

Solution: Trigger the photo processing from Blob storage events.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer: B**

You need to catch the triggered event, so move the photo processing to an Azure Function triggered from the blob upload.

Note: Azure Storage events allow applications to react to events. Common Blob storage event scenarios include image or video processing, search indexing, or any file-oriented workflow.

Events are pushed using Azure Event Grid to subscribers such as Azure Functions, Azure Logic Apps, or even to your own http listener.

However, the processing must start in less than one minute.

Note: Only storage accounts of kind StorageV2 (general purpose v2) and BlobStorage support event integration. Storage (general purpose v1) does not support integration with Event Grid.

Reference:

<https://docs.microsoft.com/en-us/azure/storage/blobs/storage-blob-event-overview>

## Question #5

## Topic 1

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 develop and deploy an Azure App Service API app to a Windows-hosted deployment slot named Development. You create additional deployment slots named Testing and Production. You enable auto swap on the Production deployment slot.

You need to ensure that scripts run and resources are available before a swap operation occurs.

Solution: Update the web.config file to include the applicationInitialization configuration element. Specify custom initialization actions to run the scripts.

Does the solution meet the goal?

A. No

B. Yes

**Correct Answer: A** 

Specify custom warm-up.

Some apps might require custom warm-up actions before the swap. The applicationInitialization configuration element in web.config lets you specify custom initialization actions. The swap operation waits for this custom warm-up to finish before swapping with the target slot. Here's a sample web.config fragment.

```
<system.webServer>
<applicationInitialization>
<add initializationPage="/" hostName="[app hostname]" />
<add initializationPage="/Home/About" hostName="[app hostname]" />
</applicationInitialization>
</system.webServer>
```

Reference:

<https://docs.microsoft.com/en-us/azure/app-service/deploy-staging-slots#troubleshoot-swaps>

## Question #6

## Topic 1

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 develop and deploy an Azure App Service API app to a Windows-hosted deployment slot named Development. You create additional deployment slots named Testing and Production. You enable auto swap on the Production deployment slot. You need to ensure that scripts run and resources are available before a swap operation occurs.

Solution: Enable auto swap for the Testing slot. Deploy the app to the Testing slot.

Does the solution meet the goal?

A. No

B. Yes

**Correct Answer:** B 

Instead update the web.config file to include the applicationInitialization configuration element. Specify custom initialization actions to run the scripts.

Note: Some apps might require custom warm-up actions before the swap. The applicationInitialization configuration element in web.config lets you specify custom initialization actions. The swap operation waits for this custom warm-up to finish before swapping with the target slot.

Here's a sample web.config fragment.

```
<system.webServer>
<applicationInitialization>
<add initializationPage="/" hostName="[app hostname]" />
<add initializationPage="/Home/About" hostName="[app hostname]" />
</applicationInitialization>
</system.webServer>
```

Reference:

<https://docs.microsoft.com/en-us/azure/app-service/deploy-staging-slots#troubleshoot-swaps>

## Question #7

## Topic 1

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 develop and deploy an Azure App Service API app to a Windows-hosted deployment slot named Development. You create additional deployment slots named Testing and Production. You enable auto swap on the Production deployment slot.

You need to ensure that scripts run and resources are available before a swap operation occurs.

Solution: Disable auto swap. Update the app with a method named statuscheck to run the scripts. Re-enable auto swap and deploy the app to the Production slot.

Does the solution meet the goal?

A. No

B. Yes

**Correct Answer:** B 

Instead update the web.config file to include the applicationInitialization configuration element. Specify custom initialization actions to run the scripts.

Note: Some apps might require custom warm-up actions before the swap. The applicationInitialization configuration element in web.config lets you specify custom initialization actions. The swap operation waits for this custom warm-up to finish before swapping with the target slot.

Here's a sample web.config fragment.

```
<system.webServer>
<applicationInitialization>
<add initializationPage="/" hostName="[app hostname]" />
<add initializationPage="/Home/About" hostName="[app hostname]" />
</applicationInitialization>
</system.webServer>
```

Reference:

<https://docs.microsoft.com/en-us/azure/app-service/deploy-staging-slots#troubleshoot-swaps>

## Question #8

## Topic 1

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 develop a software as a service (SaaS) offering to manage photographs. Users upload photos to a web service which then stores the photos in Azure.

Storage Blob storage. The storage account type is General-purpose V2.

When photos are uploaded, they must be processed to produce and save a mobile-friendly version of the image. The process to produce a mobile-friendly version of the image must start in less than one minute.

You need to design the process that starts the photo processing.

Solution: Convert the Azure Storage account to a BlockBlobStorage storage account.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer:** B 

Not necessary to convert the account, instead move photo processing to an Azure Function triggered from the blob upload..

Azure Storage events allow applications to react to events. Common Blob storage event scenarios include image or video processing, search indexing, or any file-oriented workflow.

Note: Only storage accounts of kind StorageV2 (general purpose v2) and BlobStorage support event integration. Storage (general purpose v1) does not support integration with Event Grid.

Reference:

<https://docs.microsoft.com/en-us/azure/storage/blobs/storage-blob-event-overview>

Question #9

Topic 1

**HOTSPOT -**

You are developing an Azure Web App. You configure TLS mutual authentication for the web app.

You need to validate the client certificate in the web app. To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

Property	Value
Client certificate location	<input type="checkbox"/> HTTP request header <input type="checkbox"/> Client cookie <input type="checkbox"/> HTTP message body <input type="checkbox"/> URL query string
Encoding type	<input type="checkbox"/> HTML <input type="checkbox"/> URL <input type="checkbox"/> Unicode <input type="checkbox"/> Base64

**Answer Area**

Property	Value
Client certificate location	<input checked="" type="checkbox"/> HTTP request header <input checked="" type="checkbox"/> Client cookie <input checked="" type="checkbox"/> HTTP message body <input checked="" type="checkbox"/> URL query string
Encoding type	<input checked="" type="checkbox"/> HTML <input checked="" type="checkbox"/> URL <input checked="" type="checkbox"/> Unicode <input checked="" type="checkbox"/> Base64

Accessing the client certificate from App Service.

If you are using ASP.NET and configure your app to use client certificate authentication, the certificate will be available through the `HttpRequest.ClientCertificate` property. For other application stacks, the client cert will be available in your app through a base64 encoded value in the "X-ARR-ClientCert" request header. Your application can create a certificate from this value and then use it for authentication and authorization purposes in your application.

Reference:

<https://docs.microsoft.com/en-us/azure/app-service/app-service-web-configure-tls-mutual-auth>

Question #10

Topic 1

**DRAG DROP -**

You are developing a Docker/Go using Azure App Service Web App for Containers. You plan to run the container in an App Service on Linux. You identify a Docker container image to use.

None of your current resource groups reside in a location that supports Linux. You must minimize the number of resource groups required.

You need to create the application and perform an initial deployment.

Which three Azure CLI commands should you use to develop the solution? To answer, move the appropriate commands from the list of commands to the answer area and arrange them in the correct order.

Select and Place:

**Azure CLI Commands****Answer Area**

az group create

az group update

az webapp update

az webapp create

az appservice plan create

**Azure CLI Commands****Answer Area**

az group create

az group update

Correct Answer:

az webapp update

az webapp create

az appservice plan create



You can host native Linux applications in the cloud by using Azure Web Apps. To create a Web App for Containers, you must run Azure CLI commands that create a group, then a service plan, and finally the web app itself.

Step 1: az group create -

In the Cloud Shell, create a resource group with the az group create command.

Step 2: az appservice plan create

In the Cloud Shell, create an App Service plan in the resource group with the az appservice plan create command.

Step 3: az webapp create -

In the Cloud Shell, create a web app in the myAppServicePlan App Service plan with the az webapp create command. Don't forget to replace <unique-app-name> with a unique app name, and <docker-ID> with your Docker ID.

Reference:

<https://docs.microsoft.com/mt-mt/azure/app-service/containers/quickstart-docker-go?view=sql-server-ver15>

Question #11

Topic 1

**DRAG DROP -**

Fourth Coffee has an ASP.NET Core web app that runs in Docker. The app is mapped to the www.fourthcoffee.com domain.

Fourth Coffee is migrating this application to Azure.

You need to provision an App Service Web App to host this docker image and map the custom domain to the App Service web app.

A resource group named FourthCoffeePublicWebResourceGroup has been created in the WestUS region that contains an App Service Plan named AppServiceLinuxDockerPlan.

Which order should the CLI commands be used to develop the solution? To answer, move all of the Azure CLI commands from the list of commands to the answer area and arrange them in the correct order.

Select and Place:

**Azure CLI Commands****Answer Area**

```
az webapp config container set
--docker-custom-image-name
$dockerHubContainerPath
--name $appName
--resource-group
fourthCoffeePublicWebResourceGroup
```



```
az webapp config hostname add
--webapp-name $appName
--resource-group
fourthCoffeePublicWebResourceGroup \
--hostname $fqdn
```

```
az webapp create
--name $appName
--plan AppServiceLinuxDockerPlan
--resource-group
fourthCoffeePublicWebResourceGroup
```

```
#!/bin/bash
appName="FourthCoffeePublicWeb$random"
location="WestUS"
dockerHubContainerPath="FourthCoffee/publicweb:v1"
fqdn="http://www.fourthcoffee.com">www.fourthcoffee.com
```

**Correct Answer:**

### Azure CLI Commands

```
az webapp config container set
--docker-custom-image-name
$dockerHubContainerPath
--name $appName
--resource-group
fourthCoffeePublicWebResourceGroup
```

```
az webapp config hostname add
--webapp-name $appName
--resource-group
fourthCoffeePublicWebResourceGroup \
--hostname $fqdn
```

```
az webapp create
--name $appName
--plan AppServiceLinuxDockerPlan
--resource-group
fourthCoffeePublicWebResourceGroup
```

```
#/bin/bash
appName="FourthCoffeePublicWeb$random"
location="WestUS"
dockerHubContainerPath="FourthCoffee/publicweb:v1"
fqdn="http://www.fourthcoffee.com">www.fourthcoffee.com
```

### Answer Area

```
#/bin/bash
appName="FourthCoffeePublicWeb$random"
location="WestUS"
dockerHubContainerPath="FourthCoffee/publicweb:v1"
fqdn="http://www.fourthcoffee.com">www.fourthcoffee.com
```



```
az webapp create
--name $appName
--plan AppServiceLinuxDockerPlan
--resource-group
fourthCoffeePublicWebResourceGroup
```

```
az webapp config container set
--docker-custom-image-name
$dockerHubContainerPath
--name $appName
--resource-group
fourthCoffeePublicWebResourceGroup
```

```
az webapp config hostname add
--webapp-name $appName
--resource-group
fourthCoffeePublicWebResourceGroup \
--hostname $fqdn
```

Step 1: #bin/bash -

The appName is used when the webapp-name is created in step 2.

Step 2: az webapp create -

Create a web app. In the Cloud Shell, create a web app in the myAppServicePlan App Service plan with the az webapp create command.

Step 3: az webapp config container set

In Create a web app, you specified an image on Docker Hub in the az webapp create command. This is good enough for a public image. To use a private image, you need to configure your Docker account ID and password in your Azure web app.

Step 4: az webapp config hostname add

The webapp-name is used when the webapp is created in step 2.

In the Cloud Shell, follow the az webapp create command with az webapp config container set.

Reference:

<https://docs.microsoft.com/en-us/azure/app-service/containers/tutorial-custom-docker-image> <https://docs.microsoft.com/en-us/azure/app-service/tutorial-custom-container?pivots=container-linux> <https://docs.microsoft.com/en-us/azure/app-service/scripts/cli-configure-custom-domain>

## Question #12

## DRAG DROP -

You are developing a serverless Java application on Azure. You create a new Azure Key Vault to work with secrets from a new Azure Functions application.

The application must meet the following requirements:

- Reference the Azure Key Vault without requiring any changes to the Java code.
- Dynamically add and remove instances of the Azure Functions host based on the number of incoming application events.
- Ensure that instances are perpetually warm to avoid any cold starts.
- Connect to a VNet.
- Authentication to the Azure Key Vault instance must be removed if the Azure Function application is deleted.

You need to grant the Azure Functions application access to the Azure Key Vault.

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
Create a user-assigned managed identity for the application.	
Create the Azure Functions app with a Premium plan type.	
Create an access policy in Azure Key Vault for the application identity.	▶
Create an SSL certification in Azure Key Vault for the application identity.	◀
Create the Azure Functions app with an App Service plan type.	
Create the Azure Functions app with a Consumption plan type.	
Create a system-assigned managed identity for the application.	

Actions	Answer Area
Create a user-assigned managed identity for the application.	Create the Azure Functions app with a Consumption plan type.
Create the Azure Functions app with a Premium plan type.	Create a user-assigned managed identity for the application.
Create an access policy in Azure Key Vault for the application identity.	Create an access policy in Azure Key Vault for the application identity.
<b>Correct Answer:</b> Create an SSL certification in Azure Key Vault for the application identity.	
Create the Azure Functions app with an App Service plan type.	
Create the Azure Functions app with a Consumption plan type.	
Create a system-assigned managed identity for the application.	

Step 1: Create the Azure Functions app with a Consumption plan type.

Use the Consumption plan for serverless.

Step 2: Create a system-assigned managed identity for the application.

Create a system-assigned managed identity for your application.

Key Vault references currently only support system-assigned managed identities. User-assigned identities cannot be used.

Step 3: Create an access policy in Key Vault for the application identity.

Create an access policy in Key Vault for the application identity you created earlier. Enable the "Get" secret permission on this policy. Do not

configure the "authorized application" or applicationId settings, as this is not compatible with a managed identity.

Reference:

<https://docs.microsoft.com/en-us/azure/app-service/app-service-key-vault-references>

Question #13

Topic 1

You develop a website. You plan to host the website in Azure. You expect the website to experience high traffic volumes after it is published.

You must ensure that the website remains available and responsive while minimizing cost.

You need to deploy the website.

What should you do?

- A. Deploy the website to a virtual machine. Configure the virtual machine to automatically scale when the CPU load is high.
- B. Deploy the website to an App Service that uses the Shared service tier. Configure the App Service plan to automatically scale when the CPU load is high.
- C. Deploy the website to a virtual machine. Configure a Scale Set to increase the virtual machine instance count when the CPU load is high.
- D. Deploy the website to an App Service that uses the Standard service tier. Configure the App Service plan to automatically scale when the CPU load is high.

**Correct Answer: D**

Windows Azure Web Sites (WAWS) offers 3 modes: Standard, Free, and Shared.

Standard mode carries an enterprise-grade SLA (Service Level Agreement) of 99.9% monthly, even for sites with just one instance.

Standard mode runs on dedicated instances, making it different from the other ways to buy Windows Azure Web Sites.

Incorrect Answers:

B: Shared and Free modes do not offer the scaling flexibility of Standard, and they have some important limits.

Shared mode, just as the name states, also uses shared Compute resources, and also has a CPU limit. So, while neither Free nor Shared is likely to be the best choice for your production environment due to these limits.

## Question #14

## HOTSPOT -

A company is developing a Java web app. The web app code is hosted in a GitHub repository located at <https://github.com/Contoso/webapp>. The web app must be evaluated before it is moved to production. You must deploy the initial code release to a deployment slot named staging.

You need to create the web app and deploy the code.

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

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

```
gitrepo=https://github.com/Contoso/webapp  
webappname=businesswebapp  
resourcegroupname=BusinessAppResourceGroup
```

- |    |  |   |
|----|--|---|
| az | <input type="checkbox"/> group<br><input type="checkbox"/> webapp<br><input type="checkbox"/> appservice plan<br><input checked="" type="checkbox"/> webapp deployment slot<br><input type="checkbox"/> webapp deployment source | create --location centralus --name \$resourcegroupname  |
| az | <input type="checkbox"/> group<br><input type="checkbox"/> webapp<br><input type="checkbox"/> appservice plan<br><input checked="" type="checkbox"/> webapp deployment slot<br><input type="checkbox"/> webapp deployment source | create --name \$webappname --resource-group \$resourcegroupname --sku S3  |
| az | <input type="checkbox"/> group<br><input type="checkbox"/> webapp<br><input type="checkbox"/> appservice plan<br><input checked="" type="checkbox"/> webapp deployment slot<br><input type="checkbox"/> webapp deployment source | create --name \$webappname --resource-group \$resourcegroupname --plan \$webappname   |
| az | <input type="checkbox"/> group<br><input type="checkbox"/> webapp<br><input type="checkbox"/> appservice plan<br><input checked="" type="checkbox"/> webapp deployment slot<br><input type="checkbox"/> webapp deployment source | create --name \$webappname --resource-group \$resourcegroupname --slot staging  |
| az | <input type="checkbox"/> group<br><input type="checkbox"/> webapp<br><input type="checkbox"/> appservice plan<br><input checked="" type="checkbox"/> webapp deployment slot<br><input type="checkbox"/> webapp deployment source | config --name \$webappname --resource-group \$resourcegroupname \<br>--slot staging --repo-url \$gitrepo --branch master --manual-integration |

**Correct Answer:****Answer Area**

gitrepo=https://github.com/Contoso/webapp							
webappname=businesswebapp							
resourcegroupname=BusinessAppResourceGroup							
az	<div style="border: 1px solid black; padding: 5px; width: fit-content;"> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>group</td></tr> <tr><td>webapp</td></tr> <tr><td>appservice plan</td></tr> <tr><td>webapp deployment slot</td></tr> <tr><td>webapp deployment source</td></tr> </table> </div>	group	webapp	appservice plan	webapp deployment slot	webapp deployment source	create --location centralus --name \$resourcegroupname
group							
webapp							
appservice plan							
webapp deployment slot							
webapp deployment source							
az	<div style="border: 1px solid black; padding: 5px; width: fit-content;"> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>group</td></tr> <tr><td>webapp</td></tr> <tr><td>appservice plan</td></tr> <tr><td>webapp deployment slot</td></tr> <tr><td>webapp deployment source</td></tr> </table> </div>	group	webapp	appservice plan	webapp deployment slot	webapp deployment source	create --name \$webappname --resource-group \$resourcegroupname --sku S3
group							
webapp							
appservice plan							
webapp deployment slot							
webapp deployment source							
az	<div style="border: 1px solid black; padding: 5px; width: fit-content;"> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>group</td></tr> <tr><td>webapp</td></tr> <tr><td>appservice plan</td></tr> <tr><td>webapp deployment slot</td></tr> <tr><td>webapp deployment source</td></tr> </table> </div>	group	webapp	appservice plan	webapp deployment slot	webapp deployment source	create --name \$webappname --resource-group \$resourcegroupname --plan \$webappname
group							
webapp							
appservice plan							
webapp deployment slot							
webapp deployment source							
az	<div style="border: 1px solid black; padding: 5px; width: fit-content;"> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>group</td></tr> <tr><td>webapp</td></tr> <tr><td>appservice plan</td></tr> <tr><td>webapp deployment slot</td></tr> <tr><td>webapp deployment source</td></tr> </table> </div>	group	webapp	appservice plan	webapp deployment slot	webapp deployment source	create --name \$webappname --resource-group \$resourcegroupname --slot staging
group							
webapp							
appservice plan							
webapp deployment slot							
webapp deployment source							
az	<div style="border: 1px solid black; padding: 5px; width: fit-content;"> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>group</td></tr> <tr><td>webapp</td></tr> <tr><td>appservice plan</td></tr> <tr><td>webapp deployment slot</td></tr> <tr><td>webapp deployment source</td></tr> </table> </div>	group	webapp	appservice plan	webapp deployment slot	webapp deployment source	config --name \$webappname --resource-group \$resourcegroupname \ --slot staging --repo-url \$gitrepo --branch master --manual-integration
group							
webapp							
appservice plan							
webapp deployment slot							
webapp deployment source							

Box 1: group -

# Create a resource group.

az group create --location westeurope --name myResourceGroup

Box 2: appservice plan -

# Create an App Service plan in STANDARD tier (minimum required by deployment slots). az appservice plan create --name \$webappname --resource-group myResourceGroup --sku S1

Box 3: webapp -

# Create a web app.

az webapp create --name \$webappname --resource-group myResourceGroup \  
--plan \$webappname

Box 4: webapp deployment slot -

#Create a deployment slot with the name "staging".

az webapp deployment slot create --name \$webappname --resource-group myResourceGroup \  
--slot staging

Box 5: webapp deployment source -

# Deploy sample code to "staging" slot from GitHub.

az webapp deployment source config --name \$webappname --resource-group myResourceGroup \  
--slot staging --repo-url \$gitrepo --branch master --manual-integration

Reference:

<https://docs.microsoft.com/en-us/azure/app-service/scripts/cli-deploy-staging-environment>



## Question #15

## HOTSPOT -

You have a web service that is used to pay for food deliveries. The web service uses Azure Cosmos DB as the data store.

You plan to add a new feature that allows users to set a tip amount. The new feature requires that a property named tip on the document in Cosmos DB must be present and contain a numeric value.

There are many existing websites and mobile apps that use the web service that will not be updated to set the tip property for some time.

How should you complete the trigger?

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

```
function ensureTip() {
    var r = 
        .value();
        .readDocument('item');
        getContext().getRequest();
        getContext().getResponse();

    var i = r.getBody();

    if (!("tip" in i)) {
        if (request.getValue("tip") === null) {
            if (isNaN(i["tip"]) || i["tip"] === null) {
                if (typeof __.pluck("tip") === 'number') {
                    i["tip"] = 0;
                }
            }
        }
        r.setBody(i);
        r.setValue(i);
        __.upsertDocument(i);
        __.replaceDocument(i);
    }
}
```

**Answer Area**

```
function ensureTip() {
    var r = 
        .value();
        .readDocument('item');
        getContext().getRequest();
        getContext().getResponse();

    var i = r.getBody();

    if (!("tip" in i)) {
        if (request.getValue("tip") === null) {
            if (isNaN(i["tip"]) || i["tip"] === null) {
                if (typeof __.pluck("tip") === 'number') {
                    i["tip"] = 0;
                }
            }
        }
        r.setBody(i);
        r.setValue(i);
        __.upsertDocument(i);
        __.replaceDocument(i);
    }
}
```

Correct Answer:

## Question #16

Topic 1

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 develop an HTTP triggered Azure Function app to process Azure Storage blob data. The app is triggered using an output binding on the blob.

The app continues to time out after four minutes. The app must process the blob data.

You need to ensure the app does not time out and processes the blob data.

Solution: Use the Durable Function async pattern to process the blob data.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer: B** 

Instead pass the HTTP trigger payload into an Azure Service Bus queue to be processed by a queue trigger function and return an immediate HTTP success response.

Note: Large, long-running functions can cause unexpected timeout issues. General best practices include:

Whenever possible, refactor large functions into smaller function sets that work together and return responses fast. For example, a webhook or HTTP trigger function might require an acknowledgment response within a certain time limit; it's common for webhooks to require an immediate response. You can pass the

HTTP trigger payload into a queue to be processed by a queue trigger function. This approach lets you defer the actual work and return an immediate response.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-functions/functions-best-practices>

## Question #17

Topic 1

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 develop an HTTP triggered Azure Function app to process Azure Storage blob data. The app is triggered using an output binding on the blob. The app continues to time out after four minutes. The app must process the blob data.

You need to ensure the app does not time out and processes the blob data.

Solution: Pass the HTTP trigger payload into an Azure Service Bus queue to be processed by a queue trigger function and return an immediate HTTP success response.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer: A**

Large, long-running functions can cause unexpected timeout issues. General best practices include:

Whenever possible, refactor large functions into smaller function sets that work together and return responses fast. For example, a webhook or HTTP trigger function might require an acknowledgment response within a certain time limit; it's common for webhooks to require an immediate response. You can pass the

HTTP trigger payload into a queue to be processed by a queue trigger function. This approach lets you defer the actual work and return an immediate response.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-functions/functions-best-practices>

## Question #18

Topic 1

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 develop an HTTP triggered Azure Function app to process Azure Storage blob data. The app is triggered using an output binding on the blob.

The app continues to time out after four minutes. The app must process the blob data.

You need to ensure the app does not time out and processes the blob data.

Solution: Configure the app to use an App Service hosting plan and enable the Always On setting.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer: B**

Instead pass the HTTP trigger payload into an Azure Service Bus queue to be processed by a queue trigger function and return an immediate HTTP success response.

Note: Large, long-running functions can cause unexpected timeout issues. General best practices include:

Whenever possible, refactor large functions into smaller function sets that work together and return responses fast. For example, a webhook or HTTP trigger function might require an acknowledgment response within a certain time limit; it's common for webhooks to require an immediate response. You can pass the

HTTP trigger payload into a queue to be processed by a queue trigger function. This approach lets you defer the actual work and return an immediate response.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-functions/functions-best-practices>

## Question #19

Topic 1

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 develop a software as a service (SaaS) offering to manage photographs. Users upload photos to a web service which then stores the photos in Azure.

Storage Blob storage. The storage account type is General-purpose V2.

When photos are uploaded, they must be processed to produce and save a mobile-friendly version of the image. The process to produce a mobile-friendly version of the image must start in less than one minute.

You need to design the process that starts the photo processing.

Solution: Move photo processing to an Azure Function triggered from the blob upload.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer: A**

Azure Storage events allow applications to react to events. Common Blob storage event scenarios include image or video processing, search indexing, or any file-oriented workflow.

Events are pushed using Azure Event Grid to subscribers such as Azure Functions, Azure Logic Apps, or even to your own http listener.

Note: Only storage accounts of kind StorageV2 (general purpose v2) and BlobStorage support event integration. Storage (general purpose v1) does not support integration with Event Grid.

Reference:

<https://docs.microsoft.com/en-us/azure/storage/blobs/storage-blob-event-overview>

## Question #20

Topic 1

You are developing an application that uses Azure Blob storage.

The application must read the transaction logs of all the changes that occur to the blobs and the blob metadata in the storage account for auditing purposes. The changes must be in the order in which they occurred, include only create, update, delete, and copy operations and be retained for compliance reasons.

You need to process the transaction logs asynchronously.

What should you do?

- A. Process all Azure Blob storage events by using Azure Event Grid with a subscriber Azure Function app.
- B. Enable the change feed on the storage account and process all changes for available events.
- C. Process all Azure Storage Analytics logs for successful blob events.
- D. Use the Azure Monitor HTTP Data Collector API and scan the request body for successful blob events.

**Correct Answer: B**

Change feed support in Azure Blob Storage

The purpose of the change feed is to provide transaction logs of all the changes that occur to the blobs and the blob metadata in your storage account. The change feed provides ordered, guaranteed, durable, immutable, read-only log of these changes. Client applications can read these logs at any time, either in streaming or in batch mode. The change feed enables you to build efficient and scalable solutions that process change events that occur in your Blob Storage account at a low cost.

Reference:

<https://docs.microsoft.com/en-us/azure/storage/blobs/storage-blob-change-feed>

Question #21

Topic 1

**DRAG DROP -**

You plan to create a Docker image that runs an ASP.NET Core application named ContosoApp. You have a setup script named setupScript.ps1 and a series of application files including ContosoApp.dll.

You need to create a Dockerfile document that meets the following requirements:

- Call setupScripts.ps1 when the container is built.
- Run ContosoApp.dll when the container starts.

The Dockerfile document must be created in the same folder where ContosoApp.dll and setupScript.ps1 are stored.

Which five commands should you use to develop the solution? To answer, move the appropriate commands from the list of commands to the answer area and arrange them in the correct order.

Select and Place:

**Commands**

```
FROM microsoft/aspnetcore:latest  
WORKDIR /apps/ContosoApp  
CMD ["dotnet", "ContosoApp.dll"]  
COPY ./ .  
RUN powershell ./setupScript.ps1
```

**Answer Area**

Correct Answer:

**Commands**

```
FROM microsoft/aspnetcore:latest  
WORKDIR /apps/ContosoApp  
CMD ["dotnet", "ContosoApp.dll"]  
COPY ./ .  
RUN powershell ./setupScript.ps1
```

**Answer Area**

```
CMD ["dotnet", "ContosoApp.dll"]  
FROM microsoft/aspnetcore:latest  
WORKDIR /apps/ContosoApp  
COPY ./ .  
RUN powershell ./setupScript.ps1
```

Box 1: CMD [...]

Cmd starts a new instance of the command interpreter, Cmd.exe.

Syntax: CMD <string>

Specifies the command you want to carry out.

Box 2: FROM microsoft/aspnetcore-build:latest

Box 3: WORKDIR /apps/ContosoApp -

Box 4: COPY ./ .

Box 5: RUN powershell ./setupScript.ps1

Question #22

Topic 1

You are developing an Azure Function App that processes images that are uploaded to an Azure Blob container. Images must be processed as quickly as possible after they are uploaded, and the solution must minimize latency. You create code to process images when the Function App is triggered. You need to configure the Function App. What should you do?

- A. Use an App Service plan. Configure the Function App to use an Azure Blob Storage input trigger.
- B. Use a Consumption plan. Configure the Function App to use an Azure Blob Storage trigger.
- C. Use a Consumption plan. Configure the Function App to use a Timer trigger.
- D. Use an App Service plan. Configure the Function App to use an Azure Blob Storage trigger.
- E. Use a Consumption plan. Configure the Function App to use an Azure Blob Storage input trigger.

**Correct Answer:** *B* 

The Blob storage trigger starts a function when a new or updated blob is detected. The blob contents are provided as input to the function.

The Consumption plan limits a function app on one virtual machine (VM) to 1.5 GB of memory.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-functions/functions-bindings-storage-blob-trigger>

## Question #23

## HOTSPOT -

You are configuring a new development environment for a Java application.

The environment requires a Virtual Machine Scale Set (VMSS), several storage accounts, and networking components.

The VMSS must not be created until the storage accounts have been successfully created and an associated load balancer and virtual network is configured.

How should you complete the Azure Resource Manager template? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

```
{  
    . . .  
    "resources": [  
        {  
            "apiVersion": "2016-01-01",  
            "type": "Microsoft.Storage/storageAccounts",  
            "name": "[concat(          (), 'storage', uniqueString(resourceGroup().id))]",  
            "copy",  
            "copyIndex",  
            "priority",  
            "dependsOn"  
            "location": "[resourceGroup().location]",  
            . . .  
            "sku": {  
                "name": "Standard_LRS"  
            },  
            "kind": "Storage",  
            "properties": {},  
            "copy",  
            "copyIndex",  
            "priority",  
            "dependsOn"  
            "name": "storagesetup",  
            "count": 3  
        }  
    ],  
    {  
        "apiVersion": "2015-06-15",  
        "type": "Microsoft.Compute/virtualMachines",  
        "name": "[concat('VM', uniqueString(resourceGroup().id))]",  
        "copy",  
        "copyIndex",  
        "priority",  
        "dependsOn"  
        "[variables('loadBalancerName')]",  
        "[variables('virtualNetworkName')]",  
        "storagesetup",  
        [ ,  
        . . .  
    }  
],  
"outputs": {}  
}
```

**Correct Answer:****Answer Area**

```
{
  ...
  "resources": [
    {
      "apiVersion": "2016-01-01",
      "type": "Microsoft.Storage/storageAccounts",
      "name": "[concat('storage', uniqueString(resourceGroup().id))]",
      "copy": [
        "copy"
        "copyIndex"
        "priority"
        "dependsOn"
      ],
      "location": "[resourceGroup().location]",
      ...
      "sku": {
        "name": "Standard_LRS"
      },
      "kind": "Storage",
      "properties": {},
      "dependsOn": [
        "copy"
        "copyIndex"
        "priority"
        "dependsOn"
      ],
      "name": "storagesetup",
      "count": 3
    },
    {
      "apiVersion": "2015-06-15",
      "type": "Microsoft.Compute/virtualMachines",
      "name": "[concat('VM', uniqueString(resourceGroup().id))]",
      "copy": [
        "copy"
        "copyIndex"
        "priority"
        "dependsOn"
      ],
      "dependsOn": [
        "[variables('loadBalancerName')]",
        "[variables('virtualNetworkName')]",
        "storagesetup",
      ],
      ...
    }
  ],
  "outputs": {}
}
```

**Box 1: copyIndex -**

Notice that the name of each resource includes the `copyIndex()` function, which returns the current iteration in the loop. `copyIndex()` is zero-based.

**Box 2: copy -**

By adding the `copy` element to the resources section of your template, you can dynamically set the number of resources to deploy.

**Box 3: dependsOn -**

Example:

```
"type": "Microsoft.Compute/virtualMachineScaleSets",
"apiVersion": "2020-06-01",
"name": "[variables('namingInfix')]",
"location": "[parameters('location')]",
"sku": {
  "name": "[parameters('vmSku')]",
  "tier": "Standard",
  "capacity": "[parameters('instanceCount')]"
},
"dependsOn": [
  "[resourceld('Microsoft.Network/loadBalancers', variables('loadBalancerName'))]"
]
```

```
[resourceId('Microsoft.Network/virtualNetworks', variables('virtualNetworkName'))]"  
],
```

Reference:

<https://docs.microsoft.com/en-us/azure/azure-resource-manager/templates/copy-resources> <https://docs.microsoft.com/en-us/azure/virtual-machine-scale-sets/quick-create-template-windows>

Question #24

Topic 1

**HOTSPOT -**

You are developing an Azure Function App by using Visual Studio. The app will process orders input by an Azure Web App. The web app places the order information into Azure Queue Storage.

You need to review the Azure Function App code shown below.

```
public static class OrderProcessor
{
    [FunctionName("ProcessOrders")]
    public static void ProcessOrders([QueueTrigger("incoming-orders")]CloudQueueMessage myQueueItem, [Table("Orders")]ICollector<Order> tableBindings, TraceWriter log)
    {
        log.Info($"Processing Order: {myQueueItem.Id}");
        log.Info($"Queue Insertion Time: {myQueueItem.InsertionTime}");
        log.Info($"Queue Expiration Time: {myQueueItem.ExpirationTime}");
        tableBindings.Add(JsonConvert.DeserializeObject<Order>(myQueueItem.AsString));
    }
    [FunctionName("ProcessOrders-Poison")]
    public static void ProcessFailedOrders([QueueTrigger("incoming-orders-poison")]CloudQueueMessage myQueueItem, TraceWriter log)
    {
        log.Error($"Failed to process order: {myQueueItem.AsString}");
        ...
    }
}
```

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

Yes	No
<input type="radio"/>	<input type="radio"/>

The code will log the time that the order was processed from the queue.

When the ProcessOrders function fails, the function will retry up to five times for a given order, including the first try.

When there are multiple orders in the queue, a batch of orders will be retrieved from the queue and the ProcessOrders function will run multiple instances concurrently to process the orders.

The ProcessOrders function will output the order to an Orders table in Azure Table Storage.

**Correct Answer:****Answer Area**

Yes	No
<input type="radio"/>	<input checked="" type="radio"/>
<input checked="" type="radio"/>	<input type="radio"/>
<input checked="" type="radio"/>	<input type="radio"/>
<input checked="" type="radio"/>	<input type="radio"/>

The code will log the time that the order was processed from the queue.

When the ProcessOrders function fails, the function will retry up to five times for a given order, including the first try.

When there are multiple orders in the queue, a batch of orders will be retrieved from the queue and the ProcessOrders function will run multiple instances concurrently to process the orders.

The ProcessOrders function will output the order to an Orders table in Azure Table Storage.

Box 1: No -

ExpirationTime - The time that the message expires.

InsertionTime - The time that the message was added to the queue.

Box 2: Yes -

maxDequeueCount - The number of times to try processing a message before moving it to the poison queue. Default value is 5.

Box 3: Yes -

When there are multiple queue messages waiting, the queue trigger retrieves a batch of messages and invokes function instances concurrently to process them.

By default, the batch size is 16. When the number being processed gets down to 8, the runtime gets another batch and starts processing those messages. So the maximum number of concurrent messages being processed per function on one virtual machine (VM) is 24.

Box 4: Yes -

Reference:

<https://docs.microsoft.com/en-us/azure/azure-functions/functions-bindings-storage-queue>

Question #25

Topic 1

#### DRAG DROP -

You are developing a solution for a hospital to support the following use cases:

- The most recent patient status details must be retrieved even if multiple users in different locations have updated the patient record.
- Patient health monitoring data retrieved must be the current version or the prior version.
- After a patient is discharged and all charges have been assessed, the patient billing record contains the final charges.

You provision a Cosmos DB NoSQL database and set the default consistency level for the database account to Strong. You set the value for Indexing Mode to

Consistent.

You need to minimize latency and any impact to the availability of the solution. You must override the default consistency level at the query level to meet the required consistency guarantees for the scenarios.

Which consistency levels should you implement? To answer, drag the appropriate consistency levels to the correct requirements. Each consistency level 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:

Consistency levels	
Strong	Bounded Staleness
Consistent Prefix	Eventual

#### Answer Area

Return the most recent patient status.

Consistency level

Return health monitoring data that is no less than one version behind.

Consistency level

After patient is discharged and all charges are assessed, retrieve the correct billing data with the final charges.

Consistency level

#### Correct Answer:

Consistency levels	
Strong	Bounded Staleness
Consistent Prefix	Eventual

#### Answer Area

Return the most recent patient status.

Strong

Return health monitoring data that is no less than one version behind.

Bounded Staleness

After patient is discharged and all charges are assessed, retrieve the correct billing data with the final charges.

Eventual

Box 1: Strong -

Strong: Strong consistency offers a linearizability guarantee. The reads are guaranteed to return the most recent committed version of an item. A client never sees an uncommitted or partial write. Users are always guaranteed to read the latest committed write.

Box 2: Bounded staleness -

Bounded staleness: The reads are guaranteed to honor the consistent-prefix guarantee. The reads might lag behind writes by at most "K" versions (that is

"updates") of an item or by "t" time interval. When you choose bounded staleness, the "staleness" can be configured in two ways:

The number of versions (K) of the item

The time interval (t) by which the reads might lag behind the writes

Box 3: Eventual -

Eventual: There's no ordering guarantee for reads. In the absence of any further writes, the replicas eventually converge.

Incorrect Answers:

Consistent prefix: Updates that are returned contain some prefix of all the updates, with no gaps. Consistent prefix guarantees that reads never see out-of-order writes.

Reference:

<https://docs.microsoft.com/en-us/azure/cosmos-db/consistency-levels>

Question #26

Topic 1

**HOTSPOT -**

You are configuring a development environment for your team. You deploy the latest Visual Studio image from the Azure Marketplace to your Azure subscription.

The development environment requires several software development kits (SDKs) and third-party components to support application development across the organization. You install and customize the deployed virtual machine (VM) for your development team. The customized VM must be saved to allow provisioning of a new team member development environment.

You need to save the customized VM for future provisioning.

Which tools or services should you use? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

Action	Tool or service
Generalize the VM.	Azure PowerShell Visual Studio command prompt Azure Migrate Azure Backup
Store images.	Azure Blob Storage Azure Data Lake Storage Azure File Storage Azure Table Storage

Correct Answer:

**Answer Area**

Action	Tool or service
Generalize the VM.	Azure PowerShell Visual Studio command prompt Azure Migrate Azure Backup
Store images.	Azure Blob Storage Azure Data Lake Storage Azure File Storage Azure Table Storage

Box 1: Azure Powershell -

Creating an image directly from the VM ensures that the image includes all of the disks associated with the VM, including the OS disk and any data disks.

Before you begin, make sure that you have the latest version of the Azure PowerShell module.

You use Sysprep to generalize the virtual machine, then use Azure PowerShell to create the image.

Box 2: Azure Blob Storage -

You can store images in Azure Blob Storage.

Reference:

<https://docs.microsoft.com/en-us/azure/virtual-machines/windows/capture-image-resource#create-an-image-of-a-vm-using-powershell>

Question #27

Topic 1

You are preparing to deploy a website to an Azure Web App from a GitHub repository. The website includes static content generated by a script. You plan to use the Azure Web App continuous deployment feature. You need to run the static generation script before the website starts serving traffic. What are two possible ways to achieve this goal? Each correct answer presents a complete solution.

NOTE: Each correct selection is worth one point.

- A. Add the path to the static content generation tool to WEBSITE\_RUN\_FROM\_PACKAGE setting in the host.json file.
- B. Add a PreBuild target in the websites csproj project file that runs the static content generation script.
- C. Create a file named run.cmd in the folder /run that calls a script which generates the static content and deploys the website.
- D. Create a file named .deployment in the root of the repository that calls a script which generates the static content and deploys the website.

**Correct Answer:** AD 

A: In Azure, you can run your functions directly from a deployment package file in your function app. The other option is to deploy your files in the d:\home\site  
\wwwroot directory of your function app (see A above).

To enable your function app to run from a package, you just add a WEBSITE\_RUN\_FROM\_PACKAGE setting to your function app settings.

Note: The host.json metadata file contains global configuration options that affect all functions for a function app.

D: To customize your deployment, include a .deployment file in the repository root.

You just need to add a file to the root of your repository with the name .deployment and the content:

[config]

command = YOUR COMMAND TO RUN FOR DEPLOYMENT

this command can be just running a script (batch file) that has all that is required for your deployment, like copying files from the repository to the web root directory for example.

Reference:

<https://github.com/projectkudu/kudu/wiki/Custom-Deployment-Script> <https://docs.microsoft.com/bs-latn-ba/azure/azure-functions/run-functions-from-deployment-package>

Question #28

Topic 1

**DRAG DROP -**

You are developing an application to use Azure Blob storage. You have configured Azure Blob storage to include change feeds.

A copy of your storage account must be created in another region. Data must be copied from the current storage account to the new storage account directly between the storage servers.

You need to create a copy of the storage account in another region and copy the data.

In which order should you perform the actions? To answer, move all actions from the list of actions to the answer area and arrange them in the correct order.

Select and Place:

**Actions**

Use AZCopy to copy the data to the new storage account.

Deploy the template to create a new storage account in the target region.

Export a Resource Manager template.

Create a new template deployment.

Modify the template by changing the storage account name and region.

**Answer Area**

Correct Answer:

**Actions**

Use AZCopy to copy the data to the new storage account.

Deploy the template to create a new storage account in the target region.

Export a Resource Manager template.

Create a new template deployment.

Modify the template by changing the storage account name and region.

**Answer Area**

Create a new template deployment.

Export a Resource Manager template.

Modify the template by changing the storage account name and region.

Deploy the template to create a new storage account in the target region.

Use AZCopy to copy the data to the new storage account.



To move a storage account, create a copy of your storage account in another region. Then, move your data to that account by using AzCopy, or another tool of your choice.

The steps are:

- ☛ Export a template.
- ☛ Modify the template by adding the target region and storage account name.
- ☛ Deploy the template to create the new storage account.
- ☛ Configure the new storage account.
- ☛ Move data to the new storage account.
- ☛ Delete the resources in the source region.

Note: You must enable the change feed on your storage account to begin capturing and recording changes. You can enable and disable changes by using Azure

Resource Manager templates on Portal or Powershell.

Reference:

<https://docs.microsoft.com/en-us/azure/storage/common/storage-account-move> <https://docs.microsoft.com/en-us/azure/storage/blobs/storage-blob-change-feed>

Question #29

Topic 1

**DRAG DROP -**

You are preparing to deploy an Azure virtual machine (VM)-based application.

The VMs that run the application have the following requirements:

- When a VM is provisioned the firewall must be automatically configured before it can access Azure resources.
- Supporting services must be installed by using an Azure PowerShell script that is stored in Azure Storage.

You need to ensure that the requirements are met.

Which features should you use? To answer, drag the appropriate features to the correct requirements. Each feature 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:

Features	Answer Area	Requirement	Feature
Run Command		Firewall configuration	
Serial console		Supporting services script	
Hybrid Runbook Worker			
Custom Script Extension			

Correct Answer:

Features	Answer Area	Requirement	Feature
		Firewall configuration	Run Command
Serial console		Supporting services script	Hybrid Runbook Worker

Reference:

<https://docs.microsoft.com/en-us/azure/automation/automation-hybrid-runbook-worker> <https://docs.microsoft.com/en-us/azure/virtual-machines/windows/run-command>

## Question #30

## HOTSPOT -

A company is developing a Node.js web app. The web app code is hosted in a GitHub repository located at <https://github.com/TailSpinToys/webapp>.

The web app must be reviewed before it is moved to production. You must deploy the initial code release to a deployment slot named review.

You need to create the web app and deploy the code.

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

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

```
$gitrepo="https://github.com/TailSpinToys/webapp"
$webappname="TailSpinToysWeb"
$location="WestUS2"
```

-Name myResourceGroup -Location \$location
New-AzWebAppSlot
New-AzWebApp
New-AzAppServicePlan
New-AzResourceGroup

-Name \$webappname -Location \$location -ResourceGroupName myResourceGroup -Tier Standard
New-AzWebAppSlot
New-AzWebApp
New-AzAppServicePlan
New-AzResourceGroup

-Name \$webappname -Location \$location -AppServicePlan \$webappname -ResourceGroupName myResourceGroup
New-AzWebAppSlot
New-AzWebApp
New-AzAppServicePlan
New-AzResourceGroup

-Name \$webappname -ResourceGroupName myResourceGroup -Slot review
New-AzWebAppSlot
New-AzWebApp
New-AzAppServicePlan
New-AzResourceGroup

```
$PropertiesObject = @{repoUrl = "$gitrepo";branch = "master";}
Set-AzResource -PropertyObject $PropertiesObject -ResourceGroupName myResourceGroup -ResourceType
Microsoft.Web/sites/slots/sourcecontrols -ResourceName $webappname/review/web -ApiVersion 2015-08-01 -Force
Switch-AzWebAppSlot -Name $webappname -ResourceGroupName myResourceGroup
-SourceSlotName review -DestinationSlotName production
```

**Correct Answer:****Answer Area**

```
$gitrepo="https://github.com/TailSpinToys/webapp"
$webappname="TailSpinToysWeb"
$location="WestUS2"
```

-Name myResourceGroup -Location \$location
New-AzWebAppSlot
New-AzWebApp
New-AzAppServicePlan
New-AzResourceGroup

-Name \$webappname -Location \$location -ResourceGroupName myResourceGroup -Tier Standard
New-AzWebAppSlot
New-AzWebApp
New-AzAppServicePlan
New-AzResourceGroup

-Name \$webappname -Location \$location -AppServicePlan \$webappname -ResourceGroupName myResourceGroup
New-AzWebAppSlot
New-AzWebApp
New-AzAppServicePlan
New-AzResourceGroup

-Name \$webappname -ResourceGroupName myResourceGroup -Slot review
New-AzWebAppSlot
New-AzWebApp
New-AzAppServicePlan
New-AzResourceGroup

```
$PropertiesObject = @{repoUrl = "$gitrepo";branch = "master";}
Set-AzResource -PropertyObject $PropertiesObject -ResourceGroupName myResourceGroup -ResourceType
Microsoft.Web/sites/slots/sourcecontrols -ResourceName $webappname/review/web -ApiVersion 2015-08-01 -Force
Switch-AzWebAppSlot -Name $webappname -ResourceGroupName myResourceGroup
-SourceSlotName review -DestinationSlotName production
```

Box 1: New-AzResourceGroup -

The New-AzResourceGroup cmdlet creates an Azure resource group.

**Box 2: New-AzAppServicePlan -**

The New-AzAppServicePlan cmdlet creates an Azure App Service plan in a given location

**Box 3: New-AzWebApp -**

The New-AzWebApp cmdlet creates an Azure Web App in a given a resource group

**Box 4: New-AzWebAppSlot -**

The New-AzWebAppSlot cmdlet creates an Azure Web App slot.

Reference:

<https://docs.microsoft.com/en-us/powershell/module/az.resources/new-azresourcegroup?view=azps-2.3.2> <https://docs.microsoft.com/en-us/powershell/module/az.websites/new-azappserviceplan?view=azps-2.3.2> <https://docs.microsoft.com/en-us/powershell/module/az.websites/new-azwebapp?view=azps-2.3.2> <https://docs.microsoft.com/en-us/powershell/module/az.websites/new-azwebappslot?view=azps-2.3.2>

Question #31

**HOTSPOT -**

You are developing an application that needs access to an Azure virtual machine (VM).

The access lifecycle for the application must be associated with the VM service instance.

You need to enable managed identity for the VM.

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

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

\$vm = Get-AzVM -ResourceGroupName "ContosoRG" -Name "ContosoVM"

Update-AzVM -ResourceGroupName "ContosoRG" -VM \$vm

-AssignIdentity:	▼
-IdentityId:	▼

\$SystemAssigned	▼
\$UserAssigned	▼

Correct Answer:

**Answer Area**

\$vm = Get-AzVM -ResourceGroupName "ContosoRG" -Name "ContosoVM"

Update-AzVM -ResourceGroupName "ContosoRG" -VM \$vm

-AssignIdentity:	▼
-IdentityId:	▼

\$SystemAssigned	▼
\$UserAssigned	▼

Box 1: -IdentityType -

Enable system-assigned managed identity on an existing Azure VM:

To enable a system-assigned managed identity, use the -IdentityType switch on the Update-AzVM cmdlet (see below).

Box 2: \$SystemAssigned -

\$vm = Get-AzVM -ResourceGroupName myResourceGroup -Name myVM

Update-AzVM -ResourceGroupName myResourceGroup -VM \$vm -IdentityType SystemAssigned

Reference:

<https://docs.microsoft.com/en-us/azure/active-directory/managed-identities-azure-resources/qs-configure-powershell-windows-vm>

Question #32

Topic 1

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 develop a software as a service (SaaS) offering to manage photographs. Users upload photos to a web service which then stores the photos in Azure

Storage Blob storage. The storage account type is General-purpose V2.

When photos are uploaded, they must be processed to produce and save a mobile-friendly version of the image. The process to produce a mobile-friendly version of the image must start in less than one minute.

You need to design the process that starts the photo processing.

Solution: Create an Azure Function app that uses the Consumption hosting model and that is triggered from the blob upload.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer: A** 

In the Consumption hosting plan, resources are added dynamically as required by your functions.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-functions/functions-create-storage-blob-triggered-function>

## Question #33

## Topic 1

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 develop and deploy an Azure App Service API app to a Windows-hosted deployment slot named Development. You create additional deployment slots named Testing and Production. You enable auto swap on the Production deployment slot.

You need to ensure that scripts run and resources are available before a swap operation occurs.

Solution: Update the app with a method named statuscheck to run the scripts. Update the app settings for the app. Set the WEBSITE\_SWAP\_WARMUP\_PING\_PATH and WEBSITE\_SWAP\_WARMUP\_PING\_STATUSES with a path to the new method and appropriate response codes.

Does the solution meet the goal?

A. No

B. Yes

**Correct Answer: A**

These are valid warm-up behavior options, but are not helpful in fixing swap problems.

Instead update the web.config file to include the applicationInitialization configuration element. Specify custom initialization actions to run the scripts.

Note: Some apps might require custom warm-up actions before the swap. The applicationInitialization configuration element in web.config lets you specify custom initialization actions. The swap operation waits for this custom warm-up to finish before swapping with the target slot.

Here's a sample web.config fragment.

```
<system.webServer>
  <applicationInitialization>
    <add initializationPage="/" hostName="[app hostname]" />
    <add initializationPage="/Home/About" hostName="[app hostname]" />
  </applicationInitialization>
</system.webServer>
```

Reference:

<https://docs.microsoft.com/en-us/azure/app-service/deploy-staging-slots#troubleshoot-swaps>

Question #34

**HOTSPOT -**

You create the following PowerShell script:

```
$source = New-AzScheduledQueryRuleSource -Query 'Heartbeat | where TimeGenerated > ago(1h)' -DataSourceId "contoso"
$schedule = New-AzScheduledQueryRuleSchedule -FrequencyInMinutes 60 -TimeWindowInMinutes 60
$triggerCondition = New-AzScheduledQueryRuleTriggerCondition -ThresholdOperator "LessThan" -Threshold 5
$aznsActionGroup = New-AzScheduledQueryRuleAznsActionGroup -ActionGroup "contoso" -EmailSubject "Custom email subject"
-CustomWebhookPayload "{ 'alert': '#alertrulename', 'IncludeSearchResults': true }"
$alertingAction = New-AzScheduledQueryRuleAlertingAction -AznsAction $aznsActionGroup -Severity "3" -Trigger $triggerCondition
New-AzScheduledQueryRule -ResourceGroupName "contoso" -Location "eastus" -Action $alertingAction -Enabled $true
-Description "Alert description" -Schedule $schedule -Source $source -Name "Alert Name"
```

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

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

Statements	Yes	No
A log alert is created that sends an email when the CPU percentage is above 60 percent for five minutes.	<input type="radio"/>	<input type="radio"/>
A log alert is created that sends an email when the number of virtual machine heartbeats in the past hour is less than five.	<input type="radio"/>	<input type="radio"/>
The log alert is scheduled to run every two hours.	<input type="radio"/>	<input type="radio"/>

**Answer Area**

Statements	Yes	No
A log alert is created that sends an email when the CPU percentage is above 60 percent for five minutes.	<input type="radio"/>	<input checked="" type="radio"/>
A log alert is created that sends an email when the number of virtual machine heartbeats in the past hour is less than five.	<input checked="" type="radio"/>	<input type="radio"/>
The log alert is scheduled to run every two hours.	<input type="radio"/>	<input checked="" type="radio"/>

Box 1: No -

The AzScheduledQueryRuleSource is Heartbeat, not CPU.

Box 2: Yes -

The AzScheduledQueryRuleSource is Heartbeat!

Note: New-AzScheduledQueryRuleTriggerCondition creates an object of type Trigger Condition. This object is to be passed to the command that creates Alerting

Action object.

Box 3: No -

The schedule is 60 minutes, not two hours.

-FrequencyInMinutes: The alert frequency.

-TimeWindowInMinutes: The alert time window

The New-AzAscheduledQueryRuleSchedule command creates an object of type Schedule. This object is to be passed to the command that creates Log Alert

Rule.

Reference:

<https://docs.microsoft.com/en-us/powershell/module/az.monitor/new-azscheduledqueryrule> <https://docs.microsoft.com/en-us/powershell/module/az.monitor/new-azscheduledqueryruletriggercondition>

Question #35

Topic 1

**DRAG DROP -**

You are developing an Azure Function app.

The app must meet the following requirements:

- Enable developers to write the functions by using the Rust language.
- Declaratively connect to an Azure Blob Storage account.

You need to implement the app.

Which Azure Function app features should you use? To answer, drag the appropriate features to the correct requirements. Each feature 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:

<b>Features</b>	<b>Answer Area</b>						
Custom handler							
Extension bundle							
Trigger							
Runtime							
Policy							
Hosting plan							
	<table border="1"> <thead> <tr> <th><b>Requirement</b></th><th><b>Feature</b></th></tr> </thead> <tbody> <tr> <td>Enable developers to write the functions by using the Rust language.</td><td>Feature</td></tr> <tr> <td>Declaratively connect to an Azure Blob Storage account.</td><td>Feature</td></tr> </tbody> </table>	<b>Requirement</b>	<b>Feature</b>	Enable developers to write the functions by using the Rust language.	Feature	Declaratively connect to an Azure Blob Storage account.	Feature
<b>Requirement</b>	<b>Feature</b>						
Enable developers to write the functions by using the Rust language.	Feature						
Declaratively connect to an Azure Blob Storage account.	Feature						

**Correct Answer:**

<b>Features</b>	<b>Answer Area</b>						
Extension bundle							
Runtime							
Policy							
Hosting plan							
	<table border="1"> <thead> <tr> <th><b>Requirement</b></th><th><b>Feature</b></th></tr> </thead> <tbody> <tr> <td>Enable developers to write the functions by using the Rust language.</td><td>Custom handler</td></tr> <tr> <td>Declaratively connect to an Azure Blob Storage account.</td><td>Trigger</td></tr> </tbody> </table>	<b>Requirement</b>	<b>Feature</b>	Enable developers to write the functions by using the Rust language.	Custom handler	Declaratively connect to an Azure Blob Storage account.	Trigger
<b>Requirement</b>	<b>Feature</b>						
Enable developers to write the functions by using the Rust language.	Custom handler						
Declaratively connect to an Azure Blob Storage account.	Trigger						

Box 1: Custom handler -

Custom handlers can be used to create functions in any language or runtime by running an HTTP server process, for example Go or Rust.

Box 2: Trigger -

Functions are invoked by a trigger and can have exactly one. In addition to invoking the function, certain triggers also serve as bindings. You may also define multiple bindings in addition to the trigger. Bindings provide a declarative way to connect data to your code.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-functions/create-first-function-vs-code-other> <https://docs.microsoft.com/en-us/dotnet/architecture/serverless/azure-functions>

Question #36

Topic 1

**HOTSPOT -**

You are developing an ASP.NET Core web application. You plan to deploy the application to Azure Web App for Containers.

The application needs to store runtime diagnostic data that must be persisted across application restarts. You have the following code:

```
public void SaveDiagData(string data)
{
    var path = Environment.GetEnvironmentVariable("DIAGDATA");
    File.WriteAllText(Path.Combine(path, "data"), data);
}
```

You need to configure the application settings so that diagnostic data is stored as required.

How should you configure the web app's settings? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

App setting	Value
LOCALAPPDATA	
WEBSITE_LOCALCACHE_ENABLED	
DOTNET_HOSTING_OPTIMIZATION_CACHE	
WEBSITES_ENABLE_APP_SERVICE_STORAGE	
DIAGDATA	true /home /local D:\home D:\local

Correct Answer:

**Answer Area**

App setting	Value
LOCALAPPDATA	
WEBSITE_LOCALCACHE_ENABLED	
DOTNET_HOSTING_OPTIMIZATION_CACHE	
WEBSITES_ENABLE_APP_SERVICE_STORAGE	
DIAGDATA	true /home /local D:\home D:\local

Box 1: If WEBSITES\_ENABLE\_APP\_SERVICE\_STORAGE

If WEBSITES\_ENABLE\_APP\_SERVICE\_STORAGE setting is unspecified or set to true, the /home/ directory will be shared across scale instances, and files written will persist across restarts

Box 2: /home -

Reference:

<https://docs.microsoft.com/en-us/azure/app-service/containers/app-service-linux-faq>

Question #37

Topic 1

You are developing a web app that is protected by Azure Web Application Firewall (WAF). All traffic to the web app is routed through an Azure Application

Gateway instance that is used by multiple web apps. The web app address is contoso.azurewebsites.net.

All traffic must be secured with SSL. The Azure Application Gateway instance is used by multiple web apps.

You need to configure the Azure Application Gateway for the web app.

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

NOTE: Each correct selection is worth one point.

- A. In the Azure Application Gateway's HTTP setting, enable the Use for App service setting.
- B. Convert the web app to run in an Azure App service environment (ASE).
- C. Add an authentication certificate for contoso.azurewebsites.net to the Azure Application Gateway.
- D. In the Azure Application Gateway's HTTP setting, set the value of the Override backend path option to contoso22.azurewebsites.net.

**Correct Answer:** AD 

D: The ability to specify a host override is defined in the HTTP settings and can be applied to any back-end pool during rule creation.

The ability to derive the host name from the IP or FQDN of the back-end pool members. HTTP settings also provide an option to dynamically pick the host name from a back-end pool member's FQDN if configured with the option to derive host name from an individual back-end pool member.

A (not C): SSL termination and end to end SSL with multi-tenant services.

In case of end to end SSL, trusted Azure services such as Azure App service web apps do not require whitelisting the backends in the application gateway.

Therefore, there is no need to add any authentication certificates.

## Add HTTP setting

saiappgw-appgw

Question #38 Topic 1

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 develop a software as a service (SaaS) offering to manage photographs. Users upload photos to a web service which then stores the photos in Azure

Storage Blob storage. The storage account type is General-purpose V2.

When photos are uploaded, they must be processed to produce and save a mobile-friendly version of the image. The process to produce a mobile-friendly version of the image must start in less than one minute.

You need to design the process that starts the photo processing.

Solution: Use the Azure Blob Storage change feed to trigger photo processing.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer: B**

The change feed is a log of changes that are organized into hourly segments but appended to and updated every few minutes. These segments are created only when there are blob change events that occur in that hour.

Instead catch the triggered event, so move the photo processing to an Azure Function triggered from the blob upload.

Reference:

<https://docs.microsoft.com/en-us/azure/storage/blobs/storage-blob-change-feed> <https://docs.microsoft.com/en-us/azure/storage/blobs/storage-blob-event-overview>

Reference:

Topic 2 - Question Set 2

Question #1

Topic 2

**HOTSPOT -**

A company develops a series of mobile games. All games use a single leaderboard service.

You have the following requirements:

- Code must be scalable and allow for growth.
- Each record must consist of a playerId, gameId, score, and time played.
- When users reach a new high score, the system will save the new score using the SaveScore function below.
- Each game is assigned an Id based on the series title.

You plan to store customer information in Azure Cosmos DB. The following data already exists in the database:

PartitionKey	RowKey	Email
Harp	Walter	wharp@contoso.com
Smith	Steve	ssmith@contoso.com
Smith	Jeff	jsmith@contoso.com

You develop the following code to save scores in the data store. (Line numbers are included for reference only.)

```

01 public void SaveScore(string gameId, string playerId, int score, long timePlayed)
02 {
03     CloudStorageAccount storageAccount = CloudStorageAccount.Parse(connectionString);
04     CloudTableClient tableClient = storageAccount.CreateCloudTableClient();
05     CloudTable table = tableClient.GetTableReference("scoreTable");
06     table.CreateIfNotExists();
07     var scoreRecord = new PlayerScore(gameId, playerId, score, timePlayed);
08     TableOperation insertOperation = TableOperation.Insert(scoreRecord);
09     table.Execute(insertOperation);
10 }
```

You develop the following code to query the database. (Line numbers are included for reference only.)

```

01 CloudTableClient tableClient = account.CreateCloudTableClient();
02 CloudTable table = tableClient.GetTableReference("people");
03 TableQuery<CustomerEntity> query = new TableQuery<CustomerEntity>()
04     .Where(TableQuery.CombineFilters(
05         TableQuery.GenerateFilterCondition("PartitionKey", QueryComparisons.Equal, "Smith"),
06         TableOperators.And,
07         TableQuery.GenerateFilterCondition("Email", QueryComparisons.Equal, "ssmith@contoso.com")
08     ));
09 await table.ExecuteQuerySegmentedAsync<CustomerEntity>(query, null);
```

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

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

Statements	Yes	No
SaveScore will work with Cosmos DB.	<input type="radio"/>	<input type="radio"/>
SaveScore will update and replace a record if one already exists with the same playerId and gameId.	<input type="radio"/>	<input type="radio"/>
Leader board data for the game will be automatically partitioned using gameId.	<input type="radio"/>	<input type="radio"/>
SaveScore will store the values for the gameId and playerId parameters in the database.	<input type="radio"/>	<input type="radio"/>

**Answer Area**

	<b>Statements</b>	<b>Yes</b>	<b>No</b>
	SaveScore will work with Cosmos DB.	<input checked="" type="radio"/>	<input type="radio"/>
<b>Correct Answer:</b>	SaveScore will update and replace a record if one already exists with the same playerId and gameId.	<input type="radio"/>	<input checked="" type="radio"/>
	Leader board data for the game will be automatically partitioned using gameId.	<input type="radio"/>	<input checked="" type="radio"/>
	SaveScore will store the values for the gameId and playerId parameters in the database.	<input checked="" type="radio"/>	<input type="radio"/>

Box 1: Yes -

Create a table.

A CloudTableClient object lets you get reference objects for tables and entities. The following code creates a CloudTableClient object and uses it to create a new

CloudTable object, which represents a table

```
// Retrieve storage account from connection-string.
```

```
CloudStorageAccount storageAccount =
```

```
CloudStorageAccount.parse(storageConnectionString);
```

```
// Create the table client.
```

```
CloudTableClient tableClient = storageAccount.createCloudTableClient();
```

```
// Create the table if it doesn't exist.
```

```
String tableName = "people";
```

```
CloudTable cloudTable = tableClient.getTableReference(tableName); cloudTable.createIfNotExists();
```

Box 2: No -

New records are inserted with TableOperation.insert. Old records are not updated.

To update old records TableOperation.insertOrReplace should be used instead.

Box 3: No -

Box 4: Yes -

Reference:

<https://docs.microsoft.com/en-us/azure/cosmos-db/table-storage-how-to-use-java>

## Question #2

## Topic 2

## HOTSPOT -

You are developing a solution that uses the Azure Storage Client library for .NET. You have the following code: (Line numbers are included for reference only.)

```

01 CloudBlockBlob src = null;
02 try
03 {
04     src = container.ListBlobs().OfType<CloudBlockBlob>().FirstOrDefault();
05     var id = await src.AcquireLeaseAsync(null);
06     var dst = container.GetBlockBlobReference(src.Name);
07     string cpid = await dst.StartCopyAsync(src);
08     await dst.FetchAttributeAsync();
09     return id;
10 }
11 catch (Exception e)
12 {
13     throw;
14 }
15 finally
16 {
17     if (src != null)
18         await src.FetchAttributesAsync();
19     if (src.Properties.LeaseState != LeaseState.Available)
20         await src.BreakLeaseAsync(new TimeSpan(0));
21 }

```

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

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

Statement	Yes	No
The code creates an infinite lease	<input type="radio"/>	<input type="radio"/>
The code at line 06 always creates a new blob	<input type="radio"/>	<input type="radio"/>
The finally block releases the lease	<input type="radio"/>	<input type="radio"/>

**Answer Area**

Statement	Yes	No
The code creates an infinite lease	<input checked="" type="radio"/>	<input type="radio"/>
The code at line 06 always creates a new blob	<input type="radio"/>	<input checked="" type="radio"/>
The finally block releases the lease	<input checked="" type="radio"/>	<input type="radio"/>

Box 1: Yes -

AcquireLeaseAsync does not specify leaseTime.

leaseTime is a TimeSpan representing the span of time for which to acquire the lease, which will be rounded down to seconds. If null, an infinite lease will be acquired. If not null, this must be 15 to 60 seconds.

Box 2: No -

The GetBlockBlobReference method just gets a reference to a block blob in this container.

Box 3: Yes -

The BreakLeaseAsync method initiates an asynchronous operation that breaks the current lease on this container.

Reference:

<https://docs.microsoft.com/en-us/dotnet/api/microsoft.azure.storage.blob.cloudblobcontainer.acquireleaseasync>  
<https://docs.microsoft.com/en-us/dotnet/api/microsoft.azure.storage.blob.cloudblobcontainer.getblockblobreference>  
<https://docs.microsoft.com/en-us/dotnet/api/microsoft.azure.storage.blob.cloudblobcontainer.breakleaseasync>

## Question #3

## Topic 2

You are building a website that uses Azure Blob storage for data storage. You configure Azure Blob storage lifecycle to move all blobs to the archive tier after 30 days.

Customers have requested a service-level agreement (SLA) for viewing data older than 30 days.

You need to document the minimum SLA for data recovery.

Which SLA should you use?

- A. at least two days
- B. between one and 15 hours
- C. at least one day
- D. between zero and 60 minutes

**Correct Answer:** B 

The archive access tier has the lowest storage cost. But it has higher data retrieval costs compared to the hot and cool tiers. Data in the archive tier can take several hours to retrieve depending on the priority of the rehydration. For small objects, a high priority rehydrate may retrieve the object from archive in under 1 hour.

Reference:

<https://docs.microsoft.com/en-us/azure/storage/blobs/storage-blob-storage-tiers?tabs=azure-portal>

Question #4

Topic 2

**HOTSPOT -**

You are developing a ticket reservation system for an airline.

The storage solution for the application must meet the following requirements:

- Ensure at least 99.99% availability and provide low latency.
- Accept reservations even when localized network outages or other unforeseen failures occur.
- Process reservations in the exact sequence as reservations are submitted to minimize overbooking or selling the same seat to multiple travelers.
- Allow simultaneous and out-of-order reservations with a maximum five-second tolerance window.

You provision a resource group named `airlineResourceGroup` in the Azure South-Central US region.

You need to provision a SQL API Cosmos DB account to support the app.

How should you complete the Azure CLI commands? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

```
resourceGroupName='airlineResourceGroup'
name='docdb-airline-reservations'
databaseName='docdb-tickets-database'
collectionName='docdb-tickets-collection'
consistencyLevel= 
az cosmosdb create \
--name $name \
--enable-virtual-network true \
--enable-automatic-failover true \
--kind 'GlobalDocumentDB' \
--kind 'MongoDB' \
--resource-group $resourceGroupName \
--max-interval 5 \
--locations 'southcentralus'
--locations 'eastus'
--locations 'southcentralus=0 eastus=1 westus=2'
--locations 'southcentralus=0'
--default-consistency-level = $consistencylevel
```

## Answer Area

```
resourceGroupName='airlineResourceGroup'
name='docdb-airline-reservations'
databaseName='docdb-tickets-database'
collectionName='docdb-tickets-collection'
consistencyLevel=
```

Strong
Eventual
ConsistentPrefix
BoundedStaleness

```
az cosmosdb create \
--name $name \
```

Correct Answer:

```
--enable-virtual-network true \
--enable-automatic-failover true \
--kind 'GlobalDocumentDB' \
--kind 'MongoDB' \
--resource-group $resourceGroupName \
--max-interval 5 \
--locations 'southcentralus'
--locations 'eastus'
--locations 'southcentralus=0 eastus=1 westus=2'
--locations 'southcentralus=0'
--default-consistency-level = $consistencylevel
```

Box 1: BoundedStaleness -

Bounded staleness: The reads are guaranteed to honor the consistent-prefix guarantee. The reads might lag behind writes by at most "K" versions (that is,

"updates") of an item or by "T" time interval. In other words, when you choose bounded staleness, the "staleness" can be configured in two ways:

The number of versions (K) of the item

The time interval (T) by which the reads might lag behind the writes

Incorrect Answers:

Strong -

Strong consistency offers a linearizability guarantee. Linearizability refers to serving requests concurrently. The reads are guaranteed to return the most recent committed version of an item. A client never sees an uncommitted or partial write. Users are always guaranteed to read the latest committed write.

Box 2: --enable-automatic-failover true\

For multi-region Cosmos accounts that are configured with a single-write region, enable automatic-failover by using Azure CLI or Azure portal.

After you enable automatic failover, whenever there is a regional disaster, Cosmos DB will automatically failover your account.

Question: Accept reservations event when localized network outages or other unforeseen failures occur.

Box 3: --locations'southcentralus=0 eastus=1 westus=2

Need multi-region.

Reference:

<https://docs.microsoft.com/en-us/azure/cosmos-db/consistency-levels> <https://github.com/MicrosoftDocs/azure-docs/blob/master/articles/cosmos-db/manage-with-cli.md>

Question #5

Topic 2

**HOTSPOT -**

You are preparing to deploy a Python website to an Azure Web App using a container. The solution will use multiple containers in the same container group. The

Dockerfile that builds the container is as follows:

```
FROM python:3
ADD website.py
CMD [ "python", "./website.py"]
```

You build a container by using the following command. The Azure Container Registry instance named images is a private registry.

```
docker build -t images.azurecr.io/website:v1.0.0
```

The user name and password for the registry is admin.

The Web App must always run the same version of the website regardless of future builds.

You need to create an Azure Web App to run the website.

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

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

```
az configure --defaults web=website
az configure --defaults group=website
az appservice plan create --name websitePlan
```

- sku SHARED
- tags container
- sku B1 --hyper-v
- sku B1 --is-linux

```
az webapp create --plan websitePlan
```

- deployment-source-url images.azurecr.io/website:v1.0.0
- deployment-source-url images.azurecr.io/website:latest
- deployment-container-image-name images.azurecr.io/website:v1.0.0
- deployment-container-image-name images.azurecr.io/website:latest

```
az webapp config
```

- set --python-version 2.7 --generic-configurations user=admin password=admin
- set --python-version 3.6 --generic-configurations user=admin password=admin
- container set --docker-registry-server-url https://images.azurecr.io -u admin -p admin
- container set --docker-registry-server-url https://images.azurecr.io/website -u admin -p admin

**Correct Answer:****Answer Area**

```
az configure --defaults web=website  
az configure --defaults group=website  
az appservice plan create --name websitePlan
```

--sku SHARED
--tags container
<b>--sku B1 --hyper-v</b>
--sku B1 --is-linux

```
az webapp create --plan websitePlan
```

--deployment-source-url images.azurecr.io/website:v1.0.0
--deployment-source-url images.azurecr.io/website:latest
--deployment-container-image-name images.azurecr.io/website:v1.0.0
--deployment-container-image-name images.azurecr.io/website:latest

```
az webapp config
```

set --python-version 2.7 --generic-configurations user=admin password=admin
set --python-version 3.6 --generic-configurations user=admin password=admin
<b>container set --docker-registry-server-url https://images.azurecr.io -u admin -p admin</b>
container set --docker-registry-server-url https://images.azurecr.io/website -u admin -p admin

Box 1: --SKU B1 --hyper-v -

--hyper-v

Host web app on Windows container.

Box 2: --deployment-source-url images.azurecr.io/website:v1.0.0

--deployment-source-url -u

Git repository URL to link with manual integration.

The Web App must always run the same version of the website regardless of future builds.

Incorrect:

--deployment-container-image-name -i

Linux only. Container image name from Docker Hub, e.g. publisher/image-name:tag.

Box 3: az webapp config container set -url https://images.azurecr.io -u admin -p admin az webapp config container set

Set a web app container's settings.

Parameter: --docker-registry-server-url -r

The container registry server url.

The Azure Container Registry instance named images is a private registry.

Example:

```
az webapp config container set --docker-registry-server-url https://{{azure-container-registry-name}}.azurecr.io
```

Reference:

<https://docs.microsoft.com/en-us/cli/azure/appservice/plan>

Question #6

Topic 2

**HOTSPOT -**

You are developing a back-end Azure App Service that scales based on the number of messages contained in a Service Bus queue.

A rule already exists to scale up the App Service when the average queue length of unprocessed and valid queue messages is greater than 1000.

You need to add a new rule that will continuously scale down the App Service as long as the scale up condition is not met.

How should you configure the Scale rule? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**Scale rule ×

## Metric source

- ▼
  - Storage queue
  - Service Bus queue
  - Current resource
  - Storage queue (classic)

## Resource type

Service Bus Namespaces ▼

## Resource

MessageQueue1103 ▼

## \* Queues

itemqueue ▼

## Criteria

## \* Metric name

- ▼
  - Message Count
  - Active Message Count

1 minute time grain

\* Time grain statistic ●

- ▼
  - Total
  - Maximum
  - Average
  - Count

## \* Operator

- ▼

**Answer Area**Scale rule x

Metric source

- Storage queue
- Service Bus queue
- Current resource
- Storage queue (classic)

Resource type

Service Bus Namespaces

Resource

MessageQueue1103

\* Queues

itemqueue

**Correct Answer:****Criteria**

\* Metric name

- Message Count
- Active Message Count

1 minute time grain

\* Time grain statistic ⓘ

- Total
- Maximum
- Average
- Count

\* Operator

Box 1: Service bus queue -

You are developing a back-end Azure App Service that scales based on the number of messages contained in a Service Bus queue.

Box 2: ActiveMessage Count -

ActiveMessageCount: Messages in the queue or subscription that are in the active state and ready for delivery.

Box 3: Count -

Box 4: Less than or equal to -

You need to add a new rule that will continuously scale down the App Service as long as the scale up condition is not met.

Box 5: Decrease count by



Question #7

DRAG DROP -

You have an application that uses Azure Blob storage.

You need to update the metadata of the blobs.

Which three methods should you use to develop the solution? To answer, move the appropriate methods from the list of methods to the answer area and arrange them in the correct order.

Select and Place:

**Methods**

Metadata.Add  
SetMetadataAsync  
FetchAttributesAsync  
UploadFileStream  
SetPropertiesAsync

**Answer Area****Methods**

Correct Answer:

FetchAttributesAsync  
UploadFileStream

**Answer Area**

Metadata.Add  
SetMetadataAsync  
SetPropertiesAsync

Metadata.Add example:

```
// Add metadata to the dictionary by calling the Add method
metadata.Add("docType", "textDocuments");
```

SetMetadataAsync example:

```
// Set the blob's metadata.
await blob.SetMetadataAsync(metadata);
// Set the blob's properties.
await blob.SetPropertiesAsync();
```

Reference:

<https://docs.microsoft.com/en-us/azure/storage/blobs/storage-blob-properties-metadata>

## Question #8

Topic 2

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 are developing an Azure solution to collect point-of-sale (POS) device data from 2,000 stores located throughout the world. A single device can produce

2 megabytes (MB) of data every 24 hours. Each store location has one to five devices that send data.

You must store the device data in Azure Blob storage. Device data must be correlated based on a device identifier. Additional stores are expected to open in the future.

You need to implement a solution to receive the device data.

Solution: Provision an Azure Event Grid. Configure the machine identifier as the partition key and enable capture.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer:** A 

Reference:

<https://docs.microsoft.com/en-us/azure/event-grid/compare-messaging-services>

## Question #9

Topic 2

You develop Azure solutions.

A .NET application needs to receive a message each time an Azure virtual machine finishes processing data. The messages must NOT persist after being processed by the receiving application.

You need to implement the .NET object that will receive the messages.

Which object should you use?

A. QueueClient

B. SubscriptionClient

C. TopicClient

D. CloudQueueClient

**Correct Answer:** D 

A queue allows processing of a message by a single consumer. Need a CloudQueueClient to access the Azure VM.

Incorrect Answers:

B, C: In contrast to queues, topics and subscriptions provide a one-to-many form of communication in a publish and subscribe pattern. It's useful for scaling to large numbers of recipients.

Reference:

<https://docs.microsoft.com/en-us/azure/service-bus-messaging/service-bus-queues-topics-subscriptions>

Question #10

Topic 2

**DRAG DROP -**

You are maintaining an existing application that uses an Azure Blob GPv1 Premium storage account. Data older than three months is rarely used. Data newer than three months must be available immediately. Data older than a year must be saved but does not need to be available immediately.

You need to configure the account to support a lifecycle management rule that moves blob data to archive storage for data not modified in the last year.

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
Upgrade the storage account to GPv2	
Create a new GPv2 Standard account and set its default access tier level to cool	▶
Change the storage account access tier from hot to cool	◀
Copy the data to be archived to a Standard GPv2 storage account and then delete the data from the original storage account	↑ ↓

Actions	Answer Area
	Upgrade the storage account to GPv2
Create a new GPv2 Standard account and set its default access tier level to cool	▶
	Copy the data to be archived to a Standard GPv2 storage account and then delete the data from the original storage account
	◀
	Change the storage account access tier from hot to cool

Step 1: Upgrade the storage account to GPv2

Object storage data tiering between hot, cool, and archive is supported in Blob Storage and General Purpose v2 (GPv2) accounts. General Purpose v1 (GPv1) accounts don't support tiering.

You can easily convert your existing GPv1 or Blob Storage accounts to GPv2 accounts through the Azure portal.

Step 2: Copy the data to be archived to a Standard GPv2 storage account and then delete the data from the original storage account

Step 3: Change the storage account access tier from hot to cool

Note: Hot - Optimized for storing data that is accessed frequently.

Cool - Optimized for storing data that is infrequently accessed and stored for at least 30 days.

Archive - Optimized for storing data that is rarely accessed and stored for at least 180 days with flexible latency requirements, on the order of hours.

Only the hot and cool access tiers can be set at the account level. The archive access tier can only be set at the blob level.

Reference:

<https://docs.microsoft.com/en-us/azure/storage/blobs/storage-blob-storage-tiers>

## Question #11

Topic 2

You develop Azure solutions.

You must connect to a No-SQL globally-distributed database by using the .NET API.

You need to create an object to configure and execute requests in the database.

Which code segment should you use?

- A. new Container(EndpointUri, PrimaryKey);
- B. new Database(EndpointUri, PrimaryKey);
- C. new CosmosClient(EndpointUri, PrimaryKey);

**Correct Answer:** C 

Example:

```
// Create a new instance of the Cosmos Client  
this.cosmosClient = new CosmosClient(EndpointUri, PrimaryKey)  
//ADD THIS PART TO YOUR CODE  
await this.CreateDatabaseAsync();
```

Reference:

<https://docs.microsoft.com/en-us/azure/cosmos-db/sql-api-get-started>

## Question #12

Topic 2

You have an existing Azure storage account that stores large volumes of data across multiple containers.

You need to copy all data from the existing storage account to a new storage account. The copy process must meet the following requirements:

- Automate data movement.
- Minimize user input required to perform the operation.
- Ensure that the data movement process is recoverable.

What should you use?

- A. AzCopy
- B. Azure Storage Explorer
- C. Azure portal
- D. .NET Storage Client Library

**Correct Answer:** A 

You can copy blobs, directories, and containers between storage accounts by using the AzCopy v10 command-line utility.

The copy operation is synchronous so when the command returns, that indicates that all files have been copied.

Reference:

<https://docs.microsoft.com/en-us/azure/storage/common/storage-use-azcopy-blobs-copy>

Question #13

Topic 2

**DRAG DROP -**

You are developing a web service that will run on Azure virtual machines that use Azure Storage. You configure all virtual machines to use managed identities.

You have the following requirements:

- Secret-based authentication mechanisms are not permitted for accessing an Azure Storage account.
- Must use only Azure Instance Metadata Service endpoints.

You need to write code to retrieve an access token to access Azure Storage. To answer, drag the appropriate code segments to the correct locations. Each code segment may be used 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:

**Code segment 1**

`http://localhost:50342/oauth2/token`  
`http://169.254.169.254:50432/oauth2/token`  
`http://localhost/metadata/identity/oauth2/token`  
`http://169.254.169.254/metadata/identity/oauth2/token`

**Answer Area**

```
var url = " Code segment 1 " ;  

var queryString = "...";  

var client = new HttpClient();  

var response = await client.GetAsync(url + queryString);  

var payload = await response.Content.ReadAsStringAsync();  

return Code segment 2
```

**Code segment 2**

`XDocument.Parse(payload);`  
`new MultipartContent(payload);`  
`new NetworkCredential("Azure", payload);`  
`JsonConvert.DeserializeObject<Dictionary<string, string>>(payload);`

**Correct Answer:****Code segment 1**

`http://localhost:50342/oauth2/token`  
`http://169.254.169.254:50432/oauth2/token`  
`http://localhost/metadata/identity/oauth2/token`

**Answer Area**

```
var url = " http://169.254.169.254/metadata/identity/oauth2/token " ;  

var queryString = "...";  

var client = new HttpClient();  

var response = await client.GetAsync(url + queryString);  

var payload = await response.Content.ReadAsStringAsync();  

return JsonConvert.DeserializeObject<Dictionary<string, string>>(payload);
```

**Code segment 2**

`XDocument.Parse(payload);`  
`new MultipartContent(payload);`  
`new NetworkCredential("Azure", payload);`

Azure Instance Metadata Service endpoints "/oauth2/token"

Box 1: `http://169.254.169.254/metadata/identity/oauth2/token`

Sample request using the Azure Instance Metadata Service (IMDS) endpoint (recommended):

GET '`http://169.254.169.254/metadata/identity/oauth2/token?api-version=2018-02-01&resource=https://management.azure.com/`' HTTP/1.1

Metadata: true

Box 2: `JsonConvert.DeserializeObject<Dictionary<string, string>>(payload);`

Deserialized token response; returning access code.

Reference:

<https://docs.microsoft.com/en-us/azure/active-directory/managed-identities-azure-resources/how-to-use-vm-token>

<https://docs.microsoft.com/en-us/azure/service-fabric/how-to-managed-identity-service-fabric-app-code>



Question #14

Topic 2

**DRAG DROP -**

You are developing a new page for a website that uses Azure Cosmos DB for data storage. The feature uses documents that have the following format:

```
{  
    "name": "John",  
    "city" : "Seattle"  
}
```

You must display data for the new page in a specific order. You create the following query for the page:

```
SELECT*  
FROM People p  
ORDER BY p.name, p.city DESC
```

You need to configure a Cosmos DB policy to support the query.

How should you configure the policy? To answer, drag the appropriate JSON segments to the correct locations. Each JSON 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:

JSON segments	Answer Area
orderBy	{
sortOrder	"automatic": true, "ngMode": "Consistent", "includedPaths": [
ascending	{ "path": "/**"
descending	} ], "excludedPaths": [], " [
compositeIndexes	{ "path": "/name", "order": "descending" }, { "path": "/city", "order": " [ ] "

**Correct Answer:****JSON segments**

orderBy  
sortOrder  
ascending  
descending  
compositeIndexes

**Answer Area**

```
{  
    "automatic": true,  
    "ngMode": "Consistent",  
    "includedPaths": [  
        {  
            "path": "/"  
        }  
    ], "excludedPaths": [],  
    "compositeIndexes": [  
        [  
            {  
                "path": "/name", "order": "descending"  
            },  
            {  
                "path": "/city", "order": "descending"  
            }  
        ]  
    ]  
}
```

Box 1: compositeIndexes -

You can order by multiple properties. A query that orders by multiple properties requires a composite index.

Box 2: descending -

Example: Composite index defined for (name ASC, age ASC):

It is optional to specify the order. If not specified, the order is ascending.

```
{  
    "automatic":true,  
    "indexingMode":"Consistent",  
    "includedPaths": [  
        {  
            "path": "/"  
        }  
    ],  
    "excludedPaths":[],  
    "compositeIndexes": [  
        [  
            {  
                "path": "/name",  
            },  
            {  
                "path": "/age",  
            }  
        ]  
    ]  
}
```

Question #15

Topic 2

**HOTSPOT -**

You are building a traffic monitoring system that monitors traffic along six highways. The system produces time series analysis-based reports for each highway.

Data from traffic sensors are stored in Azure Event Hub.

Traffic data is consumed by four departments. Each department has an Azure Web App that displays the time series-based reports and contains a WebJob that processes the incoming data from Event Hub. All Web Apps run on App Service Plans with three instances.

Data throughput must be maximized. Latency must be minimized.

You need to implement the Azure Event Hub.

Which settings should you use? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

Setting	Value
Number of partitions	<input type="button" value="▼"/>
	<input type="button" value="3"/> <input type="button" value="4"/> <input checked="" type="button" value="6"/> <input type="button" value="12"/>
Partition Key	<input type="button" value="▼"/>
	<input type="button" value="Highway"/> <input type="button" value="Department"/> <input type="button" value="Timestamp"/> <input type="button" value="VM name"/>

**Answer Area**

Setting	Value
Number of partitions	<input type="button" value="▼"/>
Correct Answer:	<input type="button" value="3"/> <input type="button" value="4"/> <input checked="" type="button" value="6"/> <input type="button" value="12"/>
Partition Key	<input type="button" value="▼"/>
	<input checked="" type="button" value="Highway"/> <input type="button" value="Department"/> <input type="button" value="Timestamp"/> <input type="button" value="VM name"/>

Box 1: 6 -

The number of partitions is specified at creation and must be between 2 and 32.

There are 6 highways.

Box 2: Highway -

Reference:

<https://docs.microsoft.com/en-us/azure/event-hubs/event-hubs-features>

Question #16

Topic 2

**DRAG DROP -**

You are developing a microservices solution. You plan to deploy the solution to a multinode Azure Kubernetes Service (AKS) cluster.

You need to deploy a solution that includes the following features:

- reverse proxy capabilities
- configurable traffic routing
- TLS termination with a custom certificate

Which components should you use? To answer, drag the appropriate components to the correct requirements. Each component 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**

Components	Action	Component
Helm		
Draft	Deploy solution.	
Brigade	View cluster and external IP addressing.	
KubeCtl	Implement a single, public IP endpoint that is routed to multiple microservices.	
Ingress Controller		
CoreDNS		
Virtual Kubelet		

**Answer Area**

Components	Action	Component
Helm		
Draft	Deploy solution.	Helm
Brigade	View cluster and external IP addressing.	KubeCtl
KubeCtl	Implement a single, public IP endpoint that is routed to multiple microservices.	Ingress Controller
Ingress Controller		
CoreDNS		
Virtual Kubelet		

Correct Answer:

Box 1: Helm -

To create the ingress controller, use Helm to install nginx-ingress.

Box 2: kubectl -

To find the cluster IP address of a Kubernetes pod, use the kubectl get pod command on your local machine, with the option -o wide .

**Box 3: Ingress Controller -**

An ingress controller is a piece of software that provides reverse proxy, configurable traffic routing, and TLS termination for Kubernetes services. Kubernetes ingress resources are used to configure the ingress rules and routes for individual Kubernetes services.

**Incorrect Answers:**

Virtual Kubelet: Virtual Kubelet is an open-source Kubernetes kubelet implementation that masquerades as a kubelet. This allows Kubernetes nodes to be backed by Virtual Kubelet providers such as serverless cloud container platforms.

CoreDNS: CoreDNS is a flexible, extensible DNS server that can serve as the Kubernetes cluster DNS. Like Kubernetes, the CoreDNS project is hosted by the

CNCF.

**Reference:**

<https://docs.microsoft.com/bs-cyrl-ba/azure/aks/ingress-basic> <https://www.digitalocean.com/community/tutorials/how-to-inspect-kubernetes-networking>

Question #17

Topic 2

**DRAG DROP -**

You are implementing an order processing system. A point of sale application publishes orders to topics in an Azure Service Bus queue. The Label property for the topic includes the following data:

Property	Description
ShipLocation	the country/region where the order will be shipped
CorrelationId	a priority value for the order
Quantity	a user-defined field that stores the quantity of items in an order
AuditedAt	a user-defined field that records the date an order is audited

The system has the following requirements for subscriptions:

Subscription type	Comments
FutureOrders	This subscription is reserved for future use and must not receive any orders
HighPriorityOrders	Handle all high priority orders and international orders
InternationalOrders	Handle orders where the country/region is not United States
HighQuantityOrders	Handle only orders with quantities greater than 100 units
AllOrders	This subscription is used for auditing purposes. This subscription must receive every single order. AllOrders has an Action defined that updates the AuditedAt property to include the date and time it was received by the subscription.

You need to implement filtering and maximize throughput while evaluating filters.

Which filter types should you implement? To answer, drag the appropriate filter types to the correct subscriptions. Each filter 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:

Filter types		Answer Area	
SQLFilter		Subscription	Filter type
CorrelationFilter	•	FutureOrders	filter type
No Filter	•	HighPriorityOrders	filter type
	•	InternationalOrders	filter type
	•	HighQuantityOrders	filter type
	•	AllOrders	filter type

Filter types		Answer Area	
SQLFilter		Subscription	Filter type
CorrelationFilter	•	FutureOrders	SQLFilter
Correct Answer: No Filter	•	HighPriorityOrders	CorrelationFilter
	•	InternationalOrders	SQLFilter
	•	HighQuantityOrders	SQLFilter
	•	AllOrders	No Filter

FutureOrders: SQLFilter -

HighPriorityOrders: CorrelationFilter

CorrelationID only -

InternationalOrders: SQLFilter -

Country NOT USA requires an SQL Filter

HighQuantityOrders: SQLFilter -

Need to use relational operators so an SQL Filter is needed.

AllOrders: No Filter -

SQL Filter: SQL Filters - A SqlFilter holds a SQL-like conditional expression that is evaluated in the broker against the arriving messages' user-defined properties and system properties. All system properties must be prefixed with sys. in the conditional expression. The SQL-language subset for filter conditions tests for the existence of properties (EXISTS), as well as for null-values (IS NULL), logical NOT/AND/OR, relational operators, simple numeric arithmetic, and simple text pattern matching with LIKE.

Correlation Filters - A CorrelationFilter holds a set of conditions that are matched against one or more of an arriving message's user and system properties. A common use is to match against the CorrelationId property, but the application can also choose to match against ContentType, Label, MessageId, ReplyTo,

ReplyToSessionId, SessionId, To, and any user-defined properties. A match exists when an arriving message's value for a property is equal to the value specified in the correlation filter. For string expressions, the comparison is case-sensitive. When specifying multiple match properties, the filter combines them as a logical

AND condition, meaning for the filter to match, all conditions must match.

Boolean filters - The TrueFilter and FalseFilter either cause all arriving messages (true) or none of the arriving messages (false) to be selected for the subscription.

Reference:

<https://docs.microsoft.com/en-us/azure/service-bus-messaging/topic-filters>

Question #18

Topic 2

**DRAG DROP -**

Your company has several websites that use a company logo image. You use Azure Content Delivery Network (CDN) to store the static image. You need to determine the correct process of how the CDN and the Point of Presence (POP) server will distribute the image and list the items in the correct order.

In which order do the actions occur? To answer, move all actions from the list of actions to the answer area and arrange them in the correct order. Select and Place:

**Actions****Answer Area**

If no edge servers in the POP have the image in cache, the POP requests the file from the origin server.



A user requests the image from the CDN URL. The DNS routes the request to the best performing POP location.

Subsequent requests for the file may be directed to the same POP using the CDN logo image URL. The POP edge server returns the file from cache if the TTL has not expired.

The origin server returns the logo image to an edge server in the POP. An edge server in the POP caches the logo image and returns the image to the client.

**Correct Answer:****Actions****Answer Area**

If no edge servers in the POP have the image in cache, the POP requests the file from the origin server.



A user requests the image from the CDN URL. The DNS routes the request to the best performing POP location.

Subsequent requests for the file may be directed to the same POP using the CDN logo image URL. The POP edge server returns the file from cache if the TTL has not expired.

The origin server returns the logo image to an edge server in the POP. An edge server in the POP caches the logo image and returns the image to the client.

A user requests the image from the CDN URL. The DNS routes the request to the best performing POP location.

If no edge servers in the POP have the image in cache, the POP requests the file from the origin server.

The origin server returns the logo image to an edge server in the POP. An edge server in the POP caches the logo image and returns the image to the client.

Subsequent requests for the file may be directed to the same POP using the CDN logo image URL. The POP edge server returns the file from cache if the TTL has not expired.

Step 1: A user requests the image..

A user requests a file (also called an asset) by using a URL with a special domain name, such as <endpoint name>.azureedge.net. This name can be an endpoint hostname or a custom domain. The DNS routes the request to the best performing POP location, which is usually the POP that is geographically closest to the user.

Step 2: If no edge servers in the POP have the..

If no edge servers in the POP have the file in their cache, the POP requests the file from the origin server. The origin server can be an Azure Web App, Azure

Cloud Service, Azure Storage account, or any publicly accessible web server.

Step 3: The origin server returns the..

The origin server returns the file to an edge server in the POP.

An edge server in the POP caches the file and returns the file to the original requestor (Alice). The file remains cached on the edge server in the POP until the time-to-live (TTL) specified by its HTTP headers expires. If the origin server didn't specify a TTL, the default TTL is seven days.

Step 4: Subsequent requests for..

Additional users can then request the same file by using the same URL that the original user used, and can also be directed to the same POP.

If the TTL for the file hasn't expired, the POP edge server returns the file directly from the cache. This process results in a faster, more responsive user experience.

Reference:

<https://docs.microsoft.com/en-us/azure/cdn/cdn-overview>

Question #19

*Topic 2*

You are developing an Azure Cosmos DB solution by using the Azure Cosmos DB SQL API. The data includes millions of documents. Each document may contain hundreds of properties.

The properties of the documents do not contain distinct values for partitioning. Azure Cosmos DB must scale individual containers in the database to meet the performance needs of the application by spreading the workload evenly across all partitions over time.

You need to select a partition key.

Which two partition keys can you use? Each correct answer presents a complete solution.

NOTE: Each correct selection is worth one point.

- A. a single property value that does not appear frequently in the documents
- B. a value containing the collection name
- C. a single property value that appears frequently in the documents
- D. a concatenation of multiple property values with a random suffix appended
- E. a hash suffix appended to a property value

**Correct Answer:** DE 

You can form a partition key by concatenating multiple property values into a single artificial partitionKey property. These keys are referred to as synthetic keys.

Another possible strategy to distribute the workload more evenly is to append a random number at the end of the partition key value. When you distribute items in this way, you can perform parallel write operations across partitions.

Note: It's the best practice to have a partition key with many distinct values, such as hundreds or thousands. The goal is to distribute your data and workload evenly across the items associated with these partition key values. If such a property doesn't exist in your data, you can construct a synthetic partition key.

Reference:

<https://docs.microsoft.com/en-us/azure/cosmos-db/synthetic-partition-keys>

Question #20

Topic 2

**HOTSPOT -**

You are developing an Azure-hosted e-commerce web application. The application will use Azure Cosmos DB to store sales orders. You are using the latest SDK to manage the sales orders in the database.

You create a new Azure Cosmos DB instance. You include a valid endpoint and valid authorization key to an appSettings.json file in the code project.

You are evaluating the following application code: (Line number are included for reference only.)

```
01 using System;
02 using System.Threading.Tasks;
03 using Microsoft.Azure.Cosmos;
04 using Microsoft.Extensions.Configuration;
05 using Newtonsoft.Json;
06 namespace SalesOrders
07 {
08     public class SalesOrder
09     {
10         ...
11     }
12     internal class ManageSalesOrders
13     {
14         private static async Task GenerateSalesOrders()
15         {
16             IConfigurationRoot configuration = new ConfigurationBuilder().AddJsonFile("appSettings.json").Build();
17             string endpoint = configuration["EndPointUrl"];
18             string authKey = configuration["AuthorizationKey"];
19             using CosmosClient client = new CosmosClient(endpoint, authKey);
20             Database database = null;
21             using (await client.GetDatabase("SalesOrders").DeleteStreamAsync()) { }
22             database = await client.CreateDatabaseIfNotExistsAsync("SalesOrders");
23             Container container1 = await database.CreateContainerAsync(id: "Container1", partitionKeyPath: "/AccountNumber");
24             Container container2 = await database.CreateContainerAsync(id: "Container2", partitionKeyPath: "/AccountNumber");
25             SalesOrder salesOrder1 = new SalesOrder() { AccountNumber = "123456" };
26             await container1.CreateItemAsync(salesOrder1, new PartitionKey(salesOrder1.AccountNumber));
27             SalesOrder salesOrder2 = new SalesOrder() { AccountNumber = "654321" };
28             await container1.CreateItemAsync(salesOrder2, new PartitionKey(salesOrder2.AccountNumber));
29             SalesOrder salesOrder3 = new SalesOrder() { AccountNumber = "109876" };
30             await container2.CreateItemAsync(salesOrder3, new PartitionKey(salesOrder3.AccountNumber));
31             _ = await database.CreateUserAsync("User1");
32             User user1 = database.GetUser("User1");
33             _ = await user1.ReadAsync();
34         }
35     }
36 }
```

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

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

Statements	Yes	No
A database named SalesOrders is created. The database will include two containers.	<input type="radio"/>	<input type="radio"/>
Container1 will contain two items.	<input type="radio"/>	<input type="radio"/>
Container2 will contain one item.	<input type="radio"/>	<input type="radio"/>

**Answer Area**

	<b>Statements</b>	<b>Yes</b>	<b>No</b>
<b>Correct Answer:</b>	A database named SalesOrders is created. The database will include two containers.	<input checked="" type="radio"/>	<input type="radio"/>
	Container1 will contain two items.	<input checked="" type="radio"/>	<input type="radio"/>
	Container2 will contain one item.	<input checked="" type="radio"/>	<input type="radio"/>

Box 1: Yes -

The createDatabaseIfNotExistsAsync method checks if a database exists, and if it doesn't, create it.

The Database.CreateContainerAsync method creates a container as an asynchronous operation in the Azure Cosmos service.

Box 2: Yes -

The CosmosContainer.CreateItemAsync method creates an item as an asynchronous operation in the Azure Cosmos service.

Box 3: Yes -

Reference:

<https://docs.microsoft.com/en-us/dotnet/api/microsoft.azure.cosmos.cosmosclient.createdatabaseifnotexistsasync>

<https://docs.microsoft.com/en-us/dotnet/api/microsoft.azure.cosmos.database.createcontainerasync> <https://docs.microsoft.com/en-us/dotnet/api/azure.cosmos.cosmoscontainer.createitemasync>

Question #21

Topic 2

**DRAG DROP -**

You develop an Azure solution that uses Cosmos DB.

The current Cosmos DB container must be replicated and must use a partition key that is optimized for queries.

You need to implement a change feed processor solution.

Which change feed processor components should you use? To answer, drag the appropriate components to the correct requirements. Each component 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 the content.

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

Select and Place:

**Components**

Host
Delegate
Lease container
Monitored container

**Answer Area****Requirement****Component**

Store the data from which the change feed is generated.

Component

Coordinate processing of the change feed across multiple workers.

Component

Use the change feed processor to listen for changes.

Component

Handle each batch of changes.

Component

**Correct Answer:****Components**


**Answer Area****Requirement****Component**

Store the data from which the change feed is generated.

Monitored container

Coordinate processing of the change feed across multiple workers.

Lease container

Use the change feed processor to listen for changes.

Host

Handle each batch of changes.

Delegate

Box 1: The monitored container -

The monitored container has the data from which the change feed is generated. Any inserts and updates to the monitored container are reflected in the change feed of the container.

Box 2: The lease container -

The lease container acts as a state storage and coordinates processing the change feed across multiple workers. The lease container can be stored in the same account as the monitored container or in a separate account.

Box 3: The host: A host is an application instance that uses the change feed processor to listen for changes. Multiple instances with the same lease configuration can run in parallel, but each instance should have a different instance name.

Box 4: The delegate -

The delegate is the code that defines what you, the developer, want to do with each batch of changes that the change feed processor reads.

Reference:

<https://docs.microsoft.com/en-us/azure/cosmos-db/change-feed-processor>

**Topic 3 - Question Set 3**

Question #1

DRAG DROP -

You develop a web application.

You need to register the application with an active Azure Active Directory (Azure AD) tenant.

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

Select and Place:

**Actions****Answer Area**

Select **Manifest** from the middle-tier service registration.

In Enterprise Applications, select **New application**.

Add a Cryptographic key.

Create a new application and provide the name, account type, and redirect URI.

Select the Azure AD instance.

Use an access token to access the secure resource.

In App Registrations, select **New registration**.



Correct Answer:

**Actions****Answer Area**

Select **Manifest** from the middle-tier service registration.

In App Registrations, select **New registration**.

In Enterprise Applications, select **New application**.

Select the Azure AD instance.

Add a Cryptographic key.

Create a new application and provide the name, account type, and redirect URI.



Select the Azure AD instance.

Use an access token to access the secure resource.

In App Registrations, select **New registration**.

Register a new application using the Azure portal

1. Sign in to the Azure portal using either a work or school account or a personal Microsoft account.

2. If your account gives you access to more than one tenant, select your account in the upper right corner. Set your portal session to the Azure AD tenant that you want.

3. Search for and select Azure Active Directory. Under Manage, select App registrations.
4. Select New registration. (Step 1)
5. In Register an application, enter a meaningful application name to display to users.
6. Specify who can use the application. Select the Azure AD instance. (Step 2)
7. Under Redirect URI (optional), select the type of app you're building: Web or Public client (mobile & desktop). Then enter the redirect URI, or reply URL, for your application. (Step 3)
8. When finished, select Register.

Question #2

Topic 3

You have a new Azure subscription. You are developing an internal website for employees to view sensitive data. The website uses Azure Active Directory (Azure AD) for authentication.

You need to implement multifactor authentication for the website.

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

NOTE: Each correct selection is worth one point.

- A. Configure the website to use Azure AD B2C.
- B. In Azure AD, create a new conditional access policy.
- C. Upgrade to Azure AD Premium.
- D. In Azure AD, enable application proxy.
- E. In Azure AD conditional access, enable the baseline policy.

**Correct Answer:** BC 

B: MFA Enabled by conditional access policy. It is the most flexible means to enable two-step verification for your users. Enabling using conditional access policy only works for Azure MFA in the cloud and is a premium feature of Azure AD.

C: Multi-Factor Authentication comes as part of the following offerings:

- Azure Active Directory Premium licenses - Full featured use of Azure Multi-Factor Authentication Service (Cloud) or Azure Multi-Factor Authentication Server (On-premises).
- Multi-Factor Authentication for Office 365
- Azure Active Directory Global Administrators

Reference:

<https://docs.microsoft.com/en-us/azure/active-directory/authentication/howto-mfa-getstarted>

## Question #3

## Topic 3

You are developing a Java application that uses Cassandra to store key and value data. You plan to use a new Azure Cosmos DB resource and the Cassandra API in the application. You create an Azure Active Directory (Azure AD) group named Cosmos DB Creators to enable provisioning of Azure Cosmos accounts, databases, and containers.

The Azure AD group must not be able to access the keys that are required to access the data.  
You need to restrict access to the Azure AD group.

Which role-based access control should you use?

- A. DocumentDB Accounts Contributor
- B. Cosmos Backup Operator
- C. Cosmos DB Operator
- D. Cosmos DB Account Reader

**Correct Answer:** C 

Azure Cosmos DB now provides a new RBAC role, Cosmos DB Operator. This new role lets you provision Azure Cosmos accounts, databases, and containers, but can't access the keys that are required to access the data. This role is intended for use in scenarios where the ability to grant access to Azure Active Directory service principals to manage deployment operations for Cosmos DB is needed, including the account, database, and containers.

Reference:

<https://azure.microsoft.com/en-us/updates/azure-cosmos-db-operator-role-for-role-based-access-control-rbac-is-now-available/>

## Question #4

## Topic 3

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 are developing a website that will run as an Azure Web App. Users will authenticate by using their Azure Active Directory (Azure AD) credentials.

You plan to assign users one of the following permission levels for the website: admin, normal, and reader. A user's Azure AD group membership must be used to determine the permission level.

You need to configure authorization.

Solution: Configure the Azure Web App for the website to allow only authenticated requests and require Azure AD log on.

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer:** B 

Instead in the Azure AD application's manifest, set value of the groupMembershipClaims option to All.

Reference:

<https://blogs.msdn.microsoft.com/waws/2017/03/13/azure-app-service-authentication-aad-groups/>

## Question #5

## Topic 3

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 are developing a website that will run as an Azure Web App. Users will authenticate by using their Azure Active Directory (Azure AD) credentials.

You plan to assign users one of the following permission levels for the website: admin, normal, and reader. A user's Azure AD group membership must be used to determine the permission level.

You need to configure authorization.

Solution:

- Create a new Azure AD application. In the application's manifest, set value of the groupMembershipClaims option to All.
- In the website, use the value of the groups claim from the JWT for the user to determine permissions.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer: A** 

To configure Manifest to include Group Claims in Auth Token

1. Go to Azure Active Directory to configure the Manifest. Click on Azure Active Directory, and go to App registrations to find your application:
2. Click on your application (or search for it if you have a lot of apps) and edit the Manifest by clicking on it.
3. Locate the `groupMembershipClaims` setting. Set its value to either `SecurityGroups` or `All`. To help you decide which:
  - `SecurityGroups` - groups claim will contain the identifiers of all security groups of which the user is a member.
  - `All` - groups claim will contain the identifiers of all security groups and all distribution lists of which the user is a member

Now your application will include group claims in your manifest and you can use this fact in your code.

Reference:

<https://blogs.msdn.microsoft.com/waws/2017/03/13/azure-app-service-authentication-aad-groups/>

## Question #6

## Topic 3

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 are developing a website that will run as an Azure Web App. Users will authenticate by using their Azure Active Directory (Azure AD) credentials.

You plan to assign users one of the following permission levels for the website: admin, normal, and reader. A user's Azure AD group membership must be used to determine the permission level.

You need to configure authorization.

Solution:

- Create a new Azure AD application. In the application's manifest, define application roles that match the required permission levels for the application.
- Assign the appropriate Azure AD group to each role. In the website, use the value of the roles claim from the JWT for the user to determine permissions.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer: B** 

To configure Manifest to include Group Claims in Auth Token

1. Go to Azure Active Directory to configure the Manifest. Click on Azure Active Directory, and go to App registrations to find your application:
2. Click on your application (or search for it if you have a lot of apps) and edit the Manifest by clicking on it.
3. Locate the `groupMembershipClaims` setting. Set its value to either `SecurityGroups` or `All`. To help you decide which:
  - `SecurityGroups` - groups claim will contain the identifiers of all security groups of which the user is a member.
  - `All` - groups claim will contain the identifiers of all security groups and all distribution lists of which the user is a member

Now your application will include group claims in your manifest and you can use this fact in your code.

Reference:

<https://blogs.msdn.microsoft.com/waws/2017/03/13/azure-app-service-authentication-aad-groups/>

Question #7

Topic 3

**DRAG DROP -**

You are developing an application to securely transfer data between on-premises file systems and Azure Blob storage. The application stores keys, secrets, and certificates in Azure Key Vault. The application uses the Azure Key Vault APIs.

The application must allow recovery of an accidental deletion of the key vault or key vault objects. Key vault objects must be retained for 90 days after deletion.

You need to protect the key vault and key vault objects.

Which Azure Key Vault feature should you use? To answer, drag the appropriate features to the correct actions. Each feature 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:

Features	Answer Area	Action	Feature
Access policy		Enable retention period and accidental deletion.	Feature
Purge protection		Enforce retention period and accidental deletion.	Feature
Soft delete			
Shared access signature			

**Correct Answer:**

Features	Answer Area	Action	Feature
Access policy		Enable retention period and accidental deletion.	Soft delete
Purge protection		Enforce retention period and accidental deletion.	Purge protection
Soft delete			
Shared access signature			

**Box 1: Soft delete -**

When soft-delete is enabled, resources marked as deleted resources are retained for a specified period (90 days by default). The service further provides a mechanism for recovering the deleted object, essentially undoing the deletion.

**Box 2: Purge protection -**

Purge protection is an optional Key Vault behavior and is not enabled by default. Purge protection can only be enabled once soft-delete is enabled.

When purge protection is on, a vault or an object in the deleted state cannot be purged until the retention period has passed. Soft-deleted vaults and objects can still be recovered, ensuring that the retention policy will be followed.

**Reference:**

<https://docs.microsoft.com/en-us/azure/key-vault/general/soft-delete-overview>

## Question #8

## Topic 3

You provide an Azure API Management managed web service to clients. The back-end web service implements HTTP Strict Transport Security (HSTS).

Every request to the backend service must include a valid HTTP authorization header.

You need to configure the Azure API Management instance with an authentication policy.

Which two policies can you use? Each correct answer presents a complete solution.

NOTE: Each correct selection is worth one point.

- A. Basic Authentication
- B. Digest Authentication
- C. Certificate Authentication
- D. OAuth Client Credential Grant

**Correct Answer:** CD 

Question #9

Topic 3

**DRAG DROP -**

You are developing an ASP.NET Core website that can be used to manage photographs which are stored in Azure Blob Storage containers.

Users of the website authenticate by using their Azure Active Directory (Azure AD) credentials.

You implement role-based access control (RBAC) role permissions on the containers that store photographs. You assign users to RBAC roles.

You need to configure the website's Azure AD Application so that user's permissions can be used with the Azure Blob containers.

How should you configure the application? To answer, drag the appropriate setting to the correct location. Each setting can 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:

<b>Settings</b>
client_id
profile
delegated
application
user_impersonation

**Answer Area**

<b>API</b>	<b>Permission</b>	<b>Type</b>
Azure Storage	Setting	Setting
Microsoft Graph	User.Read	Setting

Correct Answer:

<b>Settings</b>
client_id
profile
delegated
application
user_impersonation

**Answer Area**

<b>API</b>	<b>Permission</b>	<b>Type</b>
Azure Storage	user_impersonation	delegated
Microsoft Graph	User.Read	delegated

Box 1: user\_impersonation -

Box 2: delegated -

Example:

1. Select the API permissions section
2. Click the Add a permission button and then:  
Ensure that the My APIs tab is selected
3. In the list of APIs, select the API TodoListService-aspnetcore.
4. In the Delegated permissions section, ensure that the right permissions are checked: user\_impersonation.
5. Select the Add permissions button.

Box 3: delegated -

Example -

1. Select the API permissions section
2. Click the Add a permission button and then,  
Ensure that the Microsoft APIs tab is selected
3. In the Commonly used Microsoft APIs section, click on Microsoft Graph
4. In the Delegated permissions section, ensure that the right permissions are checked: User.Read. Use the search box if necessary.
5. Select the Add permissions button

Reference:

<https://docs.microsoft.com/en-us/samples/azure-samples/active-directory-dotnet-webapp-webapi-openidconnect-aspnetcore/calling-a-web-api-in-an-aspnet-core- web-application-using-azure-ad/>

Question #10

Topic 3

**HOTSPOT -**

You are developing an ASP.NET Core app that includes feature flags which are managed by Azure App Configuration store named AppFeatureFlagStore that contains a feature flag named Export.

You need to update the app to meet the following requirements:

- Use the Export feature in the app without requiring a restart of the app.
- Validate users before users are allowed access to secure resources.
- Permit users to access secure resources.

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

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

```
public void Configure(IApplicationBuilder app, IWebHostEnvironment env)
{
    if (env.IsDevelopment())
    {
        app.UseDeveloperExceptionPage();
    }
    else
    {
        app.UseExceptionHandler("/Error");
    }

    app. (); // Options: UseAuthentication, UseStaticFiles, UseSession, UseCookiePolicy

    app. (); // Options: UseAuthorization, UseHttpsRedirection, UseSession, UseCookiePolicy

    app. (); // Options: UseAzureAppConfiguration, UseRequestLocalization, UseCors, UseStaticFiles

    app.UseEndpoint(endpoints =>
    {
        endpoints.MapRazorPages();
    });
}
```

## Answer Area

```
public void Configure(IApplicationBuilder app, IWebHostEnvironment env)
{
    if (env.IsDevelopment())
    {
        app.UseDeveloperExceptionPage();
    }
    else
    {
        app.UseExceptionHandler("/Error");
    }

    app. ();
```

Correct Answer:

```
app. ();
```

```
app. ();
```

```
app. ();
```

```
app.UseEndpoint(endpoints =>
{
    endpoints.MapRazorPages();
});
```

Box 1: UseAuthentication -

Need to validate users before users are allowed access to secure resources.

UseAuthentication adds the AuthenticationMiddleware to the specified IApplicationBuilder, which enables authentication capabilities.

Box 2: UseAuthorization -

Need to permit users to access secure resources.

UseAuthorization adds the AuthorizationMiddleware to the specified IApplicationBuilder, which enables authorization capabilities.

Box 3: UseStaticFiles -

Need to use the Export feature in the app without requiring a restart of the app.

UseStaticFiles enables static file serving for the current request path

Reference:

<https://docs.microsoft.com/en-us/dotnet/api/microsoft.aspnetcore.builder.iapplicationbuilder?view=aspnetcore-5.0>

## Question #11

Topic 3

You have an application that includes an Azure Web app and several Azure Function apps. Application secrets including connection strings and certificates are stored in Azure Key Vault.

Secrets must not be stored in the application or application runtime environment. Changes to Azure Active Directory (Azure AD) must be minimized.

You need to design the approach to loading application secrets.

What should you do?

- A. Create a single user-assigned Managed Identity with permission to access Key Vault and configure each App Service to use that Managed Identity.
- B. Create a single Azure AD Service Principal with permission to access Key Vault and use a client secret from within the App Services to access Key Vault.
- C. Create a system assigned Managed Identity in each App Service with permission to access Key Vault.
- D. Create an Azure AD Service Principal with Permissions to access Key Vault for each App Service and use a certificate from within the App Services to access Key Vault.

**Correct Answer:** C 

Use Key Vault references for App Service and Azure Functions.

Key Vault references currently only support system-assigned managed identities. User-assigned identities cannot be used.

Reference:

<https://docs.microsoft.com/en-us/azure/app-service/app-service-key-vault-references>

## Question #12

Topic 3

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 are developing a medical records document management website. The website is used to store scanned copies of patient intake forms.

If the stored intake forms are downloaded from storage by a third party, the contents of the forms must not be compromised.

You need to store the intake forms according to the requirements.

Solution:

1. Create an Azure Key Vault key named skey.
2. Encrypt the intake forms using the public key portion of skey.
3. Store the encrypted data in Azure Blob storage.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer:** A 

## Question #13

Topic 3

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 are developing a medical records document management website. The website is used to store scanned copies of patient intake forms. If the stored intake forms are downloaded from storage by a third party, the contents of the forms must not be compromised.

You need to store the intake forms according to the requirements.

Solution:

1. Create an Azure Cosmos DB database with Storage Service Encryption enabled.
2. Store the intake forms in the Azure Cosmos DB database.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer: B** 

Instead use an Azure Key vault and public key encryption. Store the encrypted form in Azure Storage Blob storage.

## Question #14

Topic 3

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 are developing a medical records document management website. The website is used to store scanned copies of patient intake forms.

If the stored intake forms are downloaded from storage by a third party, the contents of the forms must not be compromised.

You need to store the intake forms according to the requirements.

Solution: Store the intake forms as Azure Key Vault secrets.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer: B** 

Instead use an Azure Key vault and public key encryption. Store the encrypted form in Azure Storage Blob storage.

## Question #15

## HOTSPOT -

You plan to deploy a new application to a Linux virtual machine (VM) that is hosted in Azure.

The entire VM must be secured at rest by using industry-standard encryption technology to address organizational security and compliance requirements.

You need to configure Azure Disk Encryption for the VM.

How should you complete the Azure CLI commands? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

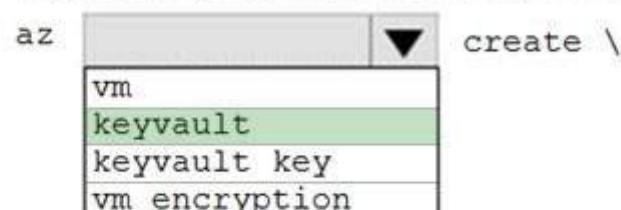
Hot Area:

**Answer Area**

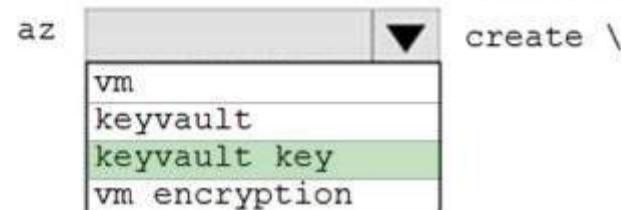
```
az provider register -n Microsoft.KeyVault  
resourcegroup="myResourceGroup"  
az group create --name $resourcegroup --location westus  
keyvault_name=myvaultname$RANDOM  
  
az [▼] create \  
  vm  
  keyvault  
  keyvault key  
  vm encryption  
  
  --name $keyvault_name \  
  --resource-group $resourcegroup \  
  --location eastus \  
  --enabled-for-disk-encryption True  
  
az [▼] create \  
  vm  
  keyvault  
  keyvault key  
  vm encryption  
  
  --vault-name $keyvault_name \  
  --name Name1 \  
  --protection software  
  
az [▼] create \  
  vm  
  keyvault  
  keyvault key  
  vm encryption  
  
  --resource-group $resourcegroup \  
  --name Name2 \  
  --image Canonical:UbuntuServer:16.04-LTS:latest \  
  --admin-username azureuser \  
  --generate-ssh-keys \  
  --data-disk-sizes-gb 5  
  
az [▼] enable\  
  vm  
  keyvault  
  keyvault key  
  vm encryption  
  
  --resource-group $resourcegroup \  
  --name Name2 \  
  --disk-encryption-keyvault $keyvault_name \  
  --key-encryption-key Name1 \  
  --volume-type [▼]  
    all  
    data  
    os
```

**Answer Area**

```
az provider register -n Microsoft.KeyVault
resourcegroup="myResourceGroup"
az group create --name $resourcegroup --location westus
keyvault_name=myvaultname$RANDOM
```



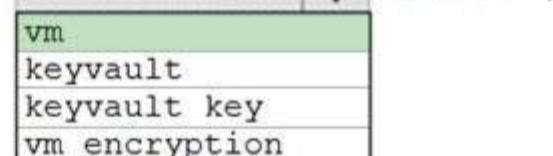
```
--name $keyvault_name \
--resource-group $resourcegroup \
--location eastus \
--enabled-for-disk-encryption True
```



```
--vault-name $keyvault_name \
--name Name1 \
--protection software
```

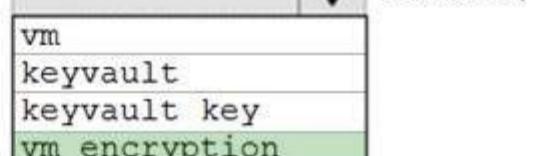
**Correct Answer:**

az [▼] create \

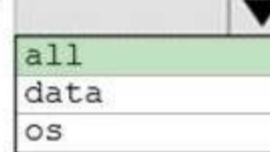


```
--resource-group $resourcegroup \
--name Name2 \
--image Canonical:UbuntuServer:16.04-LTS:latest \
--admin-username azureuser \
--generate-ssh-keys \
--data-disk-sizes-gb 5
```

az [▼] enable\



```
--resource-group $resourcegroup \
--name Name2 \
--disk-encryption-keyvault $keyvault_name \
--key-encryption-key Name1 \
--volume-type
```

**Box 1: keyvault -**

Create an Azure Key Vault with az keyvault create and enable the Key Vault for use with disk encryption. Specify a unique Key Vault name for keyvault\_name as follows: keyvault\_name=myvaultname\$RANDOM az keyvault create \

```
-name $keyvault_name \
--resource-group $resourcegroup \
--location eastus \
--enabled-for-disk-encryption True
```

**Box 2: keyvault key -**

The Azure platform needs to be granted access to request the cryptographic keys when the VM boots to decrypt the virtual disks. Create a cryptographic key in your Key Vault with az keyvault key create. The following example creates a key named myKey: az keyvault key create \

```
--vault-name $keyvault_name \
--name myKey \
--protection software
```

**Box 3: vm -**

Create a VM with az vm create. Only certain marketplace images support disk encryption. The following example creates a VM named myVM using an Ubuntu

16.04 LTS image:

```
az vm create \
--resource-group $resourcegroup \
--name myVM \
--image Canonical:UbuntuServer:16.04-LTS:latest \
--admin-username azureuser \
--generate-ssh-keys \
```

Box 4: vm encryption -

Encrypt your VM with az vm encryption enable:

```
az vm encryption enable \
--resource-group $resourcegroup \
--name myVM \
--disk-encryption-keyvault $keyvault_name \
--key-encryption-key myKey \
--volume-type all
```

Note: seems to an error in the question. Should have enable instead of create.

Box 5: all -

Encrypt both data and operating system.

Reference:

<https://docs.microsoft.com/en-us/azure/virtual-machines/linux/disk-encryption-cli-quickstart>

Question #16

Topic 3

Your company is developing an Azure API hosted in Azure.

You need to implement authentication for the Azure API to access other Azure resources. You have the following requirements:

- ☞ All API calls must be authenticated.
- ☞ Callers to the API must not send credentials to the API.

Which authentication mechanism should you use?

- A. Basic
- B. Anonymous
- C. Managed identity
- D. Client certificate

**Correct Answer:** C 

Azure Active Directory Managed Service Identity (MSI) gives your code an automatically managed identity for authenticating to Azure services, so that you can keep credentials out of your code.

Note: Use the authentication-managed-identity policy to authenticate with a backend service using the managed identity. This policy essentially uses the managed identity to obtain an access token from Azure Active Directory for accessing the specified resource. After successfully obtaining the token, the policy will set the value of the token in the Authorization header using the Bearer scheme.

Incorrect Answers:

A: Use the authentication-basic policy to authenticate with a backend service using Basic authentication. This policy effectively sets the HTTP Authorization header to the value corresponding to the credentials provided in the policy.

B: Anonymous is no authentication at all.

D: Your code needs credentials to authenticate to cloud services, but you want to limit the visibility of those credentials as much as possible. Ideally, they never appear on a developer's workstation or get checked-in to source control. Azure Key Vault can store credentials securely so they aren't in your code, but to retrieve them you need to authenticate to Azure Key Vault. To authenticate to Key Vault, you need a credential! A classic bootstrap problem.

Reference:

<https://azure.microsoft.com/en-us/blog/keep-credentials-out-of-code-introducing-azure-ad-managed-service-identity/>

<https://docs.microsoft.com/en-us/azure/api-management/api-management-authentication-policies>

Question #17

DRAG DROP -

You are developing an application. You have an Azure user account that has access to two subscriptions.

You need to retrieve a storage account key secret from Azure Key Vault.

In which order should you arrange the PowerShell commands to develop the solution? To answer, move all commands from the list of commands to the answer area and arrange them in the correct order.

Select and Place:

### Powershell commands

```
$secretvalue = ConvertTo-SecureString  
$storAcctkey -AsPlainText  
-Force  
    Set-AzKeyVaultSecret -VaultName  
$vaultName -Name $secretName  
-SecretValue $secretvalue
```

```
Get-AzStorageAccountKey -  
ResourceGroupName $resGroup -Name  
$storAcct
```

```
Set-AzContext -SubscriptionId  
$subscriptionID
```

```
Get-AzKeyVaultSecret -VaultName  
$vaultName
```

```
Get-AzSubscription
```

### Answer Area

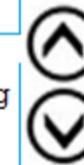


### Answer Area

```
Get-AzSubscription
```

```
Set-AzContext -SubscriptionId  
$subscriptionID
```

```
Get-AzStorageAccountKey -  
ResourceGroupName $resGroup -Name  
$storAcct
```



```
$secretvalue = ConvertTo-SecureString  
$storAcctkey -AsPlainText  
-Force  
    Set-AzKeyVaultSecret -VaultName  
$vaultName -Name $secretName  
-SecretValue $secretvalue
```

```
Get-AzKeyVaultSecret -VaultName  
$vaultName
```

Step 1: Get-AzSubscription -

If you have multiple subscriptions, you might have to specify the one that was used to create your key vault. Enter the following to see the subscriptions for your account:

Get-AzSubscription -

Step 2: Set-AzContext -SubscriptionId

To specify the subscription that's associated with the key vault you'll be logging, enter:

```
Set-AzContext -SubscriptionId <subscriptionID>
```

Step 3: Get-AzStorageAccountKey -

You must get that storage account key.

```
Step 4: $secretvalue = ConvertTo-SecureString <storageAccountKey> -AsPlainText -Force
```

```
Set-AzKeyVaultSecret -VaultName <vaultName> -Name <secretName> -SecretValue $secretvalue
```

After retrieving your secret (in this case, your storage account key), you must convert that key to a secure string, and then create a secret with that value in your key vault.

Step 5: Get-AzKeyVaultSecret -

Next, get the URI for the secret you created. You'll need this URI in a later step to call the key vault and retrieve your secret. Run the following PowerShell command and make note of the ID value, which is the secret's URI:

```
Get-AzKeyVaultSecret -VaultName <vaultName>
```

Reference:

<https://docs.microsoft.com/en-us/bs-latn-ba/Azure/key-vault/key-vault-key-rotation-log-monitoring>

#### Question #18

#### Topic 3

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 develop Azure solutions.

You must grant a virtual machine (VM) access to specific resource groups in Azure Resource Manager.

You need to obtain an Azure Resource Manager access token.

Solution: Use an X.509 certificate to authenticate the VM with Azure Resource Manager.

Does the solution meet the goal?

A. Yes

B. No

#### Correct Answer: B

Instead run the Invoke-RestMethod cmdlet to make a request to the local managed identity for Azure resources endpoint.

Reference:

<https://docs.microsoft.com/en-us/azure/active-directory/managed-identities-azure-resources/tutorial-windows-vm-access-arm>

Question #19

Topic 3

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 develop Azure solutions.

You must grant a virtual machine (VM) access to specific resource groups in Azure Resource Manager.

You need to obtain an Azure Resource Manager access token.

Solution: Use the Reader role-based access control (RBAC) role to authenticate the VM with Azure Resource Manager.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer:** *B* 

Instead run the Invoke-RestMethod cmdlet to make a request to the local managed identity for Azure resources endpoint.

Reference:

<https://docs.microsoft.com/en-us/azure/active-directory/managed-identities-azure-resources/tutorial-windows-vm-access-arm>

### Question #20

Topic 3

## HOTSPOT -

You are building a website that is used to review restaurants. The website will use an Azure CDN to improve performance and add functionality to requests.

You build and deploy a mobile app for Apple iPhones. Whenever a user accesses the website from an iPhone, the user must be redirected to the app store.

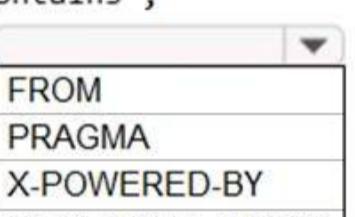
You need to implement an Azure CDN rule that ensures that iPhone users are redirected to the app store.

How should you complete the Azure Resource Manager template? To answer, select the appropriate options in the answer area.

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

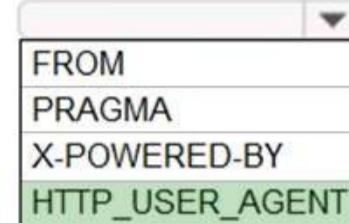
## Hot Area:

## Answer Area

```
"conditions": [ {
    "name": "IsDevice",
    "parameters": {
        "@odata.type": "#Microsoft.Azure.Cdn.Models."
    },
    "operator": "Equal",
    "matchValues": [ " "
        
    ],
    "rule": {
        "name": "DeliveryRuleIsDeviceConditionParameters"
    }
},
{
    "name": "RequestHeader",
    "parameters": {
        "@odata.type": "#Microsoft.Azure.Cdn.Models."
    },
    "operator": "Contains",
    "selector": " "
        
    ", " "
        
    ],
    "rule": {
        "name": "DeliveryRuleCookiesConditionParameters"
    }
}
]
```

**Correct Answer:****Answer Area**

```

"conditions": [ {
    "name": "IsDevice",
    "parameters": {
        "@odata.type": "#Microsoft.Azure.Cdn.Models."
        "operator": "Equal",
        "matchValues": [ " "
            
        ]
    }
},
{
    "name": "RequestHeader",
    "parameters": {
        "@odata.type": "#Microsoft.Azure.Cdn.Models."
        "operator": "Contains",
        "selector": " "
            
        "
    }
}
]

```

**Box 1: iOS -**

Azure AD Conditional Access supports the following device platforms:

- Android
- iOS
- Windows Phone
- Windows
- macOS
- 

**Box 2: DeliveryRuleIsDeviceConditionParameters**

The DeliveryRuleIsDeviceCondition defines the IsDevice condition for the delivery rule. parameters defines the parameters for the condition.

**Box 3: HTTP\_USER\_AGENT -**

Incorrect Answers:

- The Pragma HTTP/1.0 general header is an implementation-specific header that may have various effects along the request-response chain. It is used for backwards compatibility with HTTP/1.0 caches.
- "X-Powered-By" is a common non-standard HTTP response header (most headers prefixed with an 'X-' are non-standard).

**Box 4: DeliveryRuleRequestHeaderConditionParameters**

DeliveryRuleRequestHeaderCondition defines the RequestHeader condition for the delivery rule. parameters defines the parameters for the condition.

**Box 5: iOS -**

The Require approved client app requirement only supports the iOS and Android for device platform condition.

Reference:

<https://docs.microsoft.com/en-us/azure/active-directory/conditional-access/concept-conditional-access-conditions>

<https://docs.microsoft.com/en-us/azure/active-directory/conditional-access/concept-conditional-access-grant>

## Question #21

## Topic 3

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 are developing a website that will run as an Azure Web App. Users will authenticate by using their Azure Active Directory (Azure AD) credentials.

You plan to assign users one of the following permission levels for the website: admin, normal, and reader. A user's Azure AD group membership must be used to determine the permission level.

You need to configure authorization.

Solution:

- Configure and use Integrated Windows Authentication in the website.
- In the website, query Microsoft Graph API to load the group to which the user is a member.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer: B** 

Microsoft Graph is a RESTful web API that enables you to access Microsoft Cloud service resources.

Instead in the Azure AD application's manifest, set value of the groupMembershipClaims option to All. In the website, use the value of the groups claim from the

JWT for the user to determine permissions.

Reference:

<https://blogs.msdn.microsoft.com/waws/2017/03/13/azure-app-service-authentication-aad-groups/>

## Question #22

## Topic 3

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 develop Azure solutions.

You must grant a virtual machine (VM) access to specific resource groups in Azure Resource Manager.

You need to obtain an Azure Resource Manager access token.

Solution: Run the Invoke-RestMethod cmdlet to make a request to the local managed identity for Azure resources endpoint.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer: A** 

Get an access token using the VM's system-assigned managed identity and use it to call Azure Resource Manager

You will need to use PowerShell in this portion.

1. In the portal, navigate to Virtual Machines and go to your Windows virtual machine and in the Overview, click Connect.
2. Enter in your Username and Password for which you added when you created the Windows VM.
3. Now that you have created a Remote Desktop Connection with the virtual machine, open PowerShell in the remote session.
4. Using the Invoke-WebRequest cmdlet, make a request to the local managed identity for Azure resources endpoint to get an access token for Azure Resource

Manager.

Example:

```
$response = Invoke-WebRequest -Uri 'http://169.254.169.254/metadata/identity/oauth2/token?api-version=2018-02-01&resource=https://management.azure.com/' -Method GET -Headers @{Metadata="true"}
```

Reference:

<https://docs.microsoft.com/en-us/azure/active-directory/managed-identities-azure-resources/tutorial-windows-vm-access-arm>

Question #23

**HOTSPOT -**

You are building a website to access project data related to teams within your organization. The website does not allow anonymous access. Authentication is performed using an Azure Active Directory (Azure AD) app named internal.

The website has the following authentication requirements:

- Azure AD users must be able to login to the website.
- Personalization of the website must be based on membership in Active Directory groups.

You need to configure the application's manifest to meet the authentication requirements.

How should you configure the manifest? To answer, select the appropriate configuration in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

```
{
    ...
    "appId": "d61126e3-089b-4adb-b721-d5023213df7d",
    "displayName": "internal",
    "optionalClaims": {
        "groupMembershipClaims": "All"
    },
    "oauth2Permissions": [
        "allowPublicClient",
        "requiredResourceAccess"
    ],
    "oauth2AllowImplicitFlow": true
}
```

**Answer Area**

```
{
    ...
    "appId": "d61126e3-089b-4adb-b721-d5023213df7d",
    "displayName": "internal",
    "optionalClaims": {
        "groupMembershipClaims": "All"
    },
    "oauth2Permissions": [
        "allowPublicClient",
        "requiredResourceAccess"
    ],
    "oauth2AllowImplicitFlow": true
}
```

Box 1: groupMembershipClaims -

Scenario: Personalization of the website must be based on membership in Active Directory groups.

Group claims can also be configured in the Optional Claims section of the Application Manifest.

Enable group membership claims by changing the groupMembershipClaim

The valid values are:

- "All"
- "SecurityGroup"
- "DistributionList"
- "DirectoryRole"

Box 2: oauth2Permissions -

Scenario: Azure AD users must be able to login to the website. oauth2Permissions specifies the collection of OAuth 2.0 permission scopes that the web API (resource) app exposes to client apps. These permission scopes may be granted to client apps during consent.

Incorrect Answers:

oauth2AllowImplicitFlow. oauth2AllowImplicitFlow specifies whether this web app can request OAuth2.0 implicit flow access tokens. The

default is false. This flag is used for browser-based apps, like Javascript single-page apps.

Reference:

<https://docs.microsoft.com/en-us/azure/active-directory/hybrid/how-to-connect-fed-group-claims>

Question #24

Topic 3

You develop an app that allows users to upload photos and videos to Azure storage. The app uses a storage REST API call to upload the media to a blob storage account named Account1. You have blob storage containers named Container1 and Container2.

Uploading of videos occurs on an irregular basis.

You need to copy specific blobs from Container1 to Container2 when a new video is uploaded.

What should you do?

- A. Copy blobs to Container2 by using the Put Blob operation of the Blob Service REST API
- B. Create an Event Grid topic that uses the Start-AzureStorageBlobCopy cmdlet
- C. Use AzCopy with the Snapshot switch to copy blobs to Container2
- D. Download the blob to a virtual machine and then upload the blob to Container2

**Correct Answer: B**

The Start-AzureStorageBlobCopy cmdlet starts to copy a blob.

Example 1: Copy a named blob -

```
C:\PS>Start-AzureStorageBlobCopy -SrcBlob "ContosoPlanning2015" -DestContainer "ContosoArchives" -SrcContainer "ContosoUploads"
```

This command starts the copy operation of the blob named ContosoPlanning2015 from the container named ContosoUploads to the container named

ContosoArchives.

Reference:

<https://docs.microsoft.com/en-us/powershell/module/azure.storage/start-azurestorageblobcopy?view=azurermps-6.13.0>

Question #25

Topic 3

You are developing an ASP.NET Core website that uses Azure FrontDoor. The website is used to build custom weather data sets for researchers.

Data sets are downloaded by users as Comma Separated Value (CSV) files. The data is refreshed every 10 hours.

Specific files must be purged from the FrontDoor cache based upon Response Header values.

You need to purge individual assets from the Front Door cache.

Which type of cache purge should you use?

- A. single path
- B. wildcard
- C. root domain

**Correct Answer: A**

These formats are supported in the lists of paths to purge:

⊖ Single path purge: Purge individual assets by specifying the full path of the asset (without the protocol and domain), with the file extension, for example, /

[1]

⊖ Wildcard purge: Asterisk (\*) may be used as a wildcard. Purge all folders, subfolders, and files under an endpoint with /\* in the path or purge all subfolders and files under a specific folder by specifying the folder followed by /\*, for example, /pictures/\*.

⊖ Root domain purge: Purge the root of the endpoint with "/" in the path.

Reference:

<https://docs.microsoft.com/en-us/azure/frontdoor/front-door-caching>

## Question #26

Topic 3

Your company is developing an Azure API.

You need to implement authentication for the Azure API. You have the following requirements:

All API calls must be secure.

- ☞ Callers to the API must not send credentials to the API.

Which authentication mechanism should you use?

- A. Basic
- B. Anonymous
- C. Managed identity
- D. Client certificate

**Correct Answer:** C 

Use the authentication-managed-identity policy to authenticate with a backend service using the managed identity of the API Management service. This policy essentially uses the managed identity to obtain an access token from Azure Active Directory for accessing the specified resource. After successfully obtaining the token, the policy will set the value of the token in the Authorization header using the Bearer scheme.

Reference:

<https://docs.microsoft.com/bs-cyrl-ba/azure/api-management/api-management-authentication-policies>

## Question #27

Topic 3

You are a developer for a SaaS company that offers many web services.

All web services for the company must meet the following requirements:

- ☞ Use API Management to access the services
- ☞ Use OpenID Connect for authentication
- ☞ Prevent anonymous usage

A recent security audit found that several web services can be called without any authentication.

Which API Management policy should you implement?

- A. jsonp
- B. authentication-certificate
- C. check-header
- D. validate-jwt

**Correct Answer:** D 

Add the validate-jwt policy to validate the OAuth token for every incoming request.

Incorrect Answers:

A: The jsonp policy adds JSON with padding (JSONP) support to an operation or an API to allow cross-domain calls from JavaScript browser-based clients.

JSONP is a method used in JavaScript programs to request data from a server in a different domain. JSONP bypasses the limitation enforced by most web browsers where access to web pages must be in the same domain.

JSONP - Adds JSON with padding (JSONP) support to an operation or an API to allow cross-domain calls from JavaScript browser-based clients.

Reference:

<https://docs.microsoft.com/en-us/azure/api-management/api-management-howto-protect-backend-with-aad>

Question #28

Topic 3

**DRAG DROP -**

Contoso, Ltd. provides an API to customers by using Azure API Management (APIM). The API authorizes users with a JWT token.

You must implement response caching for the APIM gateway. The caching mechanism must detect the user ID of the client that accesses data for a given location and cache the response for that user ID.

You need to add the following policies to the policies file:

- a set-variable policy to store the detected user identity
- a cache-lookup-value policy
- a cache-store-value policy
- a find-and-replace policy to update the response body with the user profile information

To which policy section should you add the policies? To answer, drag the appropriate sections to the correct policies. Each section 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**

Policy section	Policy	Policy section
	Set-variable	
Inbound	Cache-lookup-value	
Outbound	Cache-store-value	
	Find-and-replace	

**Answer Area**

Policy section	Policy	Policy section
	Set-variable	Inbound
Inbound	Cache-lookup-value	Inbound
Outbound	Cache-store-value	Outbound
	Find-and-replace	Outbound

Box 1: Inbound.

A set-variable policy to store the detected user identity.

Example:

```
<policies>
<inbound>
<!-- How you determine user identity is application dependent -->
<set-variable
name="enduserid"
value="@(<context>.Request.Headers.GetValueOrDefault("Authorization","").Split(' ')[1].AsJwt()?.Subject)" />
```

Box 2: Inbound -

A cache-lookup-value policy -

Example:

```
<inbound>
<base />
<cache-lookup vary-by-developer="true | false" vary-by-developer-groups="true | false" downstream-caching-type="none | private | public" must-revalidate="true | false">
<vary-by-query-parameter>parameter name</vary-by-query-parameter> <!-- optional, can repeated several times -->
</cache-lookup>
</inbound>
```

Box 3: Outbound -

A cache-store-value policy.

Example:

```
<outbound>
<base />
<cache-store duration="3600" />
</outbound>
```

Box 4: Outbound -

A find-and-replace policy to update the response body with the user profile information.

Example:

```
<outbound>
<!-- Update response body with user profile-->
<find-and-replace
from="$userprofile$"
to="@((string)context.Variables["userprofile"])" />
<base />
</outbound>
```

Reference:

<https://docs.microsoft.com/en-us/azure/api-management/api-management-caching-policies> <https://docs.microsoft.com/en-us/azure/api-management/api-management-sample-cache-by-key>

Question #29

Topic 3

**DRAG DROP -**

You are developing an Azure solution.

You need to develop code to access a secret stored in Azure Key Vault.

How should you complete the code segment? To answer, drag the appropriate code segments 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:

**Code segments**

DefaultAzureCredential  
ClientSecretCredential  
CloudClients  
SecretClient

**Answer Area**

```
string var1 = Environment.GetEnvironmentVariable("KEY_VAULT_URI");
var var2 = new Code segment (new Uri(var1), new Code segment());
```

**Correct Answer:****Code segments**

**ClientSecretCredential**  
CloudClients

**Answer Area**

```
string var1 = Environment.GetEnvironmentVariable("KEY_VAULT_URI");
var var2 = new SecretClient (new Uri(var1), new DefaultAzureCredential());
```

Box 1: SecretClient -

Box 2: DefaultAzureCredential -

In below example, the name of your key vault is expanded to the key vault URI, in the format "https://<your-key-vault-name>.vault.azure.net".

This example is using 'DefaultAzureCredential()' class from Azure Identity Library, which allows to use the same code across different environments with different options to provide identity.

```
string keyVaultName = Environment.GetEnvironmentVariable("KEY_VAULT_NAME");
var kvUri = "https://" + keyVaultName + ".vault.azure.net";
var client = new SecretClient(new Uri(kvUri), new DefaultAzureCredential());
```

Reference:

<https://docs.microsoft.com/en-us/azure/key-vault/secrets/quick-create-net>

## Question #30

Topic 3

You are developing an Azure App Service REST API.

The API must be called by an Azure App Service web app. The API must retrieve and update user profile information stored in Azure Active Directory (Azure AD).

You need to configure the API to make the updates.

Which two tools should you use? Each correct answer presents part of the solution.

NOTE: Each correct selection is worth one point.

- A. Microsoft Graph API
- B. Microsoft Authentication Library (MSAL)
- C. Azure API Management
- D. Microsoft Azure Security Center
- E. Microsoft Azure Key Vault SDK

**Correct Answer:** AC 

A: You can use the Azure AD REST APIs in Microsoft Graph to create unique workflows between Azure AD resources and third-party services. Enterprise developers use Microsoft Graph to integrate Azure AD identity management and other services to automate administrative workflows, such as employee onboarding (and termination), profile maintenance, license deployment, and more.

C: API Management (APIM) is a way to create consistent and modern API gateways for existing back-end services.

API Management helps organizations publish APIs to external, partner, and internal developers to unlock the potential of their data and services.

Reference:

<https://docs.microsoft.com/en-us/graph/azuread-identity-access-management-concept-overview>

## Question #31

Topic 3

You develop a REST API. You implement a user delegation SAS token to communicate with Azure Blob storage.

The token is compromised.

You need to revoke the token.

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

NOTE: Each correct selection is worth one point.

- A. Revoke the delegation key.
- B. Delete the stored access policy.
- C. Regenerate the account key.
- D. Remove the role assignment for the security principle.

**Correct Answer:** AB 

A: Revoke a user delegation SAS -

To revoke a user delegation SAS from the Azure CLI, call the `az storage account revoke-delegation-keys` command. This command revokes all of the user delegation keys associated with the specified storage account. Any shared access signatures associated with those keys are invalidated.

B: To revoke a stored access policy, you can either delete it, or rename it by changing the signed identifier. Changing the signed identifier breaks the associations between any existing signatures and the stored access policy. Deleting or renaming the stored access policy immediately effects all of the shared access signatures associated with it.

Reference:

<https://github.com/MicrosoftDocs/azure-docs/blob/master/articles/storage/blobs/storage-blob-user-delegation-sas-create-cli.md>

<https://docs.microsoft.com/en-us/rest/api/storageservices/define-stored-access-policy#modifying-or-revoking-a-stored-access-policy>

Question #32

Topic 3

**DRAG DROP -**

You are developing an Azure-hosted application that must use an on-premises hardware security module (HSM) key.

The key must be transferred to your existing Azure Key Vault by using the Bring Your Own Key (BYOK) process.

You need to securely transfer the key to Azure Key Vault.

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**

Generate a key transfer blob file by using the HSM vendor-provided tool.

Generate a Key Exchange Key (KEK).

Create a custom policy definition in Azure Policy.

Run the `az keyvault key import` command.

Run the `az keyvault key restore` command.

Retrieve the Key Exchange Key (KEK) public key.

**Answer Area****Correct Answer:****Actions**

Create a custom policy definition in Azure Policy.

Run the `az keyvault key restore` command.

**Answer Area**

Generate a Key Exchange Key (KEK).

Retrieve the Key Exchange Key (KEK) public key.

Generate a key transfer blob file by using the HSM vendor-provided tool.

Run the `az keyvault key import` command.

To perform a key transfer, a user performs following steps:

- Generate KEK.
- Retrieve the public key of the KEK.
- Using HSM vendor provided BYOK tool - Import the KEK into the target HSM and exports the Target Key protected by the KEK.
- Import the protected Target Key to Azure Key Vault.

Step 1: Generate a Key Exchange Key (KEK).

Step 2: Retrieve the Key Exchange Key (KEK) public key.

Step 3: Generate a key transfer blob file by using the HSM vendor-provided tool.

Generate key transfer blob using HSM vendor provided BYOK tool

Step 4: Run the `az keyvault key import` command

Upload key transfer blob to import HSM-key.

Customer will transfer the Key Transfer Blob ("byok" file) to an online workstation and then run a `az keyvault key import` command to import this blob as a new

HSM-backed key into Key Vault.

To import an RSA key use this command:

`az keyvault key import`

Reference:

<https://docs.microsoft.com/en-us/azure/key-vault/keys/byok-specification>

Question #33

Topic 3

You develop and deploy an Azure Logic app that calls an Azure Function app. The Azure Function app includes an OpenAPI (Swagger) definition and uses an Azure Blob storage account. All resources are secured by using Azure Active Directory (Azure AD). The Azure Logic app must securely access the Azure Blob storage account. Azure AD resources must remain if the Azure Logic app is deleted. You need to secure the Azure Logic app.

What should you do?

- A. Create a user-assigned managed identity and assign role-based access controls.
- B. Create an Azure AD custom role and assign the role to the Azure Blob storage account.
- C. Create an Azure Key Vault and issue a client certificate.
- D. Create a system-assigned managed identity and issue a client certificate.
- E. Create an Azure AD custom role and assign role-based access controls.

**Correct Answer:** A 

To give a managed identity access to an Azure resource, you need to add a role to the target resource for that identity.

Note: To easily authenticate access to other resources that are protected by Azure Active Directory (Azure AD) without having to sign in and provide credentials or secrets, your logic app can use a managed identity (formerly known as Managed Service Identity or MSI). Azure manages this identity for you and helps secure your credentials because you don't have to provide or rotate secrets.

If you set up your logic app to use the system-assigned identity or a manually created, user-assigned identity, the function in your logic app can also use that same identity for authentication.

Reference:

<https://docs.microsoft.com/en-us/azure/logic-apps/create-managed-service-identity> <https://docs.microsoft.com/en-us/azure/api-management/api-management-howto-mutual-certificates-for-clients>

Question #34

**HOTSPOT -**

You are developing an application that uses a premium block blob storage account. You are optimizing costs by automating Azure Blob Storage access tiers.

You apply the following policy rules to the storage account. You must determine the implications of applying the rules to the data. (Line numbers are included for reference only.)

```

01 {
02   "rules": [
03     {
04       "name": "agingDataRule",
05       "enabled": true,
06       "type": "Lifecycle",
07       "definition": {
08         "filters": [
09           "blobTypes": [ "blockBlob" ],
10           "prefixMatch": [ "container1/salesorders", "container2/inventory" ]
11         ],
12         "actions": [
13           "baseBlob": [
14             "tierToCool": { "daysAfterModificationGreaterThan": 60 },
15             "tierToArchive": { "daysAfterModificationGreaterThan": 120 }
16           ]
17         }
18       }
19     },
20     {
21       "enabled": true,
22       "name": "lastAccessedDataRule",
23       "type": "Lifecycle",
24       "definition": [
25         "actions": [
26           "baseBlob": [
27             "enableAutoTierToHotFromCool": true,
28             "tierToCool": [
29               "daysAfterLastAccessTimeGreaterThan": 30
30             ]
31           ]
32         ],
33         "filters": [
34           "blobTypes": [ "blockBlob" ]
35         ]
36       }
37     },
38     {
39       "rules": [
40         {
41           "name": "expirationDataRule",
42           "enabled": true,
43           "type": "Lifecycle",
44           "definition": [
45             "filters": [
46               "blobTypes": [ "blockBlob" ]
47             ],
48             "actions": [
49               "baseBlob": [
50                 "delete": { "daysAfterModificationGreaterThan": 730 }
51               ]
52             ]
53           ]
54         }
55       ]
56     }
57   ]
58 }
```

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

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

Yes	No
<input type="radio"/>	<input type="radio"/>

Block blobs prefixed with container1/salesorders or container2/inventory which have not been modified in over 60 days are moved to cool storage. Blobs that have not been modified in 120 days are moved to the archive tier.

Blobs are moved to cool storage if they have not been accessed for 30 days.

<input type="radio"/>	<input type="radio"/>
-----------------------	-----------------------

Blobs will automatically be tiered from cool back to hot if accessed again after being tiered to cool.

<input type="radio"/>	<input type="radio"/>
-----------------------	-----------------------

All block blobs older than 730 days will be deleted.

<input type="radio"/>	<input type="radio"/>
-----------------------	-----------------------

**Correct Answer:****Answer Area****Yes      No**

Block blobs prefixed with container1/salesorders or container2/inventory which have not been modified in over 60 days are moved to cool storage. Blobs that have not been modified in 120 days are moved to the archive tier.

Blobs are moved to cool storage if they have not been accessed for 30 days.

Blobs will automatically be tiered from cool back to hot if accessed again after being tiered to cool.

All block blobs older than 730 days will be deleted.

Box 1: Yes -

```
"rules": [
  {
    "name": "agingDataRule",
    "enabled": true,
    "type": "Lifecycle",
    "definition": {
      "filters": {
        "blobTypes": [ "blockBlob" ],
        "prefixMatch": [ "container1/salesorders", "container2/inventory" ]
      },
      "actions": {
        "baseBlob": {
          "tierToCool": { "daysAfterModificationGreaterThan": 60 },
          "tierToArchive": { "daysAfterModificationGreaterThan": 120 }
        }
      }
    }
  }
]
```

Box 2: Yes -

```
"enabled": true,
"name": "lastAccessedDataRule",
"type": "Lifecycle",
"definition": {
  "actions": {
    "baseBlob": {
      "enableAutoTierToHotFromCool": true,
      "tierToCool": {
        "daysAfterLastAccessTimeGreaterThan": 30
      }
    }
  }
}
```

Box 3: Yes -

Box 4: Yes -

```
"rules": [
  {
    "name": "expirationDataRule",
    "enabled": true,
    "type": "Lifecycle",
    "definition": {
      "filters": {
        "blobTypes": [ "blockBlob" ]
      },
      "actions": {
        "baseBlob": {
          "delete": { "daysAfterModificationGreaterThan": 730 }
        }
      }
    }
  }
]
```

## Question #35

*Topic 3*

You are developing a solution that will use a multi-partitioned Azure Cosmos DB database. You plan to use the latest Azure Cosmos DB SDK for development.

The solution must meet the following requirements:

- Send insert and update operations to an Azure Blob storage account.
- Process changes to all partitions immediately.
- Allow parallelization of change processing.

You need to process the Azure Cosmos DB operations.

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

NOTE: Each correct selection is worth one point.

- A. Create an Azure App Service API and implement the change feed estimator of the SDK. Scale the API by using multiple Azure App Service instances.
- B. Create a background job in an Azure Kubernetes Service and implement the change feed feature of the SDK.
- C. Create an Azure Function to use a trigger for Azure Cosmos DB. Configure the trigger to connect to the container.
- D. Create an Azure Function that uses a FeedIterator object that processes the change feed by using the pull model on the container. Use a FeedRange object to parallelize the processing of the change feed across multiple functions.

**Correct Answer: C** 

Azure Functions is the simplest option if you are just getting started using the change feed. Due to its simplicity, it is also the recommended option for most change feed use cases. When you create an Azure Functions trigger for Azure Cosmos DB, you select the container to connect, and the Azure Function gets triggered whenever there is a change in the container. Because Azure Functions uses the change feed processor behind the scenes, it automatically parallelizes change processing across your container's partitions.

Note: You can work with change feed using the following options:

- Using change feed with Azure Functions
- Using change feed with change feed processor

Reference:

<https://docs.microsoft.com/en-us/azure/cosmos-db/read-change-feed>

Question #36

Topic 3

**HOTSPOT -**

You have an Azure Web app that uses Cosmos DB as a data store. You create a CosmosDB container by running the following PowerShell script:

```
$resourceGroupName = "testResourceGroup"
$accountName = "testCosmosAccount"
$databaseName = "testDatabase"
$containerName = "testContainer"
$partitionKeyPath = "/EmployeeId"
$autoscaleMaxThroughput = 5000
```

```
New-AzCosmosDBSqlContainer -
-ResourceGroupName $resourceGroupName
-AccountName $accountName
-DatabaseName $databaseName
-Name $containerName
-PartitionKeyKind Hash
-PartitionKeyPath $partitionKeyPath
-AutoscaleMaxThroughput $autoscaleMaxThroughput
```

You create the following queries that target the container:

```
SELECT * FROM c WHERE c.EmployeeId > '12345'
```

```
SELECT * FROM c WHERE c.UserId = '12345'
```

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

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

	<b>Yes</b>	<b>No</b>
The minimum throughput for the container is 400 R/Us.	<input type="radio"/>	<input type="radio"/>
The first query statement is an in-partition query.	<input type="radio"/>	<input type="radio"/>
The second query statement is a cross-partition query.	<input type="radio"/>	<input type="radio"/>

Correct Answer:

**Answer Area**

	<b>Yes</b>	<b>No</b>
The minimum throughput for the container is 400 R/Us.	<input type="radio"/>	<input checked="" type="radio"/>
The first query statement is an in-partition query.	<input type="radio"/>	<input checked="" type="radio"/>
The second query statement is a cross-partition query.	<input checked="" type="radio"/>	<input type="radio"/>

Box 1: No -

You set the highest, or maximum RU/s Tmax you don't want the system to exceed. The system automatically scales the throughput T such that  $0.1 * Tmax \leq T \leq Tmax$ .

In this example we have autoscaleMaxThroughput = 5000, so the minimum throughput for the container is 500 R/Us.

Box 2: No -

First query: `SELECT * FROM c WHERE c.EmployeeId > '12345'`

Here's a query that has a range filter on the partition key and won't be scoped to a single physical partition. In order to be an in-partition query, the query must have an equality filter that includes the partition key:

```
SELECT * FROM c WHERE c.DeviceId > 'XMS-0001'
```

Box 3: Yes -

Example of In-partition query:

Consider the below query with an equality filter on Deviceld. If we run this query on a container partitioned on Deviceld, this query will filter to a single physical partition.

```
SELECT * FROM c WHERE c.Deviceld = 'XMS-0001'
```

Reference:

<https://docs.microsoft.com/en-us/azure/cosmos-db/how-to-choose-offer> <https://docs.microsoft.com/en-us/azure/cosmos-db/how-to-query-container>

Question #37

Topic 3

**HOTSPOT -**

You are developing a web application that makes calls to the Microsoft Graph API. You register the application in the Azure portal and upload a valid X509 certificate.

You create an appsettings.json file containing the certificate name, client identifier for the application, and the tenant identifier of the Azure Active Directory (Azure

AD). You create a method named ReadCertificate to return the X509 certificate by name.

You need to implement code that acquires a token by using the certificate.

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

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

```
AuthenticationConfig config = AuthenticationConfig.ReadFromJsonFile("appsettings.json");
X509Certificate2 certificate = ReadCertificate(config.CertificateName);
var app =  .Create(config.ClientId)


ConfidentialClientApplicationBuilder
GetAccountAsync()
GetAccountsAsync()
ConfidentialClientApplication

.WithCertificate(certificate)
.WithAuthority(new Uri(config.Authority))
.Build();

string[] scopes = new string[] { $"{config.ApiUrl}.default" };
AuthenticationResult result = await app.AcquireTokenForClient( .ExecuteAsync();


scopes
app
config
```

**Correct Answer:****Answer Area**

```
AuthenticationConfig config = AuthenticationConfig.ReadFromJsonFile("appsettings.json");
X509Certificate2 certificate = ReadCertificate(config.CertificateName);
var app =  .Create(config.ClientId)


ConfidentialClientApplicationBuilder selected
GetAccountAsync()
GetAccountsAsync()
ConfidentialClientApplication

.WithCertificate(certificate)
.WithAuthority(new Uri(config.Authority))
.Build();

string[] scopes = new string[] { $"{config.ApiUrl}.default" };
AuthenticationResult result = await app.AcquireTokenForClient( .ExecuteAsync();


scopes selected
app
config
```

Box 1: ConfidentialClientApplicationBuilder

Here's the code to instantiate the confidential client application with a client secret: app =

ConfidentialClientApplicationBuilder.Create(config.ClientId)

.WithClientSecret(config.ClientSecret)

.WithAuthority(new Uri(config.Authority))

.Build();

Box 2: scopes -

After you've constructed a confidential client application, you can acquire a token for the app by calling AcquireTokenForClient, passing the scope, and optionally forcing a refresh of the token.

Sample code: result = await app.AcquireTokenForClient(scopes)

.ExecuteAsync();

Reference:

<https://docs.microsoft.com/en-us/azure/active-directory/develop/scenario-daemon-app-configuration> <https://docs.microsoft.com/en-us/azure/active-directory/develop/scenario-daemon-acquire-token>

**Topic 4 - Question Set 4**

Question #1

*Topic 4*

You are developing an ASP.NET Core Web API web service. The web service uses Azure Application Insights for all telemetry and dependency tracking. The web service reads and writes data to a database other than Microsoft SQL Server.

You need to ensure that dependency tracking works for calls to the third-party database.

Which two dependency telemetry properties should you use? Each correct answer presents part of the solution.

NOTE: Each correct selection is worth one point.

A. Telemetry.Context.Cloud.RoleInstance

B. Telemetry.Id

C. Telemetry.Name

D. Telemetry.Context.Operation.Id

E. Telemetry.Context.Session.Id

**Correct Answer:** *BD* 

Example:

```
public async Task Enqueue(string payload)
{
    // StartOperation is a helper method that initializes the telemetry item
    // and allows correlation of this operation with its parent and children. var operation = telemetryClient.StartOperation<DependencyTelemetry>
    ("enqueue " + queueName);

    operation.Telemetry.Type = "Azure Service Bus";
    operation.Telemetry.Data = "Enqueue " + queueName;
    var message = new BrokeredMessage(payload);
    // Service Bus queue allows the property bag to pass along with the message.
    // We will use them to pass our correlation identifiers (and other context)
    // to the consumer.
    message.Properties.Add("ParentId", operation.Telemetry.Id);
    message.Properties.Add("RootId", operation.Telemetry.Context.Operation.Id);
```

Reference:

<https://docs.microsoft.com/en-us/azure/azure-monitor/app/custom-operations-tracking>

Question #2

Topic 4

**HOTSPOT -**

You are using Azure Front Door Service.

You are expecting inbound files to be compressed by using Brotli compression. You discover that inbound XML files are not compressed. The files are 9 megabytes (MB) in size.

You need to determine the root cause for the issue.

To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

Statement	Yes	No
-----------	-----	----

The file MIME type is supported by the service.

Edge nodes must be purged of all cache assets.

The compression type is supported.

**Answer Area**

Statement	Yes	No
-----------	-----	----

Correct Answer: The file MIME type is supported by the service.

Edge nodes must be purged of all cache assets.

The compression type is supported.

Box 1: No -

Front Door can dynamically compress content on the edge, resulting in a smaller and faster response to your clients. All files are eligible for compression.

However, a file must be of a MIME type that is eligible for compression list.

Box 2: No -

Sometimes you may wish to purge cached content from all edge nodes and force them all to retrieve new updated assets. This might be due to updates to your web application, or to quickly update assets that contain incorrect information.

Box 3: Yes -

These profiles support the following compression encodings: Gzip (GNU zip), Brotli

Reference:

<https://docs.microsoft.com/en-us/azure/frontdoor/front-door-caching>



Question #3

Topic 4

**HOTSPOT -**

You are developing an Azure App Service hosted ASP.NET Core web app to deliver video-on-demand streaming media. You enable an Azure Content Delivery Network (CDN) Standard for the web endpoint. Customer videos are downloaded from the web app by using the following example URL:

<http://www.contoso.com/content.mp4?quality=1>

All media content must expire from the cache after one hour. Customer videos with varying quality must be delivered to the closest regional point of presence (POP) node.

You need to configure Azure CDN caching rules.

Which options should you use? To answer, select the appropriate options in the answer area.

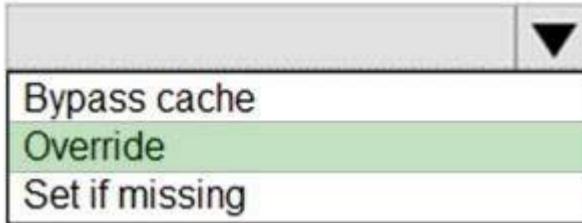
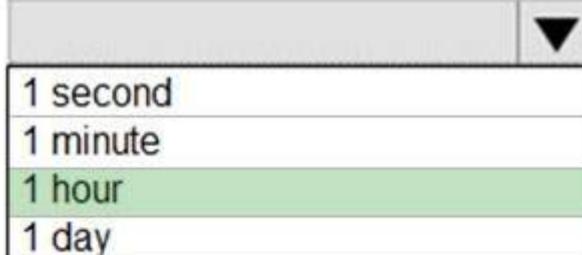
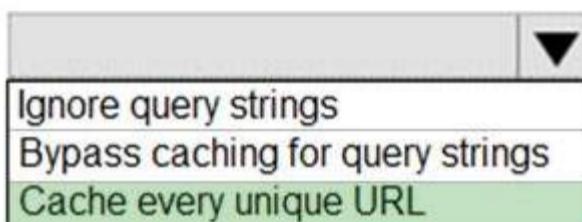
NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

Setting	Action
Caching behavior	<div style="border: 1px solid black; padding: 5px; display: inline-block;"><p>Bypass cache</p><p>Override</p><p>Set if missing</p></div>
Cache expiration duration	<div style="border: 1px solid black; padding: 5px; display: inline-block;"><p>1 second</p><p>1 minute</p><p>1 hour</p><p>1 day</p></div>
Query string caching behavior	<div style="border: 1px solid black; padding: 5px; display: inline-block;"><p>Ignore query strings</p><p>Bypass caching for query strings</p><p>Cache every unique URL</p></div>

## Answer Area

Setting	Action
Caching behavior	
Correct Answer: Cache expiration duration	
Query string caching behavior	

Box 1: Override -

Override: Ignore origin-provided cache duration; use the provided cache duration instead. This will not override cache-control: no-cache.

Set if missing: Honor origin-provided cache-directive headers, if they exist; otherwise, use the provided cache duration.

Incorrect:

Bypass cache: Do not cache and ignore origin-provided cache-directive headers.

Box 2: 1 hour -

All media content must expire from the cache after one hour.

Box 3: Cache every unique URL -

Cache every unique URL: In this mode, each request with a unique URL, including the query string, is treated as a unique asset with its own cache. For example, the response from the origin server for a request for example.ashx?q=test1 is cached at the POP node and returned for subsequent caches with the same query string. A request for example.ashx?q=test2 is cached as a separate asset with its own time-to-live setting.

Incorrect Answers:

Bypass caching for query strings: In this mode, requests with query strings are not cached at the CDN POP node. The POP node retrieves the asset directly from the origin server and passes it to the requestor with each request.

Ignore query strings: Default mode. In this mode, the CDN point-of-presence (POP) node passes the query strings from the requestor to the origin server on the first request and caches the asset. All subsequent requests for the asset that are served from the POP ignore the query strings until the cached asset expires.

Reference:

<https://docs.microsoft.com/en-us/azure/cdn/cdn-query-string>

## Question #4

## DRAG DROP -

You develop a web app that uses tier D1 app service plan by using the Web Apps feature of Microsoft Azure App Service.

Spikes in traffic have caused increases in page load times.

You need to ensure that the web app automatically scales when CPU load is about 85 percent and minimize costs.

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.

NOTE: More than one order of answer choices is correct. You will receive credit for any of the correct orders you select.

Select and Place:

**Actions****Answer Area**

Configure the web app to the Premium App Service tier.

Configure the web app to the Standard App Service tier.

Enable autoscaling on the web app.

Add a Scale rule.

Switch to an Azure App Services consumption plan.

Configure a Scale condition.

**Actions****Answer Area**

Configure the web app to the Premium App Service tier.

Configure the web app to the Standard App Service tier.

Enable autoscaling on the web app.

Add a Scale rule.

Switch to an Azure App Services consumption plan.

Configure a Scale condition.



Correct Answer:

Step 1: Configure the web app to the Standard App Service Tier

The Standard tier supports auto-scaling, and we should minimize the cost.

Step 2: Enable autoscaling on the web app

First enable autoscale -

Step 3: Add a scale rule -

Step 4: Add a Scale condition -

Reference:

<https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-autoscale-get-started>

Question #5

Topic 4

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution.

Determine whether the solution meets the stated goals.

You are developing and deploying several ASP.NET web applications to Azure App Service. You plan to save session state information and HTML output.

You must use a storage mechanism with the following requirements:

- Share session state across all ASP.NET web applications.
- Support controlled, concurrent access to the same session state data for multiple readers and a single writer.
- Save full HTTP responses for concurrent requests.

You need to store the information.

Proposed Solution: Enable Application Request Routing (ARR).

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer: B** 

Instead deploy and configure Azure Cache for Redis. Update the web applications.

Reference:

<https://docs.microsoft.com/en-us/azure/architecture/best-practices/caching#managing-concurrency-in-a-cache>

## Question #6

## Topic 4

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution.

Determine whether the solution meets the stated goals.

You are developing and deploying several ASP.NET web applications to Azure App Service. You plan to save session state information and HTML output.

You must use a storage mechanism with the following requirements:

- Share session state across all ASP.NET web applications.
- Support controlled, concurrent access to the same session state data for multiple readers and a single writer.
- Save full HTTP responses for concurrent requests.

You need to store the information.

Proposed Solution: Deploy and configure an Azure Database for PostgreSQL. Update the web applications.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer: B** 

Instead deploy and configure Azure Cache for Redis. Update the web applications.

Reference:

<https://docs.microsoft.com/en-us/azure/architecture/best-practices/caching#managing-concurrency-in-a-cache>

Question #7

Topic 4

**HOTSPOT -**

A company is developing a gaming platform. Users can join teams to play online and see leaderboards that include player statistics. The solution includes an entity named Team.

You plan to implement an Azure Redis Cache instance to improve the efficiency of data operations for entities that rarely change.

You need to invalidate the cache when team data is changed.

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

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

```
void ClearCachedTeams()  
{  
    IDatabase cache = Connection.GetDatabase();  
    ICache cache = Connection.GetDatabase();  
  
    cache.KeyDelete("Team");  
    cache.StringSet("Team", "");  
    cache.ValueDelete("Team");  
    cache.StringGet("Team", "");  
  
    ViewBag.msg += "Team data removed from cache.";  
}
```

**Answer Area**

```
void ClearCachedTeams()  
{  
    IDatabase cache = Connection.GetDatabase();  
    ICache cache = Connection.GetDatabase();  
  
    cache.KeyDelete("Team");  
    cache.StringSet("Team", "");  
    cache.ValueDelete("Team");  
    cache.StringGet("Team", "");  
  
    ViewBag.msg += "Team data removed from cache.";  
}
```

Correct Answer:

Box 1: IDatabase cache = connection.GetDatabase();

Connection refers to a previously configured ConnectionMultiplexer.

Box 2: cache.StringSet("teams", "")

To specify the expiration of an item in the cache, use the TimeSpan parameter of StringSet. cache.StringSet("key1", "value1", TimeSpan.FromMinutes(90));

Reference:

<https://azure.microsoft.com/sv-se/blog/lap-around-azure-redis-cache-preview/> <https://docs.microsoft.com/en-us/cli/azure/webapp/config/container>



## Question #8

## DRAG DROP -

A company has multiple warehouses. Each warehouse contains IoT temperature devices which deliver temperature data to an Azure Service Bus queue.

You need to send email alerts to facility supervisors immediately if the temperature at a warehouse goes above or below specified threshold temperatures.

Which five 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**

Add a logic app trigger that fires when one or more messages arrive in the queue.

Add a Recurrence trigger that schedules the app to run every 15 minutes.

Add an action that sends an email to specified personnel if the temperature is outside of those thresholds.

Add a trigger that reads IoT temperature data from a Service Bus queue.

Add a logic app action that fires when one or more messages arrive in the queue.

Add a condition that compares the temperature against the upper and lower thresholds.

Create a blank Logic app.

Add an action that reads IoT temperature data from the Service Bus queue.

**Answer Area**

## Correct Answer:

**Actions**

Add a logic app trigger that fires when one or more messages arrive in the queue.

Add a Recurrence trigger that schedules the app to run every 15 minutes.

Add an action that sends an email to specified personnel if the temperature is outside of those thresholds.

Add a trigger that reads IoT temperature data from a Service Bus queue.

Add a logic app action that fires when one or more messages arrive in the queue.

Add a condition that compares the temperature against the upper and lower thresholds.

Create a blank Logic app.

Add an action that reads IoT temperature data from the Service Bus queue.

**Answer Area**

Create a blank Logic app.

Add a logic app action that fires when one or more messages arrive in the queue.

Add an action that reads IoT temperature data from the Service Bus queue.

Add a condition that compares the temperature against the upper and lower thresholds.

Add an action that sends an email to specified personnel if the temperature is outside of those thresholds.

Step 1: Create a blank Logic app.

Create and configure a Logic App.

Step 2: Add a logical app trigger that fires when one or more messages arrive in the queue.

Configure the logic app trigger.

Under Triggers, select When one or more messages arrive in a queue (auto-complete).

Step 3: Add an action that reads IoT temperature data from the Service Bus queue

Step 4: Add a condition that compares the temperature against the upper and lower thresholds.

Step 5: Add an action that sends an email to specified personnel if the temperature is outside of those thresholds

Reference:

<https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-monitoring-notifications-with-azure-logic-apps>

Question #9

Topic 4

**DRAG DROP -**

You develop an ASP.NET Core MVC application. You configure the application to track webpages and custom events.

You need to identify trends in application usage.

Which Azure Application Insights Usage Analysis features should you use? To answer, drag the appropriate features to the correct requirements.

Each feature 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:

<b>Features</b>	<b>Answer Area</b>	<b>Requirement</b>	<b>Feature</b>
Users		Which pages visited by users most often correlate to a product purchase?	Feature
Funnels		How does load time of the product display page affect a user's decision to purchase a product?	Feature
Impact		Which events most influence a user's decision to continue to use the application?	Feature
Retention		Are there places in the application that users often perform repetitive actions?	Feature
User Flows			

**Correct Answer:**

<b>Features</b>	<b>Answer Area</b>	<b>Requirement</b>	<b>Feature</b>
Users		Which pages visited by users most often correlate to a product purchase?	Users
Funnels		How does load time of the product display page affect a user's decision to purchase a product?	Impact
Impact		Which events most influence a user's decision to continue to use the application?	Retention
Retention		Are there places in the application that users often perform repetitive actions?	User Flows
User Flows			

Box 1: Users -

Box 2: Impact -

One way to think of Impact is as the ultimate tool for settling arguments with someone on your team about how slowness in some aspect of your site is affecting whether users stick around. While users may tolerate a certain amount of slowness, Impact gives you insight into how best to balance optimization and performance to maximize user conversion.

Box 3: Retention -

The retention feature in Azure Application Insights helps you analyze how many users return to your app, and how often they perform particular tasks or achieve goals. For example, if you run a game site, you could compare the numbers of users who return to the site after losing a game with the number who return after winning. This knowledge can help you improve both your user experience and your business strategy.

Box 4: User flows -

The User Flows tool visualizes how users navigate between the pages and features of your site. It's great for answering questions like:

- How do users navigate away from a page on your site?
- What do users click on a page on your site?
- Where are the places that users churn most from your site?
- Are there places where users repeat the same action over and over?

Incorrect Answers:

Funnel: If your application involves multiple stages, you need to know if most customers are progressing through the entire process, or if they are ending the process at some point. The progression through a series of steps in a web application is known as a funnel. You can use Azure Application Insights Funnels to gain insights into your users, and monitor step-by-step conversion rates.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-monitor/app/usage-impact>

Question #10

Topic 4

You develop a gateway solution for a public facing news API. The news API back end is implemented as a RESTful service and uses an OpenAPI specification.

You need to ensure that you can access the news API by using an Azure API Management service instance.

Which Azure PowerShell command should you run?

- A. Import-AzureRmApiManagementApi -Context \$ApiMgmtContext -SpecificationFormat "Swagger" -SpecificationPath \$SwaggerPath -Path \$Path
- B. New-AzureRmApiManagementBackend -Context \$ApiMgmtContext-Url \$Url -Protocol http
- C. New-AzureRmApiManagement -ResourceGroupName \$ResourceGroup -Name \$Name `"-Location \$Location -Organization \$Org - AdminEmail \$AdminEmail
- D. New-AzureRmApiManagementBackendProxy -Url \$ApiUrl

**Correct Answer:** D 

New-AzureRmApiManagementBackendProxy creates a new Backend Proxy Object which can be piped when creating a new Backend entity.

Example: Create a Backend Proxy In-Memory Object

```
PS C:\>$secpassword = ConvertTo-SecureString "PlainTextPassword" -AsPlainText -Force  
PS C:\>$proxyCreds = New-Object System.Management.Automation.PSCredential ("foo", $secpassword)  
PS C:\>$credential = New-AzureRmApiManagementBackendProxy -Url "http://12.168.1.1:8080" -ProxyCredential $proxyCreds  
PS C:\>$apimContext = New-AzureRmApiManagementContext -ResourceGroupName "Api-Default-WestUS" -ServiceName "contoso"  
PS C:\>$backend = New-AzureRmApiManagementBackend -Context $apimContext -BackendId 123 -Url 'https://contoso.com/awesomeapi' -  
Protocol http -Title  
"first backend" -SkipCertificateChainValidation $true -Proxy $credential -Description "backend with proxy server"
```

Creates a Backend Proxy Object and sets up Backend

Incorrect Answers:

- A: The Import-AzureRmApiManagementApi cmdlet imports an Azure API Management API from a file or a URL in Web Application Description Language (WADL), Web Services Description Language (WSDL), or Swagger format.
- B: New-AzureRmApiManagementBackend creates a new backend entity in Api Management.
- C: The New-AzureRmApiManagement cmdlet creates an API Management deployment in Azure API Management.

Reference:

<https://docs.microsoft.com/en-us/powershell/module/azurerm.apimanagement/new-azurermapimanagementbackendproxy?view=azurermps-6.13.0>

## Question #11

## Topic 4

You are creating a hazard notification system that has a single signaling server which triggers audio and visual alarms to start and stop. You implement Azure Service Bus to publish alarms. Each alarm controller uses Azure Service Bus to receive alarm signals as part of a transaction. Alarm events must be recorded for audit purposes. Each transaction record must include information about the alarm type that was activated.

You need to implement a reply trail auditing solution.

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

NOTE: Each correct selection is worth one point.

- A. Assign the value of the hazard message SessionId property to the ReplyToSessionId property.
- B. Assign the value of the hazard message MessageId property to the DeliveryCount property.
- C. Assign the value of the hazard message SessionId property to the SequenceNumber property.
- D. Assign the value of the hazard message MessageId property to the CorrelationId property.
- E. Assign the value of the hazard message SequenceNumber property to the DeliveryCount property.
- F. Assign the value of the hazard message MessageId property to the SequenceNumber property.

**Correct Answer:** AD 

D: CorrelationId: Enables an application to specify a context for the message for the purposes of correlation; for example, reflecting the MessageId of a message that is being replied to.

A: ReplyToSessionId: This value augments the ReplyTo information and specifies which SessionId should be set for the reply when sent to the reply entity.

Incorrect Answers:

B, E: DeliveryCount -

Number of deliveries that have been attempted for this message. The count is incremented when a message lock expires, or the message is explicitly abandoned by the receiver. This property is read-only.

C, F: SequenceNumber -

The sequence number is a unique 64-bit integer assigned to a message as it is accepted and stored by the broker and functions as its true identifier. For partitioned entities, the topmost 16 bits reflect the partition identifier. Sequence numbers monotonically increase and are gapless. They roll over to 0 when the 48-

64 bit range is exhausted. This property is read-only.

Reference:

<https://docs.microsoft.com/en-us/azure/service-bus-messaging/service-bus-messages-payloads>

## Question #12

## Topic 4

You are developing an Azure function that connects to an Azure SQL Database instance. The function is triggered by an Azure Storage queue.

You receive reports of numerous System.InvalidOperationExceptions with the following message:

‘Timeout expired. The timeout period elapsed prior to obtaining a connection from the pool. This may have occurred because all pooled connections were in use and max pool size was reached.’

You need to prevent the exception.

What should you do?

- A. In the host.json file, decrease the value of the batchSize option
- B. Convert the trigger to Azure Event Hub
- C. Convert the Azure Function to the Premium plan
- D. In the function.json file, change the value of the type option to queueScaling

**Correct Answer:** C 

With the Premium plan the max outbound connections per instance is unbounded compared to the 600 active (1200 total) in a Consumption plan.

Note: The number of available connections is limited partly because a function app runs in a sandbox environment. One of the restrictions that the sandbox imposes on your code is a limit on the number of outbound connections, which is currently 600 active (1,200 total) connections per instance. When you reach this limit, the functions runtime writes the following message to the logs: Host thresholds exceeded: Connections.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-functions/manage-connections> <https://docs.microsoft.com/en-us/azure/azure-functions/functions-scale#service-limits>

## Question #13

## Topic 4

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution.

Determine whether the solution meets the stated goals.

You are developing and deploying several ASP.NET web applications to Azure App Service. You plan to save session state information and HTML output.

You must use a storage mechanism with the following requirements:

- Share session state across all ASP.NET web applications.
- Support controlled, concurrent access to the same session state data for multiple readers and a single writer.
- Save full HTTP responses for concurrent requests.

You need to store the information.

Proposed Solution: Deploy and configure Azure Cache for Redis. Update the web applications.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer: A** 

The session state provider for Azure Cache for Redis enables you to share session information between different instances of an ASP.NET web application.

The same connection can be used by multiple concurrent threads.

Redis supports both read and write operations.

The output cache provider for Azure Cache for Redis enables you to save the HTTP responses generated by an ASP.NET web application.

Note: Using the Azure portal, you can also configure the eviction policy of the cache, and control access to the cache by adding users to the roles provided. These roles, which define the operations that members can perform, include Owner, Contributor, and Reader. For example, members of the Owner role have complete control over the cache (including security) and its contents, members of the Contributor role can read and write information in the cache, and members of the

Reader role can only retrieve data from the cache.

Reference:

<https://docs.microsoft.com/en-us/azure/architecture/best-practices/caching>

Question #14

Topic 4

**HOTSPOT -**

You are debugging an application that is running on Azure Kubernetes cluster named cluster1. The cluster uses Azure Monitor for containers to monitor the cluster.

The application has sticky sessions enabled on the ingress controller.

Some customers report a large number of errors in the application over the last 24 hours.

You need to determine on which virtual machines (VMs) the errors are occurring.

How should you complete the Azure Monitor query? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

```
let startTimestamp =   
    ago(1d)  
    since(1d)  
    totimespan(1d)  
    date(now() - 1d);  
  
let ContainerIDs = KubePodInventory  
| where ClusterName == "Cluster1"  
|   
    top ContainerID  
    union ContainerID  
    sample ContainerID  
    distinct ContainerID;  
  
ContainerLog  
|   
    fork containerIDs  
    where ContainerID in (ContainerIDs)  
    restrict ContainerID in (ContainerIDs)  
    join ContainerID == ContainerIDs.ContainerID  
| where TimeGenerated > startTimestamp  
| where LogEntrySource == "stderr"  
|   
    project by Computer  
    summarize by Computer  
    partition count() by Computer  
    summarize count() by Computer;
```

**Answer Area**

```
let startTimestamp =  ;  
| ago(1d)  
| since(1d)  
| totimespan(1d)  
| date(now() - 1d)  
  
let ContainerIDs = KubePodInventory  
| where ClusterName == "Cluster1"  
|  ;  
| top ContainerID  
| union ContainerID  
| sample ContainerID  
| distinct ContainerID  
  
ContainerLog  
|  ;  
| fork containerIDs  
| where ContainerID in (ContainerIDs)  
| restrict ContainerID in (ContainerIDs)  
| join ContainerID == ContainerIDs.ContainerID  
|  ;  
| where TimeGenerated > startTimestamp  
| where LogEntrySource == "stderr"  
|  ;  
| project by Computer  
| summarize by Computer  
| partition count() by Computer  
| summarize count() by Computer
```

**Correct Answer:**

Box 1: ago(1d)

Box 2: distinct containerID -

Box 3: where ContainerID in (ContainerIDs)

Box 4: summarize Count by Computer

Summarize: aggregate groups of rows

Use summarize to identify groups of records, according to one or more columns, and apply aggregations to them. The most common use of summarize is count, which returns the number of results in each group.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-monitor/log-query/get-started-queries> <https://docs.microsoft.com/en-us/azure/azure-monitor/log-query/query-optimization>

Question #15

Topic 4

**HOTSPOT -**

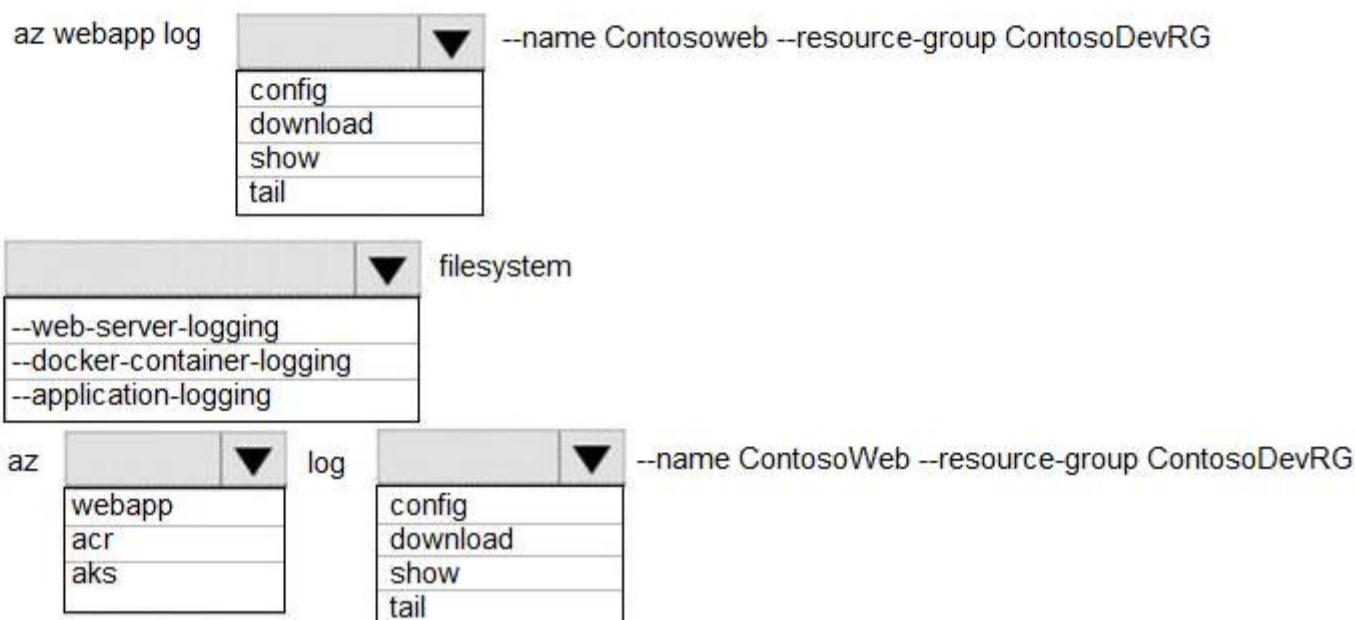
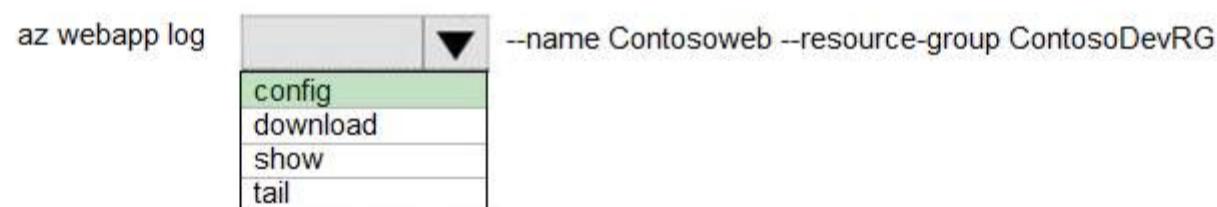
You plan to deploy a web app to App Service on Linux. You create an App Service plan. You create and push a custom Docker image that contains the web app to Azure Container Registry.

You need to access the console logs generated from inside the container in real-time.

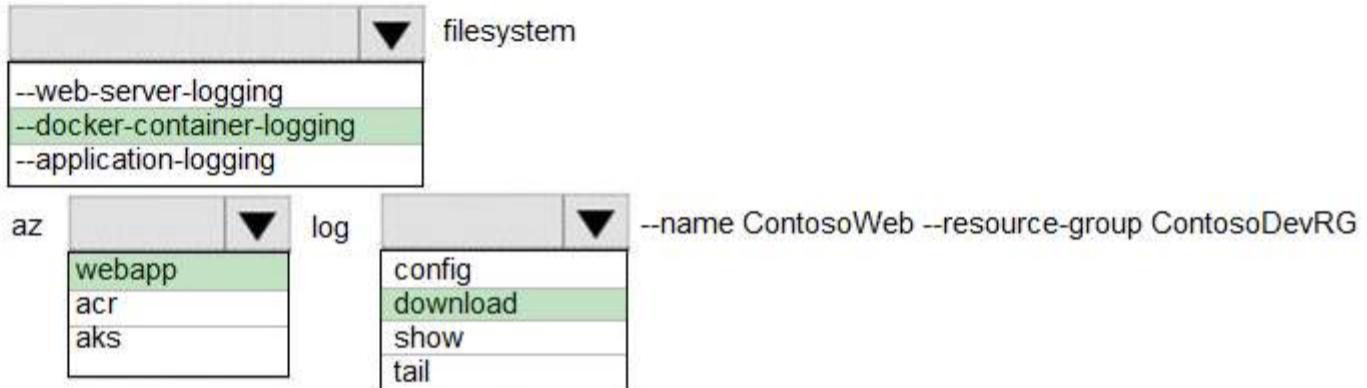
How should you complete the Azure CLI command? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area****Answer Area**

**Correct Answer:**



Box 1: config -

To Configure logging for a web app use the command:

az webapp log config

Box 2: --docker-container-logging

Syntax include:

az webapp log config [-docker-container-logging {filesystem, off}]

Box 3: webapp -

To download a web app's log history as a zip file use the command: az webapp log download

Box 4: download -

Reference:

<https://docs.microsoft.com/en-us/cli/azure/webapp/log>

## Question #16

## Topic 4

You develop and deploy an ASP.NET web app to Azure App Service. You use Application Insights telemetry to monitor the app. You must test the app to ensure that the app is available and responsive from various points around the world and at regular intervals. If the app is not responding, you must send an alert to support staff. You need to configure a test for the web app. Which two test types can you use? Each correct answer presents a complete solution.

NOTE: Each correct selection is worth one point.

- A. integration
- B. multi-step web
- C. URL ping
- D. unit
- E. load

**Correct Answer:** BC 

There are three types of availability tests:

- ⇒ URL ping test: a simple test that you can create in the Azure portal.
- ⇒ Multi-step web test: A recording of a sequence of web requests, which can be played back to test more complex scenarios. Multi-step web tests are created in Visual Studio Enterprise and uploaded to the portal for execution.
- ⇒ Custom Track Availability Tests: If you decide to create a custom application to run availability tests, the `TrackAvailability()` method can be used to send the results to Application Insights.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-monitor/app/monitor-web-app-availability>

Question #17

Topic 4

**DRAG DROP -**

A web service provides customer summary information for e-commerce partners. The web service is implemented as an Azure Function app with an HTTP trigger.

Access to the API is provided by an Azure API Management instance. The API Management instance is configured in consumption plan mode. All API calls are authenticated by using OAuth.

API calls must be cached. Customers must not be able to view cached data for other customers.

You need to configure API Management policies for caching.

How should you complete the policy statement?

Select and Place:

Targets	Answer Area			
Expect	<policies>			
Public	<inbound>			
Private	<base />			
Internal	<cache-lookup caching-type="	Target	" downstream-caching-type = "	Target
External	</cache-lookup>			
Authorization	<vary-by-header>	Target	</vary-by-header>	
	</cache-lookup>			
	</inbound>			
	</policies>			

**Correct Answer:**

Targets	Answer Area			
Expect	<policies>			
Public	<inbound>			
Private	<base />			
Internal	<cache-lookup caching-type="	Internal	" downstream-caching-type = "	Private
External	</cache-lookup>			
Authorization	<vary-by-header>	Authorization	</vary-by-header>	
	</cache-lookup>			
	</inbound>			
	</policies>			

Box 1: internal -

caching-type

Choose between the following values of the attribute:

- internal to use the built-in API Management cache,
- external to use the external cache as Azure Cache for Redis prefer-external to use external cache if configured or internal cache otherwise.

▪

Box 2: private -

downstream-caching-type

This attribute must be set to one of the following values.

- none - downstream caching is not allowed.
- private - downstream private caching is allowed.
- public - private and shared downstream caching is allowed.

Box 3: Authorization -

<vary-by-header>Authorization</vary-by-header>

<!-- should be present when allow-private-response-caching is "true"-->

Note: Start caching responses per value of specified header, such as Accept, Accept-Charset, Accept-Encoding, Accept-Language, Authorization, Expect, From,

Host, If-Match -

Reference:

<https://docs.microsoft.com/en-us/azure/api-management/api-management-caching-policies>

Question #18

Topic 4

You are developing applications for a company. You plan to host the applications on Azure App Services.

The company has the following requirements:

- Every five minutes verify that the websites are responsive.
- Verify that the websites respond within a specified time threshold. Dependent requests such as images and JavaScript files must load properly.
- Generate alerts if a website is experiencing issues.
- If a website fails to load, the system must attempt to reload the site three more times.

You need to implement this process with the least amount of effort.

What should you do?

- A. Create a Selenium web test and configure it to run from your workstation as a scheduled task.
- B. Set up a URL ping test to query the home page.
- C. Create an Azure function to query the home page.
- D. Create a multi-step web test to query the home page.
- E. Create a Custom Track Availability Test to query the home page.

**Correct Answer: D** 

You can monitor a recorded sequence of URLs and interactions with a website via multi-step web tests.

Incorrect Answers:

A: Selenium is an umbrella project for a range of tools and libraries that enable and support the automation of web browsers.

It provides extensions to emulate user interaction with browsers, a distribution server for scaling browser allocation, and the infrastructure for implementations of the W3C WebDriver specification that lets you write interchangeable code for all major web browsers.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-monitor/app/availability-multistep>

Question #19

Topic 4

You develop and add several functions to an Azure Function app that uses the latest runtime host. The functions contain several REST API endpoints secured by using SSL. The Azure Function app runs in a Consumption plan.

You must send an alert when any of the function endpoints are unavailable or responding too slowly.

You need to monitor the availability and responsiveness of the functions.

What should you do?

- A. Create a URL ping test.
- B. Create a timer triggered function that calls `TrackAvailability()` and send the results to Application Insights.
- C. Create a timer triggered function that calls `GetMetric("Request Size")` and send the results to Application Insights.
- D. Add a new diagnostic setting to the Azure Function app. Enable the `FunctionAppLogs` and `Send to Log Analytics` options.

**Correct Answer:** *B* 

You can create an Azure Function with `TrackAvailability()` that will run periodically according to the configuration given in `TimerTrigger` function with your own business logic. The results of this test will be sent to your Application Insights resource, where you will be able to query for and alert on the availability results data.

This allows you to create customized tests similar to what you can do via Availability Monitoring in the portal. Customized tests will allow you to write more complex availability tests than is possible using the portal UI, monitor an app inside of your Azure VNET, change the endpoint address, or create an availability test even if this feature is not available in your region.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-monitor/app/availability-azure-functions>

## Question #20

## DRAG DROP -

You are developing an application to retrieve user profile information. The application will use the Microsoft Graph SDK.

The app must retrieve user profile information by using a Microsoft Graph API call.

You need to call the Microsoft Graph API from the application.

In which order should you perform the actions? To answer, move all actions from the list of actions to the answer area and arrange them in the correct order.

Select and Place:

- | Actions  |
|--|
| Create an authentication provider.                             |
| Create a new instance of the GraphServiceClient.               |
| Invoke the request to the Microsoft Graph API.                 |
| Register the application with the Microsoft identity platform. |
| Build a client by using the client app ID.                     |

## Answer Area



## Correct Answer:

- | Actions |
|---------|
|         |
|         |
|         |
|         |

## Answer Area

- | Answer Area  |
|--|
| Register the application with the Microsoft identity platform. |
| Build a client by using the client app ID.                     |
| >Create an authentication provider.                            |
| Create a new instance of the GraphServiceClient.               |
| Invoke the request to the Microsoft Graph API.                 |

Step 1: Register the application with the Microsoft identity platform.

To authenticate with the Microsoft identity platform endpoint, you must first register your app at the Azure app registration portal

Step 2: Build a client by using the client app ID

Step 3: Create an authentication provider

Create an authentication provider by passing in a client application and graph scopes.

Code example:

```
DeviceCodeProvider authProvider = new DeviceCodeProvider(publicClientApplication, graphScopes);
```

```
// Create a new instance of GraphServiceClient with the authentication provider.
```

```
GraphServiceClient graphClient = new GraphServiceClient(authProvider);
```

Step 4: Create a new instance of the GraphServiceClient

Step 5: Invoke the request to the Microsoft Graph API

Reference:

<https://docs.microsoft.com/en-us/graph/auth-v2-service>

<https://docs.microsoft.com/en-us/graph/sdks/create-client>

Question #21

Topic 4

**DRAG DROP -**

You develop and deploy an Azure Logic App that calls an Azure Function app. The Azure Function App includes an OpenAPI (Swagger) definition and uses an

Azure Blob storage account. All resources are secured by using Azure Active Directory (Azure AD).

The Logic App must use Azure Monitor logs to record and store information about runtime data and events. The logs must be stored in the Azure Blob storage account.

You need to set up Azure Monitor logs and collect diagnostics data for the Azure Logic App.

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 action groups and alert rules. Create a Log Analytics workspace. Install the Logic Apps Management solution. Add a diagnostic setting to the Azure Function App. Create an Azure storage account. Add a diagnostic setting to the Azure Logic App.**Answer Area****Correct Answer:****Actions** Create action groups and alert rules. Add a diagnostic setting to the Azure Function App. Create an Azure storage account.**Answer Area** Create a Log Analytics workspace. Install the Logic Apps Management solution. Add a diagnostic setting to the Azure Logic App.

Step 1: Create a Log Analytics workspace

Before you start, you need a Log Analytics workspace.

Step 2: Install the Logic Apps Management solution

To set up logging for your logic app, you can enable Log Analytics when you create your logic app, or you can install the Logic Apps Management solution in your

Log Analytics workspace for existing logic apps.

Step 3: Add a diagnostic setting to the Azure Logic App

Set up Azure Monitor logs -

1. In the Azure portal, find and select your logic app.
2. On your logic app menu, under Monitoring, select Diagnostic settings > Add diagnostic setting.

Reference:

<https://docs.microsoft.com/en-us/azure/logic-apps/monitor-logic-apps-log-analytics>

## Question #22

## DRAG DROP -

You develop an application. You plan to host the application on a set of virtual machines (VMs) in Azure.

You need to configure Azure Monitor to collect logs from the 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**

Create a Log Analytics workspace.

Install agents on the VM and VM scale set to be monitored.

Send console logs.

Add a VMInsights solution.

Create an Application Insights resource.

**Answer Area****Correct Answer:****Actions**

Send console logs.

**Answer Area**

Create a Log Analytics workspace.

Add a VMInsights solution.

Install agents on the VM and VM scale set to be monitored.

Create an Application Insights resource.

Step 1: Create a Log Analytics workspace.

First create the workspace.

Step 2: Add a VMInsights solution.

Before a Log Analytics workspace can be used with VM insights, it must have the VMInsights solution installed.

Step 3: Install agents on the VM and VM scale set to be monitored.

Prior to onboarding agents, you must create and configure a workspace. Install or update the Application Insights Agent as an extension for Azure virtual machines and VM scale sets.

Step 4: Create an Application Insights resource

Sign in to the Azure portal, and create an Application Insights resource.

[Home > New > Application Insights >](#)

## Application Insights

Monitor web app performance and usage

[Basics](#)   [Tags](#)   [Review + create](#)

Create an Application Insights resource to monitor your live web application. With Application Insights, you have full observability into your application across all components and dependencies of your complex distributed architecture. It includes powerful analytics tools to help you diagnose issues and to understand what users actually do with your app. It's designed to help you continuously improve performance and usability. It works for apps on a wide variety of platforms including .NET, Node.js and Java EE, hosted on-premises, hybrid, or any public cloud. [Learn More](#)

**PROJECT DETAILS**

Select a subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \* ⓘ

Visual Studio Enterprise

Resource Group \* ⓘ

My\_Resource\_Group

[Create new](#)**INSTANCE DETAILS**

Name \* ⓘ

My\_AppInsights\_Resource

Region \* ⓘ

(US) West US 2

Resource Mode \* ⓘ

Classic   **Workspace-based****WORKSPACE DETAILS**

Subscription \* ⓘ

Visual Studio Enterprise

Log Analytics Workspace \* ⓘ

my-workspace-name [westus2]

[Review + create](#)[« Previous](#)[Next : Tags >](#)

Once a workspace-based Application Insights resource has been created, configuring monitoring is relatively straightforward.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-monitor/vm/vminsights-configure-workspace> <https://docs.microsoft.com/en-us/azure/azure-monitor/app/create-workspace-resource>

**Topic 5 - Question Set 5**

Question #1

Topic 5

**DRAG DROP -**

You have an application that provides weather forecasting data to external partners. You use Azure API Management to publish APIs.

You must change the behavior of the API to meet the following requirements:

- Support alternative input parameters
- Remove formatting text from responses
- Provide additional context to back-end services

Which types of policies should you implement? To answer, drag the policy types to the correct scenarios. Each policy 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:

**Policy types**

- Inbound
- Outbound
- Backend

**Answer Area****Requirement**

Rewrite the request URL to match to the format expected by the web service.

**Policy type**

policy type

Remove formatting text from responses.

policy type

Forward the user ID that is associated with the subscription key for the original request to the back-end service.

policy type

**Correct Answer:**

**Policy types**

- Inbound
- Outbound
- Backend

**Answer Area****Requirement**

Rewrite the request URL to match to the format expected by the web service.

**Policy type**

Inbound

Remove formatting text from responses.

Outbound

Forward the user ID that is associated with the subscription key for the original request to the back-end service.

Inbound

Reference:

<https://docs.microsoft.com/en-us/azure/api-management/api-management-howto-policies>

Question #2

Topic 5

You are developing an e-commerce solution that uses a microservice architecture.

You need to design a communication backplane for communicating transactional messages between various parts of the solution. Messages must be communicated in first-in-first-out (FIFO) order.

What should you use?

- A. Azure Storage Queue
- B. Azure Event Hub
- C. Azure Service Bus
- D. Azure Event Grid

**Correct Answer:** A 

As a solution architect/developer, you should consider using Service Bus queues when:

Your solution requires the queue to provide a guaranteed first-in-first-out (FIFO) ordered delivery.

Reference:

<https://docs.microsoft.com/en-us/azure/service-bus-messaging/service-bus-azure-and-service-bus-queues-compared-contrasted>

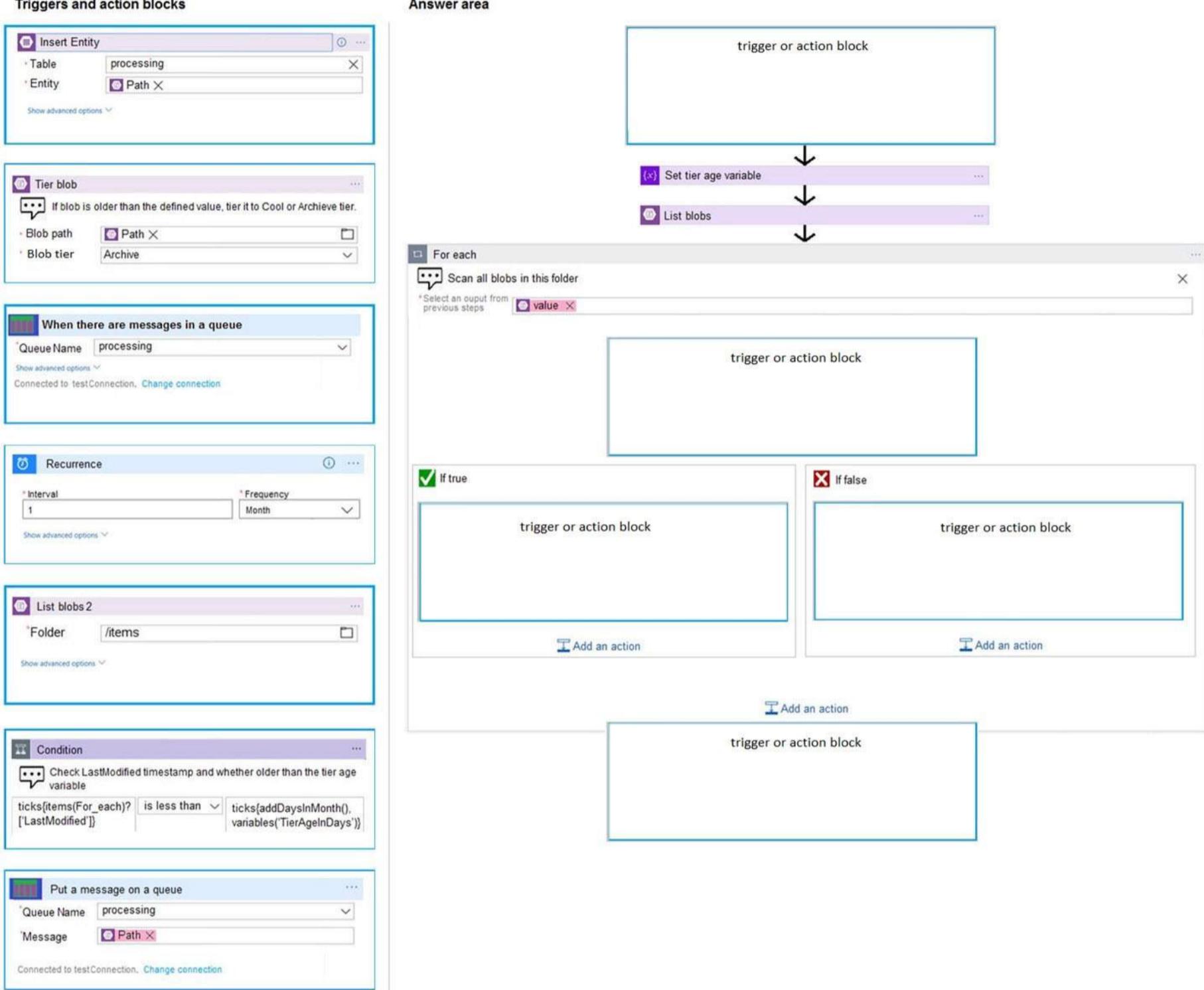
## Question #3

## DRAG DROP -

A company backs up all manufacturing data to Azure Blob Storage. Admins move blobs from hot storage to archive tier storage every month. You must automatically move blobs to Archive tier after they have not been modified within 180 days. The path for any item that is not archived must be placed in an existing queue. This operation must be performed automatically once a month. You set the value of TierAgeInDays to -180. How should you configure the Logic App? To answer, drag the appropriate triggers or action blocks to the correct trigger or action slots. Each trigger or action block 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:



**Correct Answer:**

**Triggers and action blocks**

**Answer area**

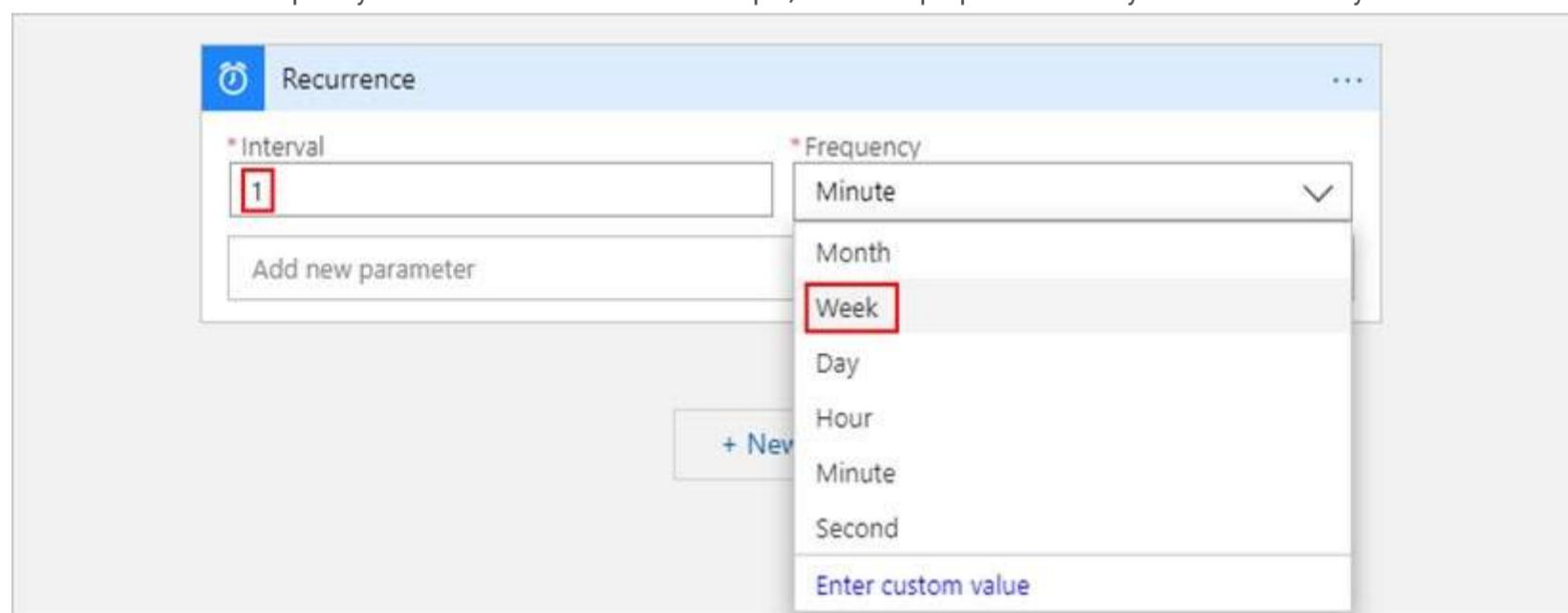
```

graph TD
    Recurrence[Recurrence] --> SetTier[Set tier age variable]
    SetTier --> ForEach[For each]
    ForEach --> Condition1[Condition]
    Condition1 -- If true --> QueueTrigger1[When there are messages in a queue]
    QueueTrigger1 --> PutQueue[Put a message on a queue]
    Condition1 -- If false --> Tierblob[Tier blob]
    Tierblob --> ListBlobs2[List blobs 2]
  
```

**Box 1: Reoccurrence..**

To regularly run tasks, processes, or jobs on specific schedule, you can start your logic app workflow with the built-in Recurrence - Schedule trigger. You can set a date and time as well as a time zone for starting the workflow and a recurrence for repeating that workflow.

Set the interval and frequency for the recurrence. In this example, set these properties to run your workflow every week.

**Box 2: Condition..**

To run specific actions in your logic app only after passing a specified condition, add a conditional statement. This control structure compares the data in your workflow against specific values or fields. You can then specify different actions that run based on whether or not the data meets the condition.

**Box 3: Put a message on a queue -**

The path for any item that is not archived must be placed in an existing queue.

Note: Under If true and If false, add the steps to perform based on whether the condition is met.

**Box 4: ..tier it to Cool or Archive tier.**

Archive item.

**Box 5: List blobs 2 -**

Reference:

<https://docs.microsoft.com/en-us/azure/connectors/connectors-native-recurrence> <https://docs.microsoft.com/en-us/azure/logic-apps/logic-apps-control-flow-loops> <https://docs.microsoft.com/en-us/azure/logic-apps/logic-apps-control-flow-conditional-statement>

## Question #4

Topic 5

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 are developing an Azure Service application that processes queue data when it receives a message from a mobile application. Messages may not be sent to the service consistently.

You have the following requirements:

- Queue size must not grow larger than 80 gigabytes (GB).
- Use first-in-first-out (FIFO) ordering of messages.
- Minimize Azure costs.

You need to implement the messaging solution.

Solution: Use the .Net API to add a message to an Azure Service Bus Queue from the mobile application. Create an Azure Function App that uses an Azure Service Bus Queue trigger.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer: A** 

You can create a function that is triggered when messages are submitted to an Azure Storage queue.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-functions/functions-create-storage-queue-triggered-function>

## Question #5

Topic 5

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 are developing an Azure solution to collect point-of-sale (POS) device data from 2,000 stores located throughout the world. A single device can produce 2 megabytes (MB) of data every 24 hours. Each store location has one to five devices that send data.

You must store the device data in Azure Blob storage. Device data must be correlated based on a device identifier. Additional stores are expected to open in the future.

You need to implement a solution to receive the device data.

Solution: Provision an Azure Notification Hub. Register all devices with the hub.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer: B** 

Instead use an Azure Service Bus, which is used for order processing and financial transactions.

Reference:

<https://docs.microsoft.com/en-us/azure/event-grid/compare-messaging-services>

## Question #6

Topic 5

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 are developing an Azure solution to collect point-of-sale (POS) device data from 2,000 stores located throughout the world. A single device can produce 2 megabytes (MB) of data every 24 hours. Each store location has one to five devices that send data.

You must store the device data in Azure Blob storage. Device data must be correlated based on a device identifier. Additional stores are expected to open in the future.

You need to implement a solution to receive the device data.

Solution: Provision an Azure Service Bus. Configure a topic to receive the device data by using a correlation filter.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer: A**

A message is raw data produced by a service to be consumed or stored elsewhere. The Service Bus is for high-value enterprise messaging, and is used for order processing and financial transactions.

Reference:

<https://docs.microsoft.com/en-us/azure/event-grid/compare-messaging-services>

## Question #7

Topic 5

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 are developing an Azure solution to collect point-of-sale (POS) device data from 2,000 stores located throughout the world. A single device can produce 2 megabytes (MB) of data every 24 hours. Each store location has one to five devices that send data.

You must store the device data in Azure Blob storage. Device data must be correlated based on a device identifier. Additional stores are expected to open in the future.

You need to implement a solution to receive the device data.

Solution: Provision an Azure Event Grid. Configure event filtering to evaluate the device identifier.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer: B**

Instead use an Azure Service Bus, which is used for order processing and financial transactions.

Note: An event is a lightweight notification of a condition or a state change. Event hubs are usually used reacting to status changes.

Reference:

<https://docs.microsoft.com/en-us/azure/event-grid/compare-messaging-services>

Question #8

Topic 5

**DRAG DROP -**

You manage several existing Logic Apps.

You need to change definitions, add new logic, and optimize these apps on a regular basis.

What should you use? To answer, drag the appropriate tools to the correct functionalities. Each tool 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**

Tools	Functionality	Tool
Logic Apps Designer	Edit B2B workflows	
Code View Editor	Edit definitions in JSON	
Enterprise Integration Pack	Visually add functionality	

**Answer Area**

Tools	Functionality	Tool
Correct Answer:	Logic Apps Designer	Edit B2B workflows
	Code View Editor	Edit definitions in JSON
	Enterprise Integration Pack	Visually add functionality

Box 1: Enterprise Integration Pack

For business-to-business (B2B) solutions and seamless communication between organizations, you can build automated scalable enterprise integration workflows by using the Enterprise Integration Pack (EIP) with Azure Logic Apps.

Box 2: Code View Editor -

Edit JSON - Azure portal -

1. Sign in to the Azure portal.
2. From the left menu, choose All services. In the search box, find "logic apps", and then from the results, select your logic app.
3. On your logic app's menu, under Development Tools, select Logic App Code View.
4. The Code View editor opens and shows your logic app definition in JSON format.

Box 3: Logic Apps Designer -

Reference:

<https://docs.microsoft.com/en-us/azure/logic-apps/logic-apps-enterprise-integration-overview> <https://docs.microsoft.com/en-us/azure/logic-apps/logic-apps-author-definitions>

## Question #9

## Topic 5

A company is developing a solution that allows smart refrigerators to send temperature information to a central location.

The solution must receive and store messages until they can be processed. You create an Azure Service Bus instance by providing a name, pricing tier, subscription, resource group, and location.

You need to complete the configuration.

Which Azure CLI or PowerShell command should you run?

A.

```
az servicebus queue create  
--resource-group fridge-rg  
--namespace-name fridge-ns  
--name fridge-q
```

B.

```
New-AzureRmResourceGroup  
-Name fridge-rg  
-Location fridge-loc
```

C.

```
az servicebus namespace create  
--resource-group fridge-rg  
--name fridge-ns  
--location fridge-loc
```

D.

```
connectionString-$) az serviceBus namespace authorization-rule keys list  
--resource-group fridge-rg  
--fridge-ns fridge-ns  
--query primaryConnectionString --output tsv)
```

**Correct Answer: A**

A service bus instance has already been created (Step 2 below). Next is step 3, Create a Service Bus queue.

Note:

Steps:

Step 1: # Create a resource group

```
resourceGroupName="myResourceGroup"
```

```
az group create --name $resourceGroupName --location eastus
```

Step 2: # Create a Service Bus messaging namespace with a unique name namespaceName=myNameSpace\$RANDOM az servicebus namespace create --resource-group \$resourceGroupName --name \$namespaceName --location eastus

Step 3: # Create a Service Bus queue

```
az servicebus queue create --resource-group $resourceGroupName --namespace-name $namespaceName --name BasicQueue
```

Step 4: # Get the connection string for the namespace

```
connectionString=$(az servicebus namespace authorization-rule keys list --resource-group $resourceGroupName --namespace-name $namespaceName --name RootManageSharedAccessKey --query primaryConnectionString --output tsv)
```

Reference:

<https://docs.microsoft.com/en-us/azure/service-bus-messaging/service-bus-quickstart-cli>

Question #10

Topic 5

**HOTSPOT -**

You are developing an application that uses Azure Storage Queues.

You have the following code:

```
CloudStorageAccount storageAccount = CloudStorageAccount.Parse
(CloudConfigurationManager.GetSetting("StorageConnectionString"));
CloudQueueClient queueClient = storageAccount.CreateCloudQueueClient();

CloudQueue queue = queueClient.GetQueueReference("appqueue");
await queue.CreateIfNotExistsAsync();

CloudQueueMessage peekedMessage = await queue.PeekMessageAsync();
if (peekedMessage != null)
{
    Console.WriteLine("The peeked message is: {0}", peekedMessageAsString);
}
CloudQueueMessage message = await queue.GetMessageAsync();
```

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

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

Statement	Yes	No
The code configures the lock duration for the queue.	<input type="radio"/>	<input type="radio"/>
The last message read remains in the queue after the code runs.	<input type="radio"/>	<input type="radio"/>
The storage queue remains in the storage account after the code runs.	<input type="radio"/>	<input type="radio"/>

Correct Answer:

**Answer Area**

Statement	Yes	No
The code configures the lock duration for the queue.	<input type="radio"/>	<input checked="" type="radio"/>
The last message read remains in the queue after the code runs.	<input checked="" type="radio"/>	<input type="radio"/>
The storage queue remains in the storage account after the code runs.	<input checked="" type="radio"/>	<input type="radio"/>

Box 1: No -

The QueueDescription.LockDuration property gets or sets the duration of a peek lock; that is, the amount of time that the message is locked for other receivers.

The maximum value for LockDuration is 5 minutes; the default value is 1 minute.

Box 2: Yes -

You can peek at the message in the front of a queue without removing it from the queue by calling the PeekMessage method.

Box 3: Yes -

Reference:

<https://docs.microsoft.com/en-us/azure/storage/queues/storage-dotnet-how-to-use-queues> <https://docs.microsoft.com/en-us/dotnet/api/microsoft.servicebus.messaging.queuedescription.lockduration>

## Question #11

## Topic 5

A company is developing a solution that allows smart refrigerators to send temperature information to a central location. The solution must receive and store messages until they can be processed. You create an Azure Service Bus instance by providing a name, pricing tier, subscription, resource group, and location.

You need to complete the configuration.

Which Azure CLI or PowerShell command should you run?

A.

```
az group create  
  --name fridge-rg  
  --location fridge-loc
```

B.

```
New-AzureRmServiceBusNamespace  
  -ResourceGroupName fridge-rg  
  -NamespaceName fridge-ns  
  -Location fridge-loc
```

C.

```
New-AzureRmServiceBusQueue  
  -ResourceGroupName fridge-rg  
  -NamespaceName fridge-ns  
  -Name fridge-q  
  -EnablePartitioning $False
```

D.

```
az servicebus namespace create  
  --resource-group fridge-rg  
  --name fridge-rg  
  --location fridge-loc
```

**Correct Answer: C**

A service bus instance has already been created (Step 2 below). Next is step 3, Create a Service Bus queue.

Note:

Steps:

Step 1: # Create a resource group

```
resourceGroupName="myResourceGroup"
```

```
az group create --name $resourceGroupName --location eastus
```

Step 2: # Create a Service Bus messaging namespace with a unique name namespaceName=myNameSpace\$RANDOM az servicebus namespace create --resource-group \$resourceGroupName --name \$namespaceName --location eastus

Step 3: # Create a Service Bus queue

```
az servicebus queue create --resource-group $resourceGroupName --namespace-name $namespaceName --name BasicQueue
```

Step 4: # Get the connection string for the namespace

```
connectionString=$(az servicebus namespace authorization-rule keys list --resource-group $resourceGroupName --namespace-name $namespaceName --name
```

```
RootManageSharedAccessKey --query primaryConnectionString --output tsv)
```

Reference:

<https://docs.microsoft.com/en-us/azure/service-bus-messaging/service-bus-quickstart-cli>

## Question #12

## Topic 5

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 are developing an Azure Service application that processes queue data when it receives a message from a mobile application. Messages may not be sent to the service consistently.

You have the following requirements:

- Queue size must not grow larger than 80 gigabytes (GB).
- Use first-in-first-out (FIFO) ordering of messages.
- Minimize Azure costs.

You need to implement the messaging solution.

Solution: Use the .Net API to add a message to an Azure Storage Queue from the mobile application. Create an Azure Function App that uses an Azure Storage Queue trigger.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer: B** 

Create an Azure Function App that uses an Azure Service Bus Queue trigger.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-functions/functions-create-storage-queue-triggered-function>

Question #13

Topic 5

**DRAG DROP -**

You develop software solutions for a mobile delivery service. You are developing a mobile app that users can use to order from a restaurant in their area. The app uses the following workflow:

1. A driver selects the restaurants for which they will deliver orders.
2. Orders are sent to all available drivers in an area.
3. Only orders for the selected restaurants will appear for the driver.
4. The first driver to accept an order removes it from the list of available orders.

You need to implement an Azure Service Bus solution.

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 single Service Bus topic.

Create a Service Bus Namespace for each restaurat for which a driver can receive messages.

Create a single Service Bus subscription.

Create a Service Bus subscription for each restaurant for which a driver can receive orders.

Create s single Service Bus Namespace.

Create a Service Bus topic for each restaurant for which a driver can receive messages.

**Answer Area****Correct Answer:****Actions**

Create a single Service Bus topic.

Create a Service Bus Namespace for each restaurat for which a driver can receive messages.

Create a single Service Bus subscription.

Create a Service Bus subscription for each restaurant for which a driver can receive orders.

Create s single Service Bus Namespace.

Create a Service Bus topic for each restaurant for which a driver can receive messages.

**Answer Area**

Create s single Service Bus Namespace.

Create a Service Bus topic for each restaurant for which a driver can receive messages.

Create a Service Bus subscription for each restaurant for which a driver can receive orders.

Box 1: Create a single Service Bus Namespace

To begin using Service Bus messaging entities in Azure, you must first create a namespace with a name that is unique across Azure. A namespace provides a scoping container for addressing Service Bus resources within your application.

Box 2: Create a Service Bus Topic for each restaurant for which a driver can receive messages.

Create topics.

Box 3: Create a Service Bus subscription for each restaurant for which a driver can receive orders.

Topics can have multiple, independent subscriptions.

Reference:

<https://docs.microsoft.com/en-us/azure/service-bus-messaging/service-bus-messaging-overview>

Question #14

Topic 5

**HOTSPOT -**

You develop a news and blog content app for Windows devices.

A notification must arrive on a user's device when there is a new article available for them to view.

You need to implement push notifications.

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

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

```
string notificationHubName = "contoso_hub";
string notificationHubConnection = "connection_string";
hub =
NotificationHubClient
NotificationHubClientSettings
NotificationHubJob
NotificationDetails

NotificationHubClient
NotificationHubClientSettings
NotificationHubJob
NotificationDetails

GetInstallation
CreateClientFromConnectionString
CreateOrUpdateInstallation
PatchInstallation

(notificationHubConnection, notificationHubName);
string windowsToastPayload =
@"<toast><visual><binding template=""ToastText01""><text id=""1"">" +
"@New item to view" + @"</text></binding></visual></toast>";
try
{
var result =
await hub.
(windowsToastPayload);

SendWindowsNativeNotificationAsync
SubmitNotificationHubJobAsync
ScheduleNotificationAsync
SendAppleNativeNotificationAsync

...
}
catch (System.Exception ex)
{
...
}
...
```

**Correct Answer:****Answer Area**

```
string notificationHubName = "contoso_hub";
string notificationHubConnection = "connection_string";
hub =
NotificationHubClient
NotificationHubClientSettings
NotificationHubJob
NotificationDetails

NotificationHubClient
NotificationHubClientSettings
NotificationHubJob
NotificationDetails

(notificationHubConnection, notificationHubName);
string windowsToastPayload =
@"<toast><visual><binding template=""ToastText01""><text id=""1"">" +
@"New item to view" + @"</text></binding></visual></toast>";
try
{
var result =
await hub.
SendWindowsNativeNotificationAsync(windowsToastPayload);
}
catch (System.Exception ex)
{
...
}
...
Box 1: NotificationHubClient -
```

Box 2: NotificationHubClient -

Box 3: CreateClientFromConnectionString

// Initialize the Notification Hub

NotificationHubClient hub = NotificationHubClient.CreateClientFromConnectionString(listenConnString, hubName);

Box 4: SendWindowsNativeNotificationAsync

Send the push notification.

var result = await hub.SendWindowsNativeNotificationAsync(windowsToastPayload);

Reference:

<https://docs.microsoft.com/en-us/azure/notification-hubs/notification-hubs-push-notification-registration-management><https://github.com/MicrosoftDocs/azure-docs/blob/master/articles/app-service-mobile/app-service-mobile-windows-store-dotnet-get-started-push.md>

Question #15

Topic 5

You are developing an Azure messaging solution.

You need to ensure that the solution meets the following requirements:

- Provide transactional support.
- Provide duplicate detection.
- Store the messages for an unlimited period of time.

Which two technologies will meet the requirements? Each correct answer presents a complete solution.

NOTE: Each correct selection is worth one point.

- A. Azure Service Bus Topic
- B. Azure Service Bus Queue
- C. Azure Storage Queue
- D. Azure Event Hub

**Correct Answer:** AB 

The Azure Service Bus Queue and Topic has duplicate detection.

Enabling duplicate detection helps keep track of the application-controlled MessageId of all messages sent into a queue or topic during a specified time window.

Incorrect Answers:

C: There is just no mechanism that can query a Storage queue and find out if a message with the same contents is already there or was there before.

D: Azure Event Hub does not have duplicate detection

Reference:

<https://docs.microsoft.com/en-us/azure/service-bus-messaging/duplicate-detection>

Question #16

Topic 5

**DRAG DROP -**

You develop a gateway solution for a public facing news API.

The news API back end is implemented as a RESTful service and hosted in an Azure App Service instance.

You need to configure back-end authentication for the API Management service instance.

Which target and gateway credential type should you use? To answer, drag the appropriate values to the correct parameters. Each value 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:

Values	Answer Area	
Azure Resource	Configuration parameter	Value
HTTP(s) endpoint	Target	
Basic	Gateway credentials	
Client cert		

Correct Answer:

Values	Answer Area	
	Configuration parameter	Value
HTTP(s) endpoint	Target	Azure Resource
Basic	Gateway credentials	Client cert

Box 1: Azure Resource -

Box 2: Client cert -

API Management allows to secure access to the back-end service of an API using client certificates.

Reference:

<https://docs.microsoft.com/en-us/rest/api/apimanagement/apimanagementrest/azure-api-management-rest-api-backend-entity>

Question #17

Topic 5

**HOTSPOT -**

You are creating an app that uses Event Grid to connect with other services. Your app's event data will be sent to a serverless function that checks compliance.

This function is maintained by your company.

You write a new event subscription at the scope of your resource. The event must be invalidated after a specific period of time.

You need to configure Event Grid.

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

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

Option	Value
WebHook event delivery	<div style="border: 1px solid black; padding: 5px; width: fit-content;"> <input type="checkbox"/> SAS tokens  <input type="checkbox"/> Key authentication  <input type="checkbox"/> Management Access Control         </div>
Topic publishing	<div style="border: 1px solid black; padding: 5px; width: fit-content;"> <input type="checkbox"/> ValidationCode handshake  <input type="checkbox"/> ValidationURL handshake  <input type="checkbox"/> JWT token         </div>

**Answer Area**

Option	Value
WebHook event delivery	<div style="border: 1px solid black; padding: 5px; width: fit-content;"> <input checked="" type="checkbox"/> SAS tokens  <input type="checkbox"/> Key authentication  <input type="checkbox"/> Management Access Control         </div>
Topic publishing	<div style="border: 1px solid black; padding: 5px; width: fit-content;"> <input checked="" type="checkbox"/> ValidationCode handshake  <input type="checkbox"/> ValidationURL handshake  <input type="checkbox"/> JWT token         </div>

Box 1: SAS tokens -

Custom topics use either Shared Access Signature (SAS) or key authentication. Microsoft recommends SAS, but key authentication provides simple programming, and is compatible with many existing webhook publishers.

In this case we need the expiration time provided by SAS tokens.

Box 2: ValidationCode handshake -

Event Grid supports two ways of validating the subscription: ValidationCode handshake (programmatic) and ValidationURL handshake (manual).

If you control the source code for your endpoint, this method is recommended.

Incorrect Answers:

ValidationURL handshake (manual): In certain cases, you can't access the source code of the endpoint to implement the ValidationCode handshake. For example, if you use a third-party service (like Zapier or IFTTT), you can't programmatically respond with the validation code.

Reference:

<https://docs.microsoft.com/en-us/azure/event-grid/security-authentication>

Question #18

Topic 5

**HOTSPOT -**

You are working for Contoso, Ltd.

You define an API Policy object by using the following XML markup:

```
<set-variable name="bodySize" value="@{context.Request.Headers["Content-Length"] [0]}"/>
<choose>
  <when condition="@(int.Parse(context.Variables.GetValueOrDefault<string> ("bodySize"))<512000)">
  </when>
<otherwise>
  <rewrite-uri template="/put"/>
  <set-backend-service base-url="http://contoso.com/api/9.1"/>
</otherwise>
</choose>
```

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

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

Statement	Yes	No
The XML segment belongs in the <inbound> section of the policy.	<input type="radio"/>	<input type="radio"/>
If the body size is >256k, an error will occur.	<input type="radio"/>	<input type="radio"/>
If the request is http://contoso.com/api/9.2/, the policy will retain the higher version.	<input type="radio"/>	<input type="radio"/>

Correct Answer:

**Answer Area**

Statement	Yes	No
The XML segment belongs in the <inbound> section of the policy.	<input checked="" type="radio"/>	<input type="radio"/>
If the body size is >256k, an error will occur.	<input type="radio"/>	<input checked="" type="radio"/>
If the request is http://contoso.com/api/9.2/, the policy will retain the higher version.	<input type="radio"/>	<input checked="" type="radio"/>

Box 1: Yes -

Use the set-backend-service policy to redirect an incoming request to a different backend than the one specified in the API settings for that operation. Syntax:

```
<set-backend-service base-url="base URL of the backend service" />
```

Box 2: No -

The condition is on 512k, not on 256k.

Box 3: No -

The set-backend-service policy changes the backend service base URL of the incoming request to the one specified in the policy.

Reference:

<https://docs.microsoft.com/en-us/azure/api-management/api-management-transformation-policies>

## Question #19

Topic 5

You are developing a solution that will use Azure messaging services.

You need to ensure that the solution uses a publish-subscribe model and eliminates the need for constant polling.

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

NOTE: Each correct selection is worth one point.

- A. Service Bus
- B. Event Hub
- C. Event Grid
- D. Queue

**Correct Answer:** AC 

It is strongly recommended to use available messaging products and services that support a publish-subscribe model, rather than building your own. In Azure, consider using Service Bus or Event Grid. Other technologies that can be used for pub/sub messaging include Redis, RabbitMQ, and Apache Kafka.

Reference:

<https://docs.microsoft.com/en-us/azure/architecture/patterns/publisher-subscriber>

## Question #20

Topic 5

A company is implementing a publish-subscribe (Pub/Sub) messaging component by using Azure Service Bus. You are developing the first subscription application.

In the Azure portal you see that messages are being sent to the subscription for each topic. You create and initialize a subscription client object by supplying the correct details, but the subscription application is still not consuming the messages.

You need to ensure that the subscription client processes all messages.

Which code segment should you use?

- A. await subscriptionClient.AddRuleAsync(new RuleDescription(RuleDescription.DefaultRuleName, new TrueFilter()));
- B. subscriptionClient = new SubscriptionClient(ServiceBusConnectionString, TopicName, SubscriptionName);
- C. await subscriptionClient.CloseAsync();
- D. subscriptionClient.RegisterMessageHandler(ProcessMessagesAsync, messageHandlerOptions);

**Correct Answer:** D 

Using topic client, call RegisterMessageHandler which is used to receive messages continuously from the entity. It registers a message handler and begins a new thread to receive messages. This handler is waited on every time a new message is received by the receiver.

`subscriptionClient.RegisterMessageHandler(ReceiveMessagesAsync, messageHandlerOptions);`

Reference:

<https://www.c-sharpcorner.com/article/azure-service-bus-topic-and-subscription-pub-sub/>

## Question #21

## Topic 5

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 are developing an Azure Service application that processes queue data when it receives a message from a mobile application. Messages may not be sent to the service consistently.

You have the following requirements:

- Queue size must not grow larger than 80 gigabytes (GB).
- Use first-in-first-out (FIFO) ordering of messages.
- Minimize Azure costs.

You need to implement the messaging solution.

Solution: Use the .Net API to add a message to an Azure Storage Queue from the mobile application. Create an Azure VM that is triggered from Azure Storage

Queue events.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer: B**

Don't use a VM, instead create an Azure Function App that uses an Azure Service Bus Queue trigger.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-functions/functions-create-storage-queue-triggered-function>

## Question #22

## Topic 5

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 are developing an Azure Service application that processes queue data when it receives a message from a mobile application. Messages may not be sent to the service consistently.

You have the following requirements:

- Queue size must not grow larger than 80 gigabytes (GB).
- Use first-in-first-out (FIFO) ordering of messages.
- Minimize Azure costs.

You need to implement the messaging solution.

Solution: Use the .Net API to add a message to an Azure Service Bus Queue from the mobile application. Create an Azure Windows VM that is triggered from

Azure Service Bus Queue.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer: B**

Don't use a VM, instead create an Azure Function App that uses an Azure Service Bus Queue trigger.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-functions/functions-create-storage-queue-triggered-function>

Question #23

**DRAG DROP -**

You are developing a REST web service. Customers will access the service by using an Azure API Management instance.

The web service does not correctly handle conflicts. Instead of returning an HTTP status code of 409, the service returns a status code of 500.

The body of the status message contains only the word conflict.

You need to ensure that conflicts produce the correct response.

How should you complete the policy? 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 one point.

Select and Place:

**Policy segments**

server  
context  
on-error  
set-status  
when-error  
override-status

**Answer Area**

```
< Policy segment >
<base />
<choose>
  <when condition = " @ [Policy segment].Response.StatusCode == 500
    && [Policy segment].LastError.Message.Contains
      <return-response>
        < Policy segment >
        </return-response>
      </when>
      <otherwise />
    </choose>
< Policy segment >
```

**Correct Answer:****Policy segments**

server  
context  
on-error  
set-status  
when-error  
override-status

**Answer Area**

```
< on-error >
<base />
<choose>
  <when condition = " @ [context].Response.StatusCode == 500
    && [context].LastError.Message.Contains
      <return-response>
        < set-status >
        </return-response>
      </when>
      <otherwise />
    </choose>
< on-error >
```

Box 1: on-error -

Policies in Azure API Management are divided into inbound, backend, outbound, and on-error.

If there is no on-error section, callers will receive 400 or 500 HTTP response messages if an error condition occurs.

Box 2: context -

Box 3: context -

Box 4: set-status -

The return-response policy aborts pipeline execution and returns either a default or custom response to the caller. Default response is 200 OK with no body.

Custom response can be specified via a context variable or policy statements.

Syntax:

```
<return-response response-variable-name="existing context variable">
<set-header/>
```

```
<set-body/>
<set-status/>
</return-response>
```

Box 5: on-error -

Reference:

<https://docs.microsoft.com/en-us/azure/api-management/api-management-error-handling-policies> <https://docs.microsoft.com/en-us/azure/api-management/api-management-transformation-policies>

## Question #24

## DRAG DROP -

You are a developer for a Software as a Service (SaaS) company. You develop solutions that provide the ability to send notifications by using Azure Notification Hubs.

You need to create sample code that customers can use as a reference for how to send raw notifications to Windows Push Notification Services (WNS) devices.

The sample code must not use external packages.

How should you complete the code segment? 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 one point.

Select and Place:

## Code segments

- raw
- windows
- windowsphone
- application/xml
- application/json
- application/octet-stream

## Answer Area

```
var endpoint = "...";
var payload = "...";
var request = new HttpRequestMessage(HttpMethod.Post, endpoint);
request.Headers.Add("X-WNS-Type", "wns/raw");
request.Headers.Add("ServiceBusNotification-Format", "Code segment");
request.Content = new StringContent(payload, Encoding.UTF8, "Code segment");
var client = new HttpClient();
await client.SendAsync(request);
```

## Correct Answer:

## Code segments

- raw
- windows
- windowsphone
- application/xml
- application/json

## Answer Area

```
var endpoint = "...";
var payload = "...";
var request = new HttpRequestMessage(HttpMethod.Post, endpoint);
request.Headers.Add("X-WNS-Type", "wns/raw");
request.Headers.Add("ServiceBusNotification-Format", "windows");
request.Content = new StringContent(payload, Encoding.UTF8, "application/octet-stream");
var client = new HttpClient();
await client.SendAsync(request);
```

Box 1: windows -

Example code:

```
var request = new HttpRequestMessage(method, $"{resourceUri}?api-version=2017-04"); request.Headers.Add("Authorization", createToken(resourceUri, KEY_NAME, KEY_VALUE)); request.Headers.Add("X-WNS-Type", "wns/raw"); request.Headers.Add("ServiceBusNotification-Format", "windows"); return request;
```

Box 2: application/octet-stream -

Example code capable of sending a raw notification:

```
string resourceUri = $"https://{{NH_NAMESPACE}}.servicebus.windows.net/{{HUB_NAME}}/messages/"; using (var request = CreateHttpRequest(HttpStatusCode.Post, resourceUri))
{
    request.Content = new StringContent(content, Encoding.UTF8, "application/octet-stream");
    request.Content.Headers.ContentType.CharSet = string.Empty;
    var httpClient = new HttpClient();
    var response = await httpClient.SendAsync(request);
```

```
Console.WriteLine(response.StatusCode);  
}  
  
Reference:  
https://stackoverflow.com/questions/31346714/how-to-send-raw-notification-to-azure-notification-hub/31347901
```

## Question #25

## Topic 5

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 are developing an Azure solution to collect point-of-sale (POS) device data from 2,000 stores located throughout the world. A single device can produce

2 megabytes (MB) of data every 24 hours. Each store location has one to five devices that send data.

You must store the device data in Azure Blob storage. Device data must be correlated based on a device identifier. Additional stores are expected to open in the future.

You need to implement a solution to receive the device data.

Solution: Provision an Azure Event Hub. Configure the machine identifier as the partition key and enable capture.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer:** A 

Reference:

<https://docs.microsoft.com/en-us/azure/event-hubs/event-hubs-programming-guide>

Question #26

Topic 5

**DRAG DROP -**

You are developing an Azure solution to collect inventory data from thousands of stores located around the world. Each store location will send the inventory data hourly to an Azure Blob storage account for processing.

The solution must meet the following requirements:

- Begin processing when data is saved to Azure Blob storage.
- Filter data based on store location information.
- Trigger an Azure Logic App to process the data for output to Azure Cosmos DB.
- Enable high availability and geographic distribution.
- Allow 24-hours for retries.
- Implement an exponential back off data processing.

You need to configure the solution.

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

NOTE: Each correct selection is worth one point.

Select and Place:

**Technologies**

Azure Event Hub

Azure Event Grid

Azure Service Bus

Azure Blob Storage

Azure App Service

Azure Logic App

**Answer Area****Object****Technology**

Event Source

Technology

Event Receiver

Technology

Event Handler

Technology

**Correct Answer:****Technologies**

Azure Event Hub

**Answer Area****Object****Technology**

Event Source

Azure Event Grid

Event Receiver

Azure Logic App

Event Handler

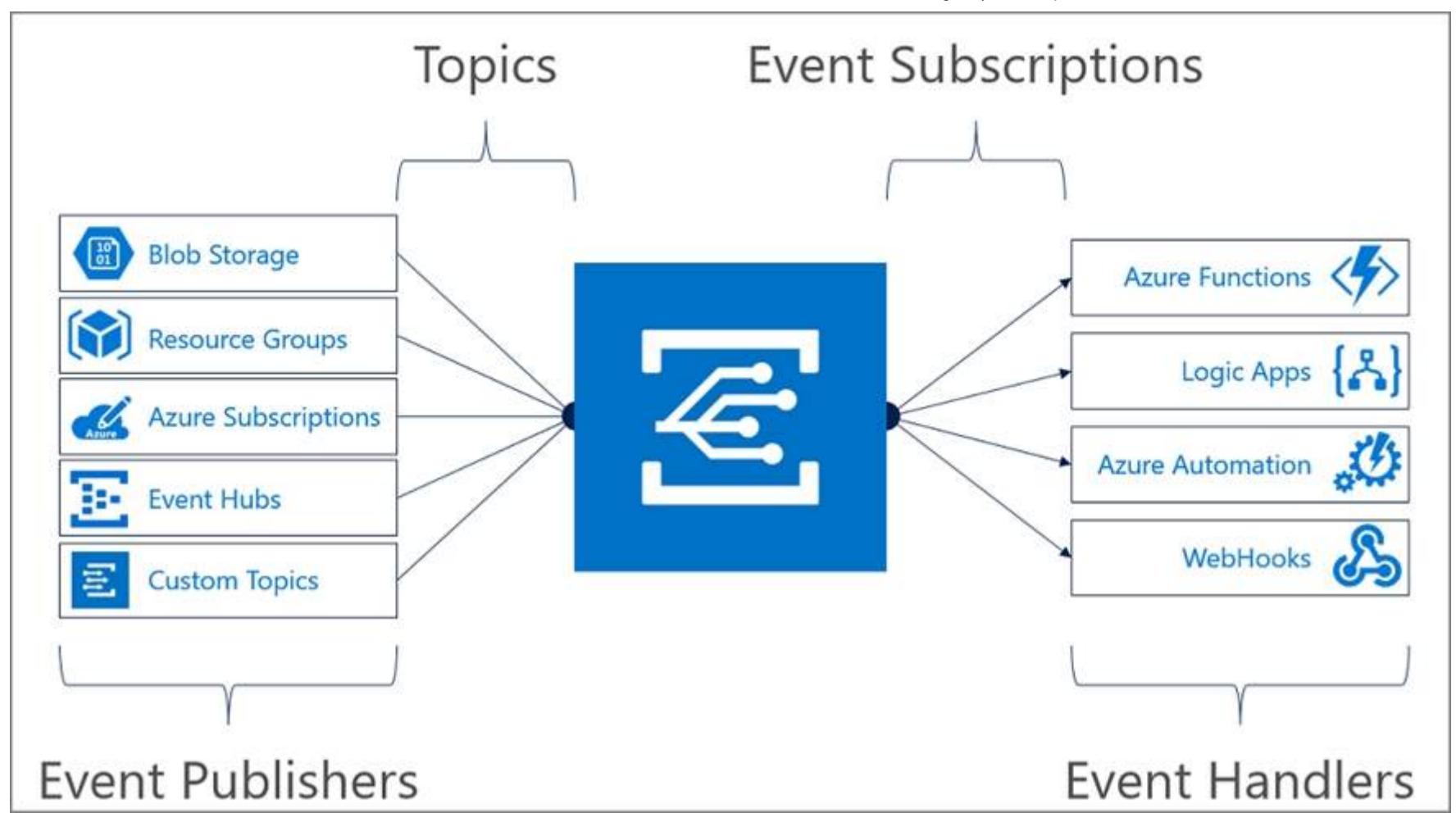
Azure Service Bus

**Box 1: Azure Event Grid -**

Blob storage events are pushed using Azure Event Grid to subscribers such as Azure Functions, Azure Logic Apps, or even to your own http listener. Event Grid provides reliable event delivery to your applications through rich retry policies and dead-lettering.

**Box 2: Azure Logic App -**

Event Grid uses event subscriptions to route event messages to subscribers. This image illustrates the relationship between event publishers, event subscriptions, and event handlers.



Box 3: Azure Service Bus -

The Event Grid service doesn't store events. Instead, events are stored in the Event Handlers, including ServiceBus, EventHub, Storage, Queue,

#### Topic 6 - Question Set 6

Reference:

<https://docs.microsoft.com/en-us/azure/storage/blobs/storage-blob-event-overview> <https://docs.microsoft.com/en-us/java/api/overview/azure/messaging-eventgrid-readme>

## Question #1

## Topic 6

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. Your company has an Azure subscription that includes a storage account, a resource group, a blob container and a file share. A fellow administrator named Jon Ross used an Azure Resource Manager template to deploy a virtual machine and an Azure Storage account. You need to identify the Azure Resource Manager template the Jon Ross used.

Solution: You access the Container blade.

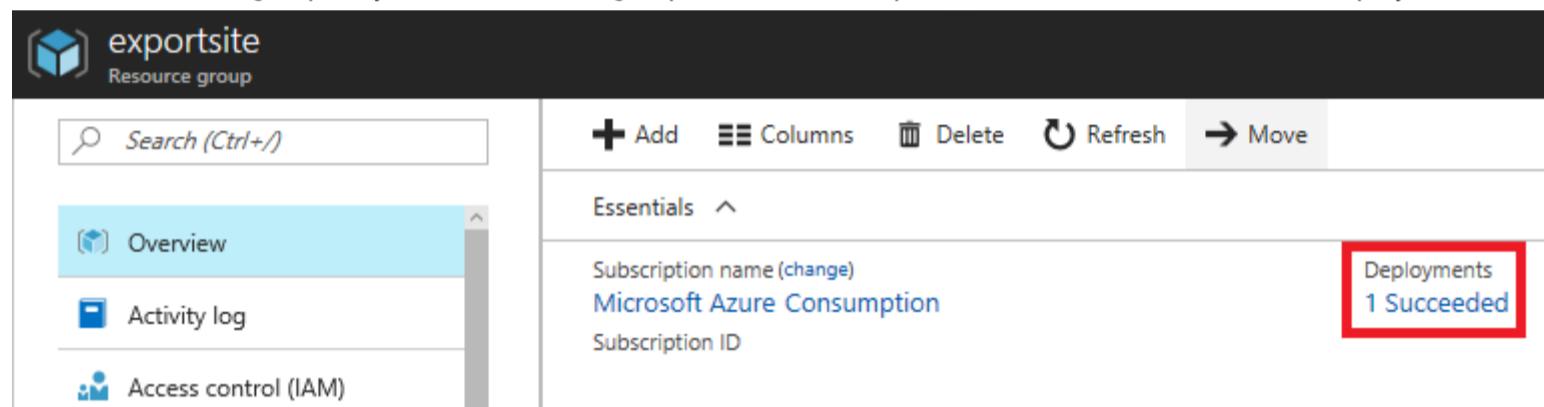
Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer:** B 

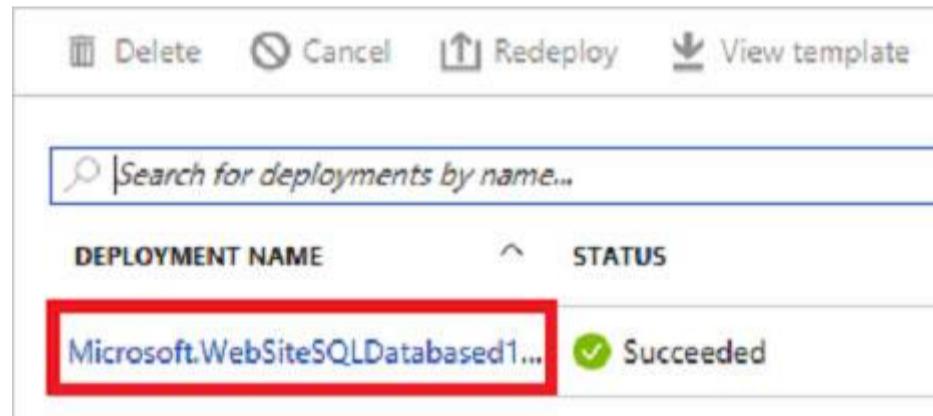
View template from deployment history

Go to the resource group for your new resource group. Notice that the portal shows the result of the last deployment. Select this link.



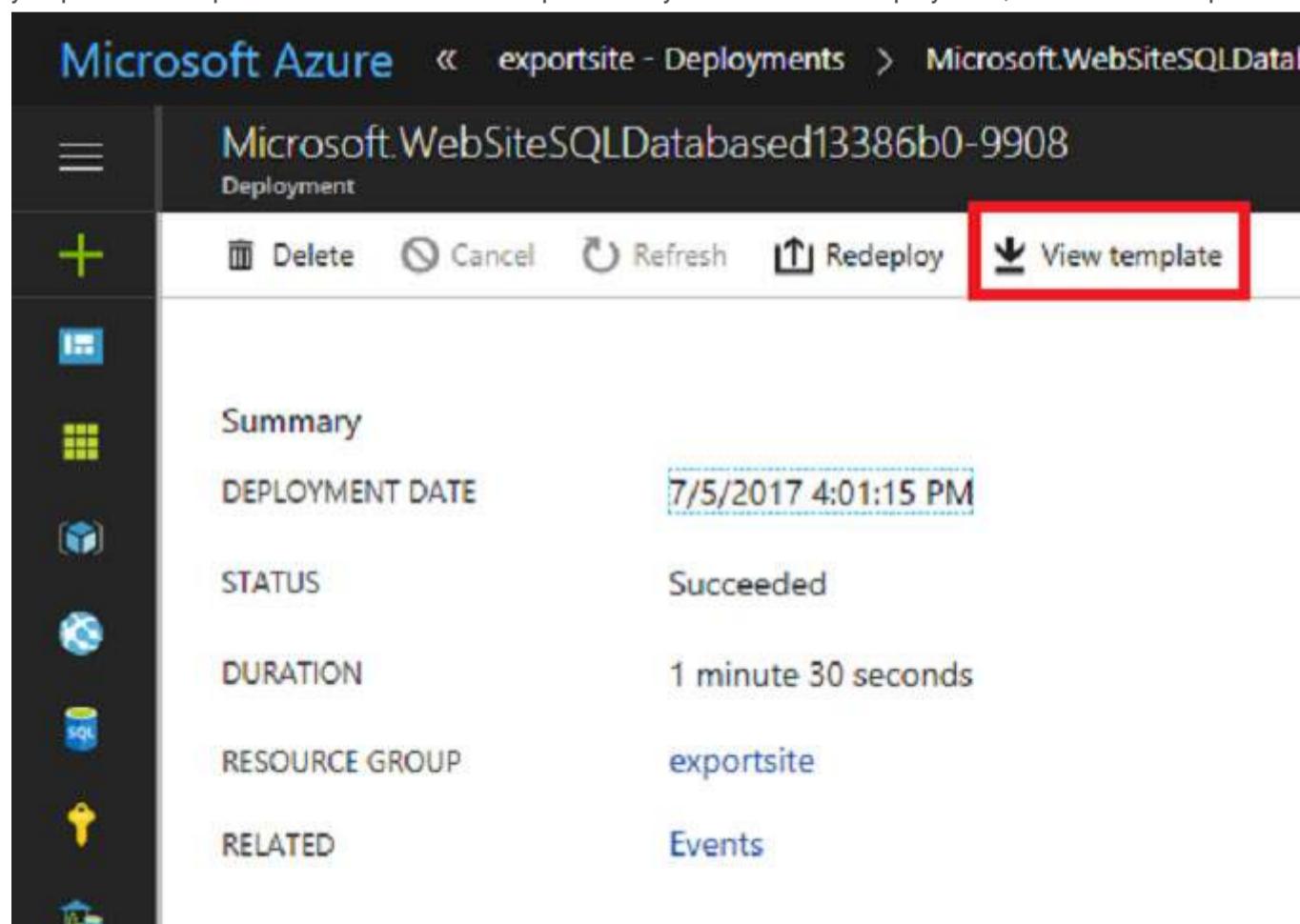
The screenshot shows the Azure Resource Group Overview page for the 'exportsite' resource group. In the top right corner, there is a 'Deployments' section with the text '1 Succeeded'. This section is highlighted with a red box.

You see a history of deployments for the group. In your case, the portal probably lists only one deployment. Select this deployment.



The screenshot shows the Azure Deployment History page. It displays a single deployment entry: 'Microsoft.WebSiteSQLDatabased1...' with a status of 'Succeeded'. This entry is highlighted with a red box.

The portal displays a summary of the deployment. The summary includes the status of the deployment and its operations and the values that you provided for parameters. To see the template that you used for the deployment, select View template.



The screenshot shows the Azure Deployment Summary page for the deployment 'Microsoft.WebSiteSQLDatabased13386b0-9908'. The 'View template' button in the top right corner is highlighted with a red box.

Parameter	Value
DEPLOYMENT DATE	7/5/2017 4:01:15 PM
STATUS	Succeeded
DURATION	1 minute 30 seconds
RESOURCE GROUP	exportsite
RELATED	Events

Reference:

<https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-manager-export-template>



## Question #2

## Topic 6

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. Your company has an Azure subscription that includes a storage account, a resource group, a blob container and a file share. A fellow administrator named Jon Ross used an Azure Resource Manager template to deploy a virtual machine and an Azure Storage account. You need to identify the Azure Resource Manager template the Jon Ross used.

Solution: You access the Virtual Machine blade.

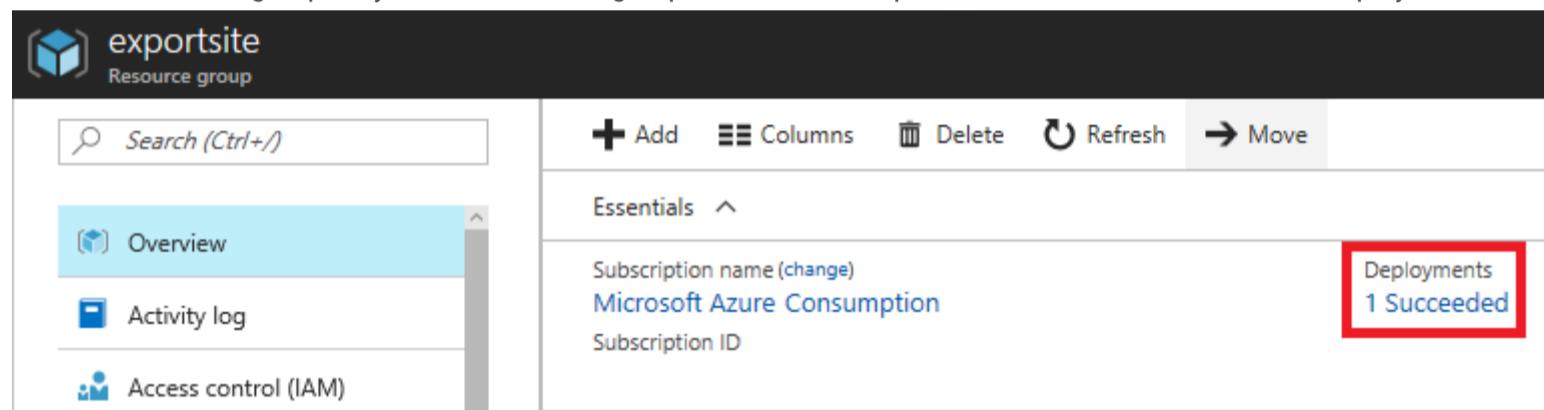
Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer: B** 

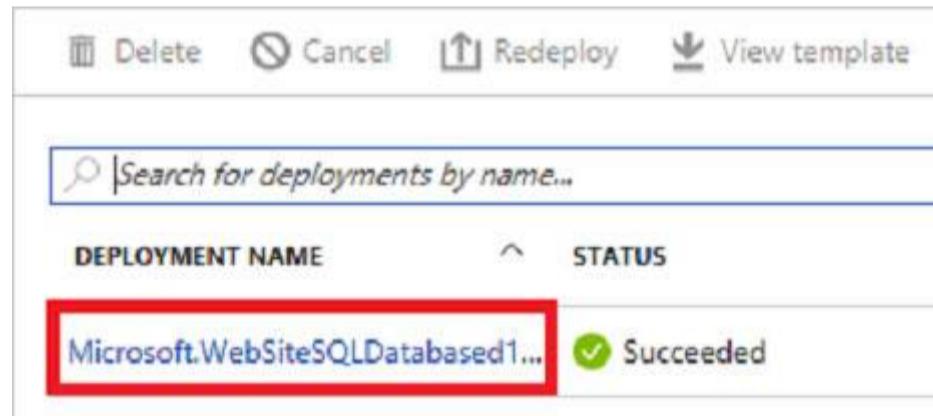
View template from deployment history

Go to the resource group for your new resource group. Notice that the portal shows the result of the last deployment. Select this link.



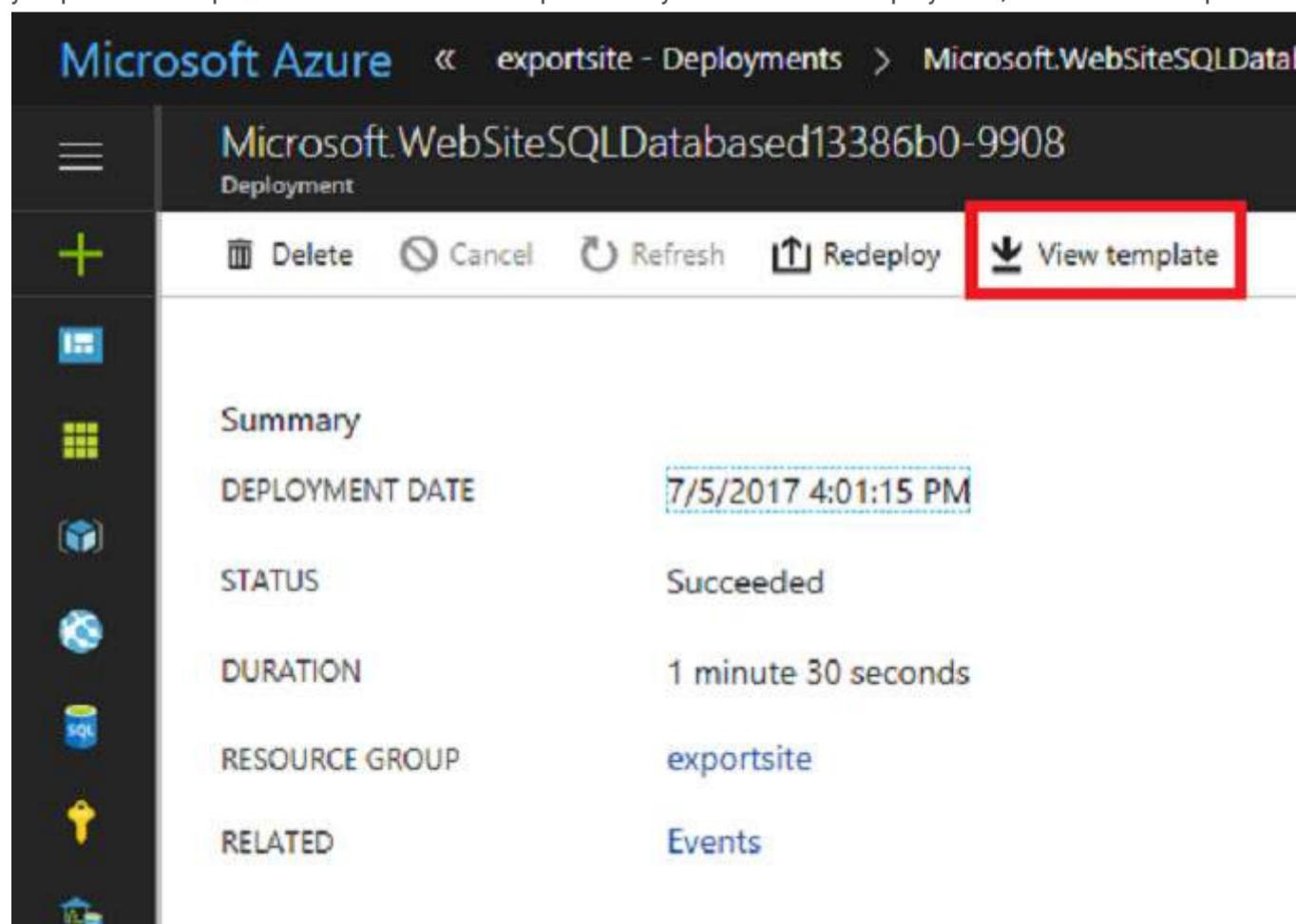
The screenshot shows the Azure Resource Group Overview page for a group named "exportsite". The "Essentials" section displays the "Subscription name (change)" as "Microsoft Azure Consumption" and the "Subscription ID". To the right, a "Deployments" box shows "1 Succeeded". A red box highlights this "1 Succeeded" text.

You see a history of deployments for the group. In your case, the portal probably lists only one deployment. Select this deployment.



The screenshot shows the Azure Deployment History page. It has buttons for "Delete", "Cancel", "Redeploy", and "View template". Below is a search bar and a table with columns "DEPLOYMENT NAME" and "STATUS". A single row is shown: "Microsoft.WebSiteSQLDatabased1..." with a status of "Succeeded". A red box highlights the "Microsoft.WebSiteSQLDatabased1..." link.

The portal displays a summary of the deployment. The summary includes the status of the deployment and its operations and the values that you provided for parameters. To see the template that you used for the deployment, select View template.



The screenshot shows the Azure Deployment Summary page for a deployment named "Microsoft.WebSiteSQLDatabased13386b0-9908". The top navigation bar includes "Microsoft Azure", a back arrow, "exportsite - Deployments", and a forward arrow, followed by the deployment name. The main area shows deployment details: "Deployment DATE" (7/5/2017 4:01:15 PM), "STATUS" (Succeeded), "DURATION" (1 minute 30 seconds), "RESOURCE GROUP" (exportsite), and "RELATED" (Events). On the far right, a "View template" button is highlighted with a red box.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-manager-export-template>



## Question #3

## Topic 6

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. Your company has an Azure subscription that includes a storage account, a resource group, a blob container and a file share. A fellow administrator named Jon Ross used an Azure Resource Manager template to deploy a virtual machine and an Azure Storage account. You need to identify the Azure Resource Manager template the Jon Ross used.

Solution: You access the Resource Group blade.

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer: A**

View template from deployment history

Go to the resource group for your new resource group. Notice that the portal shows the result of the last deployment. Select this link.

The screenshot shows the Azure Resource Group blade for the 'exportsite' group. The 'Essentials' section displays the subscription name as 'Microsoft Azure Consumption' and the subscription ID. To the right, a 'Deployments' box shows '1 Succeeded'. A red box highlights this 'Deployments' link.

You see a history of deployments for the group. In your case, the portal probably lists only one deployment. Select this deployment.

The screenshot shows the Azure Deployment blade. It lists a single deployment named 'Microsoft.WebSiteSQLDatabased1...' with a status of 'Succeeded'. A red box highlights the deployment name.

The portal displays a summary of the deployment. The summary includes the status of the deployment and its operations and the values that you provided for parameters. To see the template that you used for the deployment, select View template.

The screenshot shows the Azure Deployment blade for the 'Microsoft.WebSiteSQLDatabased13386b0-9908' deployment. It displays deployment details such as date, status, duration, and resource group. On the right, a 'View template' button is highlighted with a red box.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-manager-export-template>



## Question #4

## Topic 6

You have two Hyper-V hosts named Host1 and Host2. Host1 has an Azure virtual machine named VM1 that was deployed by using a custom Azure Resource Manager template.

You need to move VM1 to Host2.

What should you do?

- A. From the Update management blade, click Enable.
- B. From the Overview blade, move VM1 to a different subscription.
- C. From the Redeploy blade, click Redeploy.
- D. From the Profile blade, modify the usage location.

**Correct Answer:** *Answer: <map><m x1="57" x2="414" y1="161" y2="271" ss="0" a="0" /><m x1="59" x2="412" y1="285" y2="383" ss="0" a="0" /><m x1="57" x2="413" y1="395" />*

When you redeploy a VM, it moves the VM to a new node within the Azure infrastructure and then powers it back on, retaining all your configuration options and associated resources.

Reference:

<https://docs.microsoft.com/en-us/azure/virtual-machines/windows/redeploy-to-new-node>

#### Q5 DRAG DROP -

You have downloaded an Azure Resource Manager template to deploy numerous virtual machines. The template is based on a current virtual machine, but must be adapted to reference an administrative password.

You need to make sure that the password is not stored in plain text.

You are preparing to create the necessary components to achieve your goal.

Which of the following should you create to achieve your goal? Answer by dragging the correct option from the list to the answer area.

# Options

# Answer

An Azure Key Vault

An Azure Storage account

Azure Active Directory (AD)  
Identity Protection

An access policy

An Azure policy

A backup policy

y2="493" ss="0" a="0" /><m x1="60" x2="414" y1="507" y2="618" ss="0" a="0" /><m x1="56" x2="410" y1="631" y2="732" ss="0" a="0" /><m x1="58" x2="415" y1="741" y2="843" ss="0" a="0" /><m x1="457" x2="865" y1="162" y2="271" ss="1" a="0" /><m x1="457" x2="863" y1="283"

y2="386" ss="1" a="0" /><c start="0" stop="0" /><c start="3" stop="1" /></map>

# Options

An Azure Storage account

Azure Active Directory (AD)  
Identity Protection

An Azure policy

A backup policy

# Answer

An Azure Key Vault

An access policy

Question #5

Topic 6

Your company has an Azure Kubernetes Service (AKS) cluster that you manage from an Azure AD-joined device. The cluster is located in a resource group.

Developers have created an application named MyApp. MyApp was packaged into a container image.

You need to deploy the YAML manifest file for the application.

Solution: You install the Azure CLI on the device and run the kubectl apply `"-f myapp.yaml` command.

Does this meet the goal?

A. Yes

B. No

**Correct Answer:** A 

kubectl apply -f myapp.yaml applies a configuration change to a resource from a file or stdin.

Reference:

<https://kubernetes.io/docs/reference/kubectl/overview/>

<https://docs.microsoft.com/en-us/cli/azure/aks>

## Question #6

Topic 6

Your company has an Azure Kubernetes Service (AKS) cluster that you manage from an Azure AD-joined device. The cluster is located in a resource group.

Developers have created an application named MyApp. MyApp was packaged into a container image.

You need to deploy the YAML manifest file for the application.

Solution: You install the docker client on the device and run the docker run -it microsoft/azure-cli:0.10.17 command.

Does this meet the goal?

A. Yes

B. No

**Correct Answer:** B 

## Question #7

Topic 6

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 are developing a web app named mywebapp1. Mywebapp1 uses the address myapp1.azurewebsites.net. You protect mywebapp1 by implementing an Azure

Web Application Firewall (WAF). The traffic to mywebapp1 is routed through an Azure Application Gateway instance that is also used by other web apps.

You want to secure all traffic to mywebapp1 by using SSL.

Solution: You open the Azure Application Gateway's HTTP setting and set the Override backend path option to mywebapp1.azurewebsites.net. You then enable the Use for App service option.

Does this meet the goal?

A. Yes

B. No

**Correct Answer:** A 

The ability to specify a host override is defined in the HTTP settings and can be applied to any back-end pool during rule creation.

The ability to derive the host name from the IP or FQDN of the back-end pool members. HTTP settings also provide an option to dynamically pick the host name from a back-end pool member's FQDN if configured with the option to derive host name from an individual back-end pool member.

SSL termination and end to end SSL with multi-tenant services.

In case of end to end SSL, trusted Azure services such as Azure App service web apps do not require whitelisting the backends in the application gateway.

Therefore, there is no need to add any authentication certificates.

Reference:

<https://docs.microsoft.com/en-us/azure/application-gateway/application-gateway-web-app-overview>

## Question #8

Topic 6

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 are developing a web app named mywebapp1. Mywebapp1 uses the address myapp1.azurewebsites.net. You protect mywebapp1 by implementing an Azure

Web Application Firewall (WAF). The traffic to mywebapp1 is routed through an Azure Application Gateway instance that is also used by other web apps.

You want to secure all traffic to mywebapp1 by using SSL.

Solution: You configure mywebapp1 to run in an Azure App service environment (ASE).

Does this meet the goal?

A. Yes

B. No

**Correct Answer: B**

The Azure App service environment (ASE) is used to run an app in an isolated environment.

Reference:

<https://docs.microsoft.com/en-us/azure/app-service/environment/intro>

## Question #9

Topic 6

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 are developing a web app named mywebapp1. Mywebapp1 uses the address myapp1.azurewebsites.net. You protect mywebapp1 by implementing an Azure

Web Application Firewall (WAF). The traffic to mywebapp1 is routed through an Azure Application Gateway instance that is also used by other web apps.

You want to secure all traffic to mywebapp1 by using SSL.

Solution: You open the Azure Application Gateway's HTTP setting and set the Override backend path option to mywebapp1.azurewebsites.net. You then add an authentication certificate for mywebapp1.azurewebsites.net.

Does this meet the goal?

A. Yes

B. No

**Correct Answer: B**

In case of end to end SSL, trusted Azure services such as Azure App service web apps do not require whitelisting the backends in the application gateway.

Therefore, there is no need to add any authentication certificates.

Reference:

<https://docs.microsoft.com/en-us/azure/application-gateway/application-gateway-web-app-overview>

## Question #10

Topic 6

Your company has a web app named WebApp1.

You use the WebJobs SDK to design a triggered App Service background task that automatically invokes a function in the code every time new data is received in a queue.

You are preparing to configure the service processes a queue data item.

Which of the following is the service you should use?

- A. Logic Apps
- B. WebJobs
- C. Flow
- D. Functions

**Correct Answer:** B 

Reference:

<https://docs.microsoft.com/en-us/azure/azure-functions/functions-compare-logic-apps-ms-flow-webjobs>

## Question #11

Topic 6

Your company has an Azure subscription.

You need to deploy a number of Azure virtual machines to the subscription by using Azure Resource Manager (ARM) templates. The virtual machines will be included in a single availability set.

You need to ensure that the ARM template allows for as many virtual machines as possible to remain accessible in the event of fabric failure or maintenance.

Which of the following is the value that you should configure for the platformFaultDomainCount property?

- A. 10
- B. 30
- C. Min Value
- D. Max Value

**Correct Answer:** D 

The number of fault domains for managed availability sets varies by region - either two or three per region.

Reference:

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

## Question #12

Topic 6

Your company has an Azure subscription.

You need to deploy a number of Azure virtual machines to the subscription by using Azure Resource Manager (ARM) templates. The virtual machines will be included in a single availability set.

You need to ensure that the ARM template allows for as many virtual machines as possible to remain accessible in the event of fabric failure or maintenance.

Which of the following is the value that you should configure for the platformUpdateDomainCount property?

- A. 10
- B. 20
- C. 30
- D. 40

**Correct Answer:** D 

Each virtual machine in your availability set is assigned an update domain and a fault domain by the underlying Azure platform. For a given availability set, five non-user-configurable update domains are assigned by default (Resource Manager deployments can then be increased to provide up to 20 update domains) to indicate groups of virtual machines and underlying physical hardware that can be rebooted at the same time.

Reference:

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

## Question #13

Topic 6

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 are designing an Azure WebJob that will run on the same instances as a web app.

You want to make use of a suitable WebJob type. The webjob type should also allow for the option to restrict the WebJob to a single instance.

Solution: You configure the use of the Triggered WebJob type.

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer:** B 

Reference:

<https://docs.microsoft.com/en-us/azure/app-service/webjobs-create#webjob-types>

Question #14

Topic 6

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 are designing an Azure WebJob that will run on the same instances as a web app.

You want to make use of a suitable WebJob type. The webjob type should also allow for the option to restrict the WebJob to a single instance.

Solution: You configure the use of the Continuous WebJob type.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer:** Answer: <map><m x1="17" x2="310" y1="132" y2="223" ss="0" a="0" /><m x1="20" x2="309" y1="234" y2="316" ss="0" a="0" /><m x1="21" x2="312" y1="323" />

Reference:

<https://docs.microsoft.com/en-us/azure/app-service/webjobs-create#webjob-types>

Topic 2, Develop for Azure storage

Q16 DRAG DROP -

You are creating an Azure Cosmos DB account that makes use of the SQL API. Data will be added to the account every day by a web application.

You need to ensure that an email notification is sent when information is received from IoT devices, and that compute cost is reduced.

You decide to deploy a function app.

Which of the following should you configure the function app to use? Answer by dragging the correct option/s from the list to the answer area.

## Options

Azure Cosmos DB connector

SendGrid action

Consumption plan

Azure Event Hubs binding

SendGrid binding

## Answer

y2="408" ss="0" a="0" /><m

x1="19" x2="310" y1="414" y2="502" ss="0" a="0" /><m x1="20" x2="310" y1="509" y2="598" ss="0" a="0" /><m x1="348" x2="629" y1="131"

y2="219" ss="1" a="0" /><m x1="348" x2="628" y1="234" y2="320" ss="1" a="0" /><c start="2" stop="0" /><c start="4" stop="1" /></map>

# Options

Azure Cosmos DB connector

SendGrid action

Azure Event Hubs binding

# Answer

Consumption plan

SendGrid binding

## Question #15

## Topic 6

This question requires that you evaluate the underlined text to determine if it is correct.

You company has an on-premises deployment of MongoDB, and an Azure Cosmos DB account that makes use of the MongoDB API.

You need to devise a strategy to migrate MongoDB to the Azure Cosmos DB account.

You include the Data Management Gateway tool in your migration strategy.

Instructions: Review the underlined text. If it makes the statement correct, select 'No change required.' If the statement is incorrect, select the answer choice that makes the statement correct.

- A. No change required
- B. mongorestore
- C. Azure Storage Explorer
- D. AzCopy

**Correct Answer:** B 

Reference:

<https://docs.microsoft.com/en-us/azure/cosmos-db/mongodb-migrate> <https://docs.mongodb.com/manual/reference/program/mongorestore/>

## Question #16

## Topic 6

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 are developing an application that processes Azure Blob storage events.

Your application has the following requirements:

Process transaction logs asynchronously for changes that occur to the blobs and the blob metadata.

Process changes in the order in which they occurred.

Retain changes for compliance reasons.

Solution: You use Azure Event Grid with a subscriber Azure Function app.

Does the solution meet the goal?

- A. Yes
- B. No

**Correct Answer:** A 

Reference:

<https://docs.microsoft.com/en-us/azure/event-grid/overview>

<https://docs.microsoft.com/en-us/azure/event-grid/event-handlers#azure-functions>

## Question #17

Topic 6

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 are developing an application that processes Azure Blob storage events.

Your application has the following requirements:

Process transaction logs asynchronously for changes that occur to the blobs and the blob metadata.

Process changes in the order in which they occurred.

Retain changes for compliance reasons.

Solution: You use Azure Monitor HTTP Data Collector API.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer: B** 

## Question #18

Topic 6

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 are developing a mobile app that uses an Azure SQL Database named Weyland.

The database contains a table named Customers that has a field named email\_address.

You want to implement dynamic data masking to hide the data in the email\_address field.

Solution: You run the following Transact-SQL statement:

```
ALTER TABLE [dbo].[Weyland].[Customers]
```

```
ALTER COLUMN [email_address]
```

```
ADD MASKED WITH (FUNCTION = 'email()')
```

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer: A** 

Reference:

<https://docs.microsoft.com/en-us/sql/t-sql/statements/alter-table-transact-sql?view=sql-server-ver15> <https://docs.microsoft.com/en-us/sql/relational-databases/security/dynamic-data-masking?view=sql-server-ver15>

## Question #19

Topic 6

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 are developing a mobile app that uses an Azure SQL Database named Weyland.

The database contains a table named Customers that has a field named email\_address.

You want to implement dynamic data masking to hide the data in the email\_address field.

Solution: You run the Set-AzSqlDatabaseDataMaskingPolicy -DatabaseName "Weyland" Powershell cmdlet

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer: B**

The Set-AzSqlDatabaseDataMaskingPolicy cmdlet sets the masking policy for the database but does not set the masking rule.

Note: Masking is enabled by default.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-sql/database/dynamic-data-masking-overview> <https://docs.microsoft.com/en-us/powershell/module/az.sql/set-azsqlbasedatamaskingrule?view=azps-4.2.0>

## Question #20

Topic 6

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 are developing a mobile app that uses an Azure SQL Database named Weyland.

The database contains a table named Customers that has a field named email\_address.

You want to implement dynamic data masking to hide the data in the email\_address field.

Solution: You run the Set-AzSqlDatabaseDataMaskingRule

-DatabaseName "Weyland" -SchemaName "dbo"

-TableName "Customers" -ColumnName "email\_address"

-MaskingFunction "email" Powershell cmdlet

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer: A**

Reference:

<https://docs.microsoft.com/en-us/powershell/module/az.sql/Set-AzSqlDatabaseDataMaskingRule?view=azps-4.2.0>

Topic 3, Implement Azure security

Question #21

Topic 6

You are developing an e-Commerce Web App.

You want to use Azure Key Vault to ensure that sign-ins to the e-Commerce Web App are secured by using Azure App Service authentication and Azure Active Directory (AAD).

What should you do on the e-Commerce Web App?

- A. Run the az keyvault secret command.
- B. Enable Azure AD Connect.
- C. Enable Managed Service Identity (MSI).
- D. Create an Azure AD service principal.

**Correct Answer:** C 

A managed identity from Azure Active Directory allows your app to easily access other AAD-protected resources such as Azure Key Vault.

Reference:

<https://docs.microsoft.com/en-us/azure/app-service/overview-managed-identity> <https://docs.microsoft.com/en-us/samples/azure-samples/app-service-msi-keyvault-dotnet/keyvault-msi-appservice-sample/>

Question #22

Topic 6

You are developing a web app that uses Azure Active Directory (Azure AD) for authentication.

You want to configure the web app to use multifactor authentication.

What should you do?

- A. Enable mobile app authentication.
- B. In Azure AD conditional access, enable the baseline policy.
- C. In Azure AD, create a conditional access policy.
- D. Install the Azure Multi-Factor Authentication Server.

**Correct Answer:** C 

MFA is enabled by conditional access policy. It is the most flexible means to enable two-step verification for your users. Enabling using conditional access policy only works for Azure MFA in the cloud and is a premium feature of Azure AD.

Reference:

<https://docs.microsoft.com/en-us/azure/active-directory/authentication/howto-mfa-getstarted>

Question #23

Topic 6

This question requires that you evaluate the underlined text to determine if it is correct.

Your Azure Active Directory Azure (Azure AD) tenant has an Azure subscription linked to it.

Your developer has created a mobile application that obtains Azure AD access tokens using the OAuth 2 implicit grant type.

The mobile application must be registered in Azure AD.

You require a redirect URI from the developer for registration purposes.

Instructions: Review the underlined text. If it makes the statement correct, select `No change is needed.` If the statement is incorrect, select the answer choice that makes the statement correct.

- A. No change required.
- B. a secret
- C. a login hint
- D. a client ID

**Correct Answer:** A 

For Native Applications you need to provide a Redirect URI, which Azure AD will use to return token responses.

Reference:

<https://docs.microsoft.com/en-us/azure/active-directory/develop/v1-protocols-oauth-code>

Question #24

Topic 6

You are creating an Azure key vault using PowerShell. Objects deleted from the key vault must be kept for a set period of 90 days.

Which two of the following parameters must be used in conjunction to meet the requirement? (Choose two.)

- A. EnabledForDeployment
- B. EnablePurgeProtection
- C. EnabledForTemplateDeployment
- D. EnableSoftDelete

**Correct Answer:** Answer: <map><m x1="38" x2="303" y1="245" y2="305" ss="0" a="0" /><m x1="38" x2="303" y1="307" y2="365" ss="0" a="0" /><m x1="39" x2="301" y1="483" />

Reference:

<https://docs.microsoft.com/en-us/powershell/module/azurerm.keyvault/new-azurermkeyvault> <https://docs.microsoft.com/en-us/azure/key-vault/key-vault-ovw-soft-delete>

Q27 HOTSPOT -

You have an Azure Active Directory (Azure AD) tenant.

You want to implement multi-factor authentication by making use of a conditional access policy. The conditional access policy must be applied to all users when they access the Azure portal.

Which three settings should you configure? To answer, select the appropriate settings to the answer area.

NOTE: Each correct selection is worth one point.

## Answer Area

The screenshot shows the 'Require MFA for Azure portal' conditional access policy configuration page. The title bar has 'Require MFA for Azure port...' and standard window controls. Below the title bar are 'Info' and 'Delete' buttons. The main area starts with a 'Name' field containing 'MFA required for Azure portal access'. Under the 'Assignments' section, there are three expandable sections: 'Users and groups', 'Cloud apps', and 'Conditions'. Each section shows '0 users selected', '0 apps selected', and '0 conditions selected' respectively. Under the 'Access controls' section, there are two expandable sections: 'Grant' and 'Session'. Each section shows '0 controls selected'. The right side of the page has some code snippets: '<map><m x1="38" x2="303" y1="245" y2="305" ss="0" a="0" /><m x1="38" x2="303" y1="307" y2="365" ss="0" a="0" /><m x1="39" x2="301" y1="483" />' and 'y2="543" ss="0" a="0" /></map>'.

## Answer Area

Require MFA for Azure port...

\* Name  
MFA required for Azure portal access

Assignments

Users and groups  >  
0 users selected

Cloud apps  >  
0 apps selected

Conditions  >  
0 conditions selected

---

Access controls

Grant  >  
0 controls selected

Session  >

Box 1:

The conditional access policy must be applied or assigned to Users and Groups.

Box 2:

The conditional access policy must be applied when users access the Azure portal, which is a cloud app. That is: Microsoft Azure Management

Box 3:

Access control must require multi-factor authentication when granting access.

Reference:

<https://docs.microsoft.com/en-us/azure/active-directory/conditional-access/app-based-mfa>

## Question #25

Topic 6

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. Your company has an Azure Active Directory (Azure AD) environment. Users occasionally connect to Azure AD via the Internet. You need to ensure that users who connect to Azure AD via the internet using an unidentified IP address, are automatically instructed to change their passwords.

Solution: You configure the use of Azure Key Vault.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer:** B 

Reference:

<https://docs.microsoft.com/en-us/azure/active-directory/identity-protection/howto-sign-in-risk-policy>

## Question #26

Topic 6

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. Your company has an Azure Active Directory (Azure AD) environment. Users occasionally connect to Azure AD via the Internet. You need to ensure that users who connect to Azure AD via the internet using an unidentified IP address, are automatically instructed to change their passwords.

Solution: You configure the use of Azure AD Identity Protection.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer:** A 

Reference:

<https://docs.microsoft.com/en-us/azure/active-directory/identity-protection/howto-sign-in-risk-policy>

## Question #27

## Topic 6

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.

Your company has an Azure Active Directory (Azure AD) environment. Users occasionally connect to Azure AD via the Internet. You need to ensure that users who connect to Azure AD via the internet using an unidentified IP address, are automatically instructed to change their passwords.

Solution: You configure the use of Azure AD Privileged Identity Management.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer:** B 

Reference:

<https://docs.microsoft.com/en-us/azure/active-directory/identity-protection/howto-sign-in-risk-policy>

Question #28

Topic 6

You manage an Azure SQL database that allows for Azure AD authentication.

You need to make sure that database developers can connect to the SQL database via Microsoft SQL Server Management Studio (SSMS). You also need to make sure the developers use their on-premises Active Directory account for authentication. Your strategy should allow for authentication prompts to be kept to a minimum.

Which of the following should you implement?

- A. Azure AD token.
- B. Azure Multi-Factor authentication.
- C. Active Directory integrated authentication.
- D. OATH software tokens.

**Correct Answer:** C 

Azure AD can be the initial Azure AD managed domain. Azure AD can also be an on-premises Active Directory Domain Services that is federated with the Azure AD.

Using an Azure AD identity to connect using SSMS or SSDT

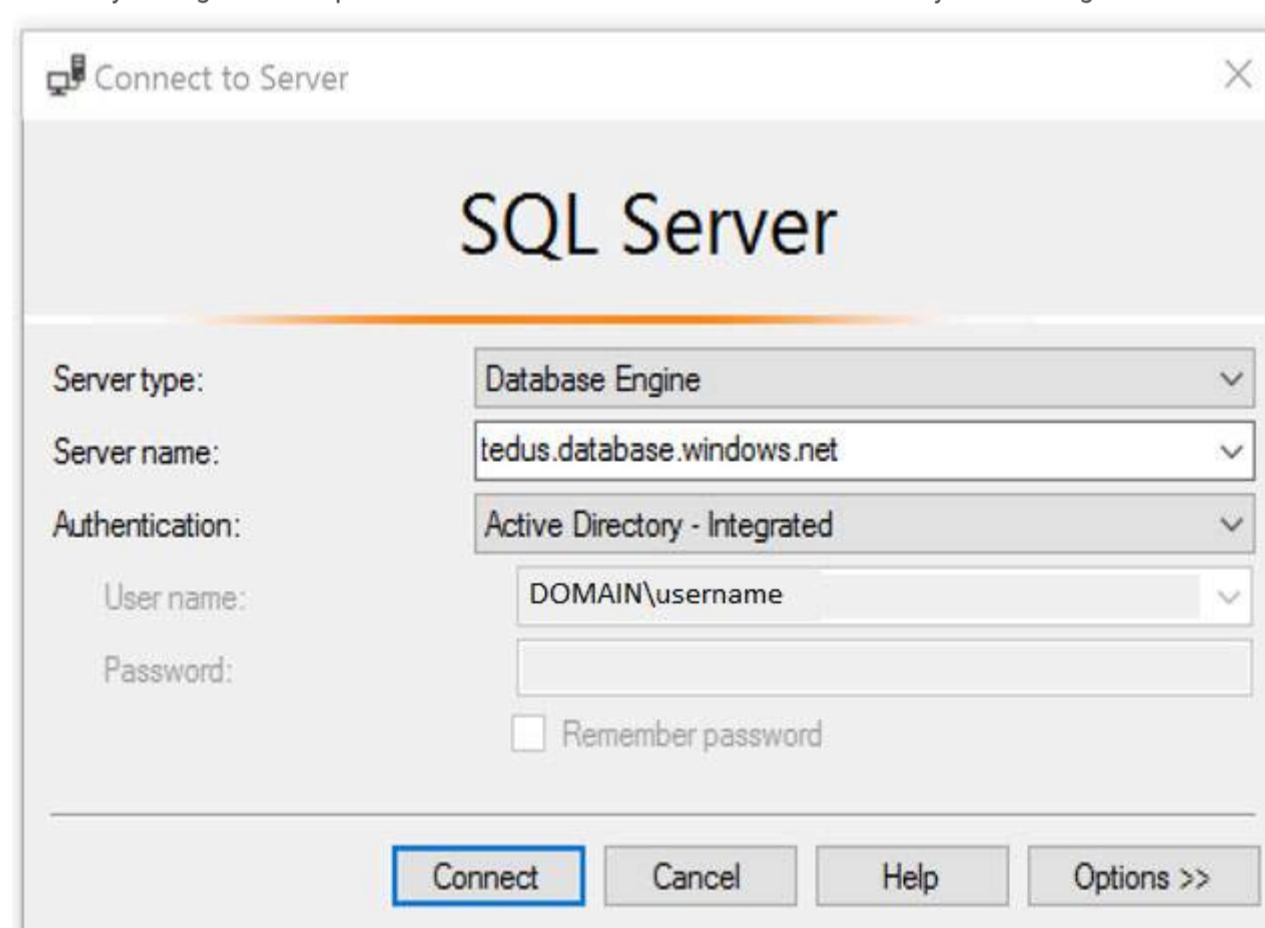
The following procedures show you how to connect to a SQL database with an Azure AD identity using SQL Server Management Studio or SQL Server Database Tools.

Active Directory integrated authentication

Use this method if you are logged in to Windows using your Azure Active Directory credentials from a federated domain.

1. Start Management Studio or Data Tools and in the Connect to Server (or Connect to Database Engine) dialog box, in the Authentication box, select Active

Directory - Integrated. No password is needed or can be entered because your existing credentials will be presented for the connection.



2. Select the Options button, and on the Connection Properties page, in the Connect to database box, type the name of the user database you want to connect to.

(The AD domain name or tenant ID option is only supported for Universal with MFA connection options, otherwise it is greyed out.)

Reference:

<https://github.com/MicrosoftDocs/azure-docs/blob/master/articles/sql-database/sql-database-aad-authentication-configure.md>



Question #29

Topic 6

You are developing an application to transfer data between on-premises file servers and Azure Blob storage. The application stores keys, secrets, and certificates in Azure Key Vault and makes use of the Azure Key Vault APIs. You want to configure the application to allow recovery of an accidental deletion of the key vault or key vault objects for 90 days after deletion. What should you do?

- A. Run the Add-AzKeyVaultKey cmdlet.
- B. Run the az keyvault update --enable-soft-delete true --enable-purge-protection true CLI.
- C. Implement virtual network service endpoints for Azure Key Vault.
- D. Run the az keyvault update --enable-soft-delete false CLI.

**Correct Answer:** Answer: <map><m x1="13" x2="172" y1="117" y2="140" ss="0" a="0" /><m x1="272" x2="432" y1="138" y2="162" ss="0" a="0" /></map> 

When soft-delete is enabled, resources marked as deleted resources are retained for a specified period (90 days by default). The service further provides a mechanism for recovering the deleted object, essentially undoing the deletion.

Purge protection is an optional Key Vault behavior and is not enabled by default. Purge protection can only be enabled once soft-delete is enabled.

When purge protection is on, a vault or an object in the deleted state cannot be purged until the retention period has passed. Soft-deleted vaults and objects can still be recovered, ensuring that the retention policy will be followed.

The default retention period is 90 days, but it is possible to set the retention policy interval to a value from 7 to 90 days through the Azure portal. Once the retention policy interval is set and saved it cannot be changed for that vault.

Reference:

<https://docs.microsoft.com/en-us/azure/key-vault/general/overview-soft-delete>

Topic 4, Monitor, troubleshoot, and optimize Azure solutions

Q33 HOTSPOT -

You have developed a Web App for your company. The Web App provides services and must run in multiple regions.

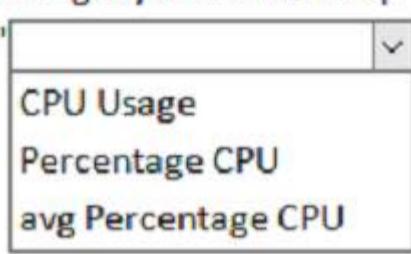
You want to be notified whenever the Web App uses more than 85 percent of the available CPU cores over a 5 minute period. Your solution must minimize costs.

Which command should you use? To answer, select the appropriate settings to the answer area.

NOTE: Each correct selection is worth one point.

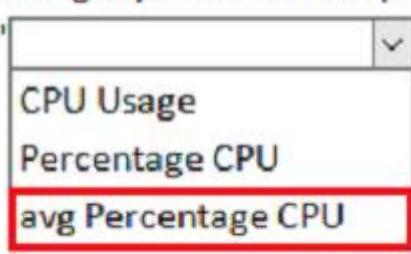
### Answer Area

```
az monitor metrics alert create -n myAlert -g myResourceGroup
--scopes targetResourceId --condition " > 85"
--window-size 5m
--evaluation-frequency
--auto-mitigate
```



### Answer Area

```
az monitor metrics alert create -n myAlert -g myResourceGroup
--scopes targetResourceId --condition " > 85"
--window-size
--evaluation-frequency
--auto-mitigate
```



Reference:

<https://docs.microsoft.com/sv-se/cli/azure/monitor/metrics/alert>

## Question #30

Topic 6

Note: The question is included in a number of questions that depicts the identical set-up. However, every question has a distinctive result.

Establish if the solution satisfies the requirements.

You are configuring a web app that delivers streaming video to users. The application makes use of continuous integration and deployment.

You need to ensure that the application is highly available and that the users' streaming experience is constant. You also want to configure the application to store data in a geographic location that is nearest to the user.

Solution: You include the use of Azure Redis Cache in your design.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer:** B 

## Question #31

Topic 6

Note: The question is included in a number of questions that depicts the identical set-up. However, every question has a distinctive result.

Establish if the solution satisfies the requirements.

You are configuring a web app that delivers streaming video to users. The application makes use of continuous integration and deployment.

You need to ensure that the application is highly available and that the users' streaming experience is constant. You also want to configure the application to store data in a geographic location that is nearest to the user.

Solution: You include the use of an Azure Content Delivery Network (CDN) in your design.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer:** A 

Reference:

<https://docs.microsoft.com/en-in/azure/cdn/>

## Question #32

Topic 6

Note: The question is included in a number of questions that depicts the identical set-up. However, every question has a distinctive result.

Establish if the solution satisfies the requirements.

You are configuring a web app that delivers streaming video to users. The application makes use of continuous integration and deployment.

You need to ensure that the application is highly available and that the users' streaming experience is constant. You also want to configure the application to store data in a geographic location that is nearest to the user.

Solution: You include the use of a Storage Area Network (SAN) in your design.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer:** B 

## Question #33

Topic 6

You develop a Web App on a tier D1 app service plan.

You notice that page load times increase during periods of peak traffic.

You want to implement automatic scaling when CPU load is above 80 percent. Your solution must minimize costs.

What should you do first?

- A. Enable autoscaling on the Web App.
- B. Switch to the Premium App Service tier plan.
- C. Switch to the Standard App Service tier plan.
- D. Switch to the Azure App Services consumption plan.

**Correct Answer:** C 

Configure the web app to the Standard App Service Tier. The Standard tier supports auto-scaling, and we should minimize the cost. We can then enable autoscaling on the web app, add a scale rule and add a Scale condition.

Reference:

<https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoring-autoscale-get-started> <https://azure.microsoft.com/en-us/pricing/details/app-service/plans/>

## Question #34

Topic 6

Your company's Azure subscription includes an Azure Log Analytics workspace.

Your company has a hundred on-premises servers that run either Windows Server 2012 R2 or Windows Server 2016, and is linked to the Azure Log Analytics workspace. The Azure Log Analytics workspace is set up to gather performance counters associated with security from these linked servers.

You must configure alerts based on the information gathered by the Azure Log Analytics workspace.

You have to make sure that alert rules allow for dimensions, and that alert creation time should be kept to a minimum. Furthermore, a single alert notification must be created when the alert is created and when the alert is resolved.

You need to make use of the necessary signal type when creating the alert rules.

Which of the following is the option you should use?

- A. The Activity log signal type.
- B. The Application Log signal type.
- C. The Metric signal type.
- D. The Audit Log signal type.

**Correct Answer:** C 

Metric alerts in Azure Monitor provide a way to get notified when one of your metrics cross a threshold. Metric alerts work on a range of multi-dimensional platform metrics, custom metrics, Application Insights standard and custom metrics.

Note: Signals are emitted by the target resource and can be of several types. Metric, Activity log, Application Insights, and Log.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-monitor/platform/alerts-metric>

Topic 5, Connect to and Consume Azure Services and Third-party Services

## Question #35

Topic 6

You are developing a .NET Core MVC application that allows customers to research independent holiday accommodation providers. You want to implement Azure Search to allow the application to search the index by using various criteria to locate documents related to accommodation. You want the application to allow customers to search the index by using regular expressions. What should you do?

- A. Configure the SearchMode property of the SearchParameters class.
- B. Configure the QueryType property of the SearchParameters class.
- C. Configure the Facets property of the SearchParameters class.
- D. Configure the Filter property of the SearchParameters class.

**Correct Answer:** B 

The SearchParameters.QueryType Property gets or sets a value that specifies the syntax of the search query. The default is 'simple'. Use 'full' if your query uses the Lucene query syntax.

You can write queries against Azure Search based on the rich Lucene Query Parser syntax for specialized query forms: wildcard, fuzzy search, proximity search, regular expressions are a few examples.

Reference:

<https://docs.microsoft.com/en-us/dotnet/api/microsoft.azure.search.models.searchparameters> <https://docs.microsoft.com/en-us/dotnet/api/microsoft.azure.search.models.searchparameters.querytype>

## Question #36

Topic 6

You are a developer at your company. You need to update the definitions for an existing Logic App. What should you use?

- A. the Enterprise Integration Pack (EIP)
- B. the Logic App Code View
- C. the API Connections
- D. the Logic Apps Designer

**Correct Answer:** B 

Edit JSON - Azure portal -

Sign in to the Azure portal.

From the left menu, choose All services. In the search box, find "logic apps", and then from the results, select your logic app.

On your logic app's menu, under Development Tools, select Logic App Code View.

The Code View editor opens and shows your logic app definition in JSON format.

Reference:

<https://docs.microsoft.com/en-us/azure/logic-apps/logic-apps-enterprise-integration-overview> <https://docs.microsoft.com/en-us/azure/logic-apps/logic-apps-author-definitions>

## Question #37

Topic 6

Note: The question is included in a number of questions that depicts the identical set-up. However, every question has a distinctive result.

Establish if the solution satisfies the requirements.

You are developing a solution for a public facing API.

The API back end is hosted in an Azure App Service instance. You have implemented a RESTful service for the API back end.

You must configure back-end authentication for the API Management service instance.

Solution: You configure Basic gateway credentials for the Azure resource.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer:** B 

API Management allows to secure access to the back-end service of an API using client certificates.

Reference:

<https://docs.microsoft.com/en-us/rest/api/apimanagement/apimanagementrest/azure-api-management-rest-api-backend-entity>

## Question #38

Topic 6

Note: The question is included in a number of questions that depicts the identical set-up. However, every question has a distinctive result.

Establish if the solution satisfies the requirements.

You are developing a solution for a public facing API.

The API back end is hosted in an Azure App Service instance. You have implemented a RESTful service for the API back end.

You must configure back-end authentication for the API Management service instance.

Solution: You configure Client cert gateway credentials for the HTTP(s) endpoint.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer:** B 

The API back end is hosted in an Azure App Service instance. It is an Azure resource and not an HTTP(s) endpoint.

Reference:

<https://docs.microsoft.com/en-us/rest/api/apimanagement/apimanagementrest/azure-api-management-rest-api-backend-entity>

## Question #39

Topic 6

Note: The question is included in a number of questions that depicts the identical set-up. However, every question has a distinctive result.

Establish if the solution satisfies the requirements.

You are developing a solution for a public facing API.

The API back end is hosted in an Azure App Service instance. You have implemented a RESTful service for the API back end.

You must configure back-end authentication for the API Management service instance.

Solution: You configure Basic gateway credentials for the HTTP(s) endpoint.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer:** *B* 

API Management allows to secure access to the back-end service of an API using client certificates. Furthermore, the API back end is hosted in an Azure App

Service instance. It is an Azure resource and not an HTTP(s) endpoint.

Reference:

<https://docs.microsoft.com/en-us/rest/api/apimanagement/apimanagementrest/azure-api-management-rest-api-backend-entity>

## Question #40

Topic 6

Note: The question is included in a number of questions that depicts the identical set-up. However, every question has a distinctive result.

Establish if the solution satisfies the requirements.

You are developing a solution for a public facing API.

The API back end is hosted in an Azure App Service instance. You have implemented a RESTful service for the API back end.

You must configure back-end authentication for the API Management service instance.

Solution: You configure Client cert gateway credentials for the Azure resource.

Does the solution meet the goal?

A. Yes

B. No

**Correct Answer:** *A* 

API Management allows to secure access to the back-end service of an API using client certificates.

Reference:

<https://docs.microsoft.com/en-us/rest/api/apimanagement/apimanagementrest/azure-api-management-rest-api-backend-entity>

Question #41

Topic 6

You are developing a .NET Core MVC application that allows customers to research independent holiday accommodation providers. You want to implement Azure Search to allow the application to search the index by using various criteria to locate documents related to accommodation venues. You want the application to list holiday accommodation venues that fall within a specific price range and are within a specified distance to an airport.

What should you do?

- A. Configure the SearchMode property of the SearchParameters class.
- B. Configure the QueryType property of the SearchParameters class.
- C. Configure the Facets property of the SearchParameters class.
- D. Configure the Filter property of the SearchParameters class.

**Correct Answer:** D 

The Filter property gets or sets the OData \$filter expression to apply to the search query.

Reference:

<https://docs.microsoft.com/en-us/dotnet/api/microsoft.azure.search.models.searchparameters> <https://docs.microsoft.com/en-us/dotnet/api/microsoft.azure.search.models.searchparameters.querytype>

Question #42

Topic 6

You are a developer at your company.  
You need to edit the workflows for an existing Logic App.  
What should you use?

- A. the Enterprise Integration Pack (EIP)
- B. the Logic App Code View
- C. the API Connections
- D. the Logic Apps Designer

**Correct Answer:** Answer: <map><m x1="8" x2="308" y1="39" y2="92" ss="0" a="0" /><m x1="8" x2="307" y1="103" y2="155" ss="0" a="0" /><m x1="8" x2="308" y1="167" />

For business-to-business (B2B) solutions and seamless communication between organizations, you can build automated scalable enterprise integration workflows by using the Enterprise Integration Pack (EIP) with Azure Logic Apps.

Reference:

<https://docs.microsoft.com/en-us/azure/logic-apps/logic-apps-enterprise-integration-overview> <https://docs.microsoft.com/en-us/azure/logic-apps/logic-apps-author-definitions>

Q47 DRAG DROP -

You are a developer for a company that provides a bookings management service in the tourism industry. You are implementing Azure Search for the tour agencies listed in your company's solution.

You create the index in Azure Search. You now need to use the Azure Search .NET SDK to import the relevant data into the Azure Search service.

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.

#### Answer Area

Create a DataSource instance and set its Container property to the DataContainer.

Create an IndexBatch that contains the documents which must be added.

Set the DataSources property of the SearchServiceClient.

Create a SearchIndexClient object to connect to the search index.

Call the Documents.Index method of the SearchIndexClient and pass the IndexBatch.

Call the Documents.Suggest method of the SearchIndexClient and pass the DataSource.

y2="217" ss="0" a="0" /><m x1="8"

x2="306" y1="228" y2="282" ss="0" a="0" /><m x1="9" x2="309" y1="291" y2="342" ss="0" a="0" /><m x1="7" x2="307" y1="354" y2="406" ss="0" a="0" /><m x1="351" x2="650" y1="40" y2="94" ss="1" a="0" /><m x1="350" x2="650" y1="102" y2="154" ss="1" a="0" /><m x1="352" x2="652" y1="166" y2="219" ss="1" a="0" /><c start="3" stop="0" /><c start="1" stop="1" /><c start="4" stop="2" /></map>

**Answer Area**

Create a DataSource instance and set its Container property to the DataContainer.

Set the DataSources property of the SearchServiceClient.

Call the Documents.Suggest method of the SearchIndexClient and pass the DataSource.

Create a SearchIndexClient object to connect to the search index.

Create an IndexBatch that contains the documents which must be added.

Call the Documents.Index method of the SearchIndexClient and pass the IndexBatch.

1. The index needs to be populated. To do this, we will need a SearchIndexClient. There are two ways to obtain one: by constructing it, or by calling

Indexes.GetClient on the SearchServiceClient. Here we will use the first method.

2. Create the indexBatch with the documents

Something like:

```
var hotels = new Hotel[];  
{  
    new Hotel()  
    {  
        HotelId = "3",  
        BaseRate = 129.99,  
        Description = "Close to town hall and the river"  
    }  
};  
var batch = IndexBatch.Upload(hotels);
```

3. The next step is to populate the newly-created index

Example:

```
var batch = IndexBatch.Upload(hotels);  
try  
{  
    indexClient.Documents.Index(batch);  
}
```

Reference:

<https://docs.microsoft.com/en-us/azure/search/search-howto-dotnet-sdk>

Question #43

Topic 6

You are developing an application that applies a set of governance policies for internal and external services, as well as for applications.

You develop a stateful ASP.NET Core 2.1 web application named PolicyApp and deploy it to an Azure App Service Web App. The PolicyApp reacts to events from

Azure Event Grid and performs policy actions based on those events.

You have the following requirements:

Authentication events must be used to monitor users when they sign in and sign out.

All authentication events must be processed by PolicyApp.

Sign outs must be processed as fast as possible.

What should you do?

- A. Create a new Azure Event Grid subscription for all authentication events. Use the subscription to process sign-out events.
- B. Create a separate Azure Event Grid handler for sign-in and sign-out events.
- C. Create separate Azure Event Grid topics and subscriptions for sign-in and sign-out events.
- D. Add a subject prefix to sign-out events. Create an Azure Event Grid subscription. Configure the subscription to use the subjectBeginsWith filter.

**Correct Answer:** D 

Reference:

<https://docs.microsoft.com/en-us/azure/event-grid/subscription-creation-schema>

**Topic 7 - Testlet 1**

Question #1

Topic 7

## Introductory Info

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Current environment -

Windows Server 2016 virtual machine

This virtual machine (VM) runs BizTalk Server 2016. The VM runs the following workflows:

Ocean Transport `" This workflow gathers and validates container information including container contents and arrival notices at various shipping ports.

Inland Transport `" This workflow gathers and validates trucking information including fuel usage, number of stops, and routes.

The VM supports the following REST API calls:

Container API `" This API provides container information including weight, contents, and other attributes.

Location API `" This API provides location information regarding shipping ports of call and trucking stops.

Shipping REST API `" This API provides shipping information for use and display on the shipping website.

Shipping Data -

The application uses MongoDB JSON document storage database for all container and transport information.

Shipping Web Site -

The site displays shipping container tracking information and container contents. The site is located at <http://shipping.wideworldimporters.com/>

Proposed solution -

The on-premises shipping application must be moved to Azure. The VM has been migrated to a new Standard\_D16s\_v3 Azure VM by using Azure Site Recovery and must remain running in Azure to complete the BizTalk component migrations. You create a Standard\_D16s\_v3 Azure VM to host BizTalk Server. The Azure architecture diagram for the proposed solution is shown below:



Requirements -

Shipping Logic app -

The Shipping Logic app must meet the following requirements:

Support the ocean transport and inland transport workflows by using a Logic App.

Support industry-standard protocol X12 message format for various messages including vessel content details and arrival notices.

Secure resources to the corporate VNet and use dedicated storage resources with a fixed costing model.

Maintain on-premises connectivity to support legacy applications and final BizTalk migrations.

Shipping Function app -

Implement secure function endpoints by using app-level security and include Azure Active Directory (Azure AD).

REST APIs -

The REST API's that support the solution must meet the following requirements:

Secure resources to the corporate VNet.

Allow deployment to a testing location within Azure while not incurring additional costs.

Automatically scale to double capacity during peak shipping times while not causing application downtime.

Minimize costs when selecting an Azure payment model.

Shipping data -

Data migration from on-premises to Azure must minimize costs and downtime.

Shipping website -

Use Azure Content Delivery Network (CDN) and ensure maximum performance for dynamic content while minimizing latency and costs.

Issues -

Windows Server 2016 VM -

The VM shows high network latency, jitter, and high CPU utilization. The VM is critical and has not been backed up in the past. The VM must enable a quick restore from a 7-day snapshot to include in-place restore of disks in case of failure.

Shipping website and REST APIs -

The following error message displays while you are testing the website:

Failed to load http://test-shippingapi.wideworldimporters.com/: No 'Access-Control-Allow-Origin' header is present on the requested resource. Origin 'http://test.wideworldimporters.com/' is therefore not allowed access.

### Question

HOTSPOT -

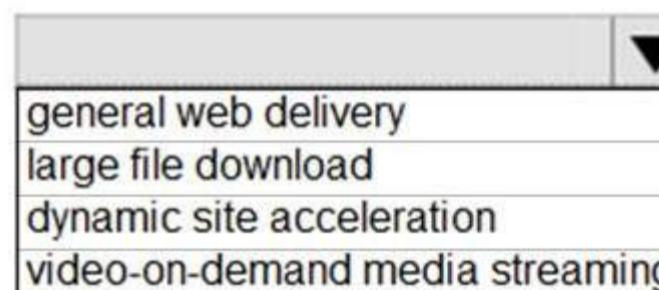
You need to configure Azure CDN for the Shipping web site.

Which configuration options should you use? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

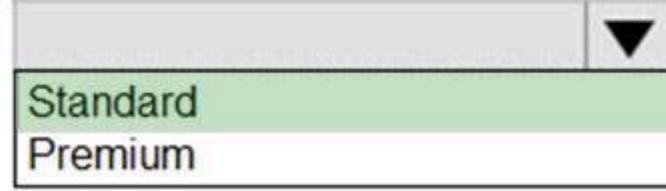
Hot Area:

## Answer Area

Option	Value
Tier	
Profile	
Optimization	

## Answer Area

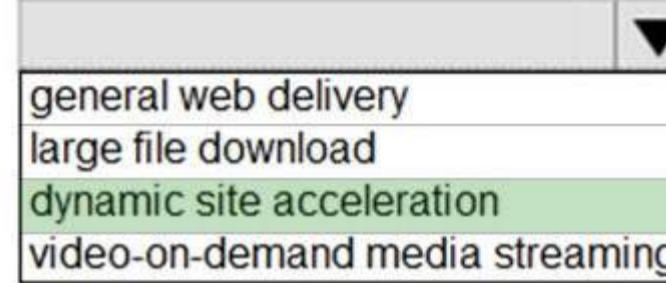
Option	Value
--------	-------

Tier	
------	--

Standard  
Premium

Correct Answer:	Profile
-----------------	---------


Akamai
Microsoft

Optimization	
--------------	---

general web delivery  
large file download  
**dynamic site acceleration**  
video-on-demand media streaming

Scenario: Shipping website -

Use Azure Content Delivery Network (CDN) and ensure maximum performance for dynamic content while minimizing latency and costs.

Tier: Standard -

Profile: Akamai -

Optimization: Dynamic site acceleration

Dynamic site acceleration (DSA) is available for Azure CDN Standard from Akamai, Azure CDN Standard from Verizon, and Azure CDN Premium from Verizon profiles.

DSA includes various techniques that benefit the latency and performance of dynamic content. Techniques include route and network optimization, TCP optimization, and more.

You can use this optimization to accelerate a web app that includes numerous responses that aren't cacheable. Examples are search results, checkout transactions, or real-time data. You can continue to use core Azure CDN caching capabilities for static data.

Reference:

<https://docs.microsoft.com/en-us/azure/cdn/cdn-optimization-overview>

Question #2

Topic 7

## Introductory Info

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Current environment -

Windows Server 2016 virtual machine

This virtual machine (VM) runs BizTalk Server 2016. The VM runs the following workflows:

Ocean Transport `" This workflow gathers and validates container information including container contents and arrival notices at various shipping ports.

Inland Transport `" This workflow gathers and validates trucking information including fuel usage, number of stops, and routes.

The VM supports the following REST API calls:

Container API `" This API provides container information including weight, contents, and other attributes.

Location API `" This API provides location information regarding shipping ports of call and trucking stops.

Shipping REST API `" This API provides shipping information for use and display on the shipping website.

Shipping Data -

The application uses MongoDB JSON document storage database for all container and transport information.

Shipping Web Site -

The site displays shipping container tracking information and container contents. The site is located at <http://shipping.wideworldimporters.com/>

Proposed solution -

The on-premises shipping application must be moved to Azure. The VM has been migrated to a new Standard\_D16s\_v3 Azure VM by using Azure Site Recovery and must remain running in Azure to complete the BizTalk component migrations. You create a Standard\_D16s\_v3 Azure VM to host BizTalk Server. The Azure architecture diagram for the proposed solution is shown below:



Requirements -

Shipping Logic app -

The Shipping Logic app must meet the following requirements:

Support the ocean transport and inland transport workflows by using a Logic App.

Support industry-standard protocol X12 message format for various messages including vessel content details and arrival notices.

Secure resources to the corporate VNet and use dedicated storage resources with a fixed costing model.

Maintain on-premises connectivity to support legacy applications and final BizTalk migrations.

Shipping Function app -

Implement secure function endpoints by using app-level security and include Azure Active Directory (Azure AD).

REST APIs -

The REST API's that support the solution must meet the following requirements:

Secure resources to the corporate VNet.

Allow deployment to a testing location within Azure while not incurring additional costs.

Automatically scale to double capacity during peak shipping times while not causing application downtime.

Minimize costs when selecting an Azure payment model.

Shipping data -

Data migration from on-premises to Azure must minimize costs and downtime.

Shipping website -

Use Azure Content Delivery Network (CDN) and ensure maximum performance for dynamic content while minimizing latency and costs.

Issues -

Windows Server 2016 VM -

The VM shows high network latency, jitter, and high CPU utilization. The VM is critical and has not been backed up in the past. The VM must enable a quick restore from a 7-day snapshot to include in-place restore of disks in case of failure.

Shipping website and REST APIs -

The following error message displays while you are testing the website:

Failed to load http://test-shippingapi.wideworldimporters.com/: No 'Access-Control-Allow-Origin' header is present on the requested resource. Origin 'http://test.wideworldimporters.com/' is therefore not allowed access.

### Question

HOTSPOT -

You need to correct the VM issues.

Which tools should you use? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

## Answer Area

Issue	Tool
Backup and Restore	Azure Site Recovery Azure Backup Azure Data Box Azure Migrate
Performance	Azure Network Watcher Azure Traffic Manager ExpressRoute Accelerated Networking

Azure Site Recovery
Azure Backup
Azure Data Box
Azure Migrate

Azure Network Watcher
Azure Traffic Manager
ExpressRoute
Accelerated Networking

## Answer Area

### Issue      Tool

Backup and Restore

Correct Answer:

Azure Site Recovery
<b>Azure Backup</b>
Azure Data Box
Azure Migrate

Performance

Azure Network Watcher
Azure Traffic Manager
ExpressRoute
<b>Accelerated Networking</b>

Box 1: Azure Backup -

### Topic 8 - Testlet 10

In-Place restore of disks in IaaS VMs is a feature of Azure Backup.

Performance: Accelerated Networking

Scenario: The VM shows high network latency, jitter, and high CPU utilization.

Box 2: Accelerated networking -

The VM shows high network latency, jitter, and high CPU utilization.

Accelerated networking enables single root I/O virtualization (SR-IOV) to a VM, greatly improving its networking performance. This high-performance path bypasses the host from the datapath, reducing latency, jitter, and CPU utilization, for use with the most demanding network workloads on supported VM types.

Reference:

<https://azure.microsoft.com/en-us/blog/an-easy-way-to-bring-back-your-azure-vm-with-in-place-restore/>

Question #1

Topic 8

## Introductory Info

### 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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

### Background -

#### Overview -

You are a developer for Contoso, Ltd. The company has a social networking website that is developed as a Single Page Application (SPA). The main web application for the social networking website loads user uploaded content from blob storage.

You are developing a solution to monitor uploaded data for inappropriate content. The following process occurs when users upload content by using the SPA:

- \* Messages are sent to ContentUploadService.
- \* Content is processed by ContentAnalysisService.
- \* After processing is complete, the content is posted to the social network or a rejection message is posted in its place.

The ContentAnalysisService is deployed with Azure Container Instances from a private Azure Container Registry named contosoimages.

The solution will use eight CPU cores.

### Azure Active Directory -

Contoso, Ltd. uses Azure Active Directory (Azure AD) for both internal and guest accounts.

### Requirements -

#### ContentAnalysisService -

The company's data science group built ContentAnalysisService which accepts user generated content as a string and returns a probable value for inappropriate content. Any values over a specific threshold must be reviewed by an employee of Contoso, Ltd.

You must create an Azure Function named CheckUserContent to perform the content checks.

### Costs -

You must minimize costs for all Azure services.

### Manual review -

To review content, the user must authenticate to the website portion of the ContentAnalysisService using their Azure AD credentials. The website is built using

React and all pages and API endpoints require authentication. In order to review content a user must be part of a ContentReviewer role. All completed reviews must include the reviewer's email address for auditing purposes.

### High availability -

All services must run in multiple regions. The failure of any service in a region must not impact overall application availability.

### Monitoring -

An alert must be raised if the ContentUploadService uses more than 80 percent of available CPU cores.

### Security -

You have the following security requirements:

Any web service accessible over the Internet must be protected from cross site scripting attacks.

All websites and services must use SSL from a valid root certificate authority.

Azure Storage access keys must only be stored in memory and must be available only to the service.

All Internal services must only be accessible from internal Virtual Networks (VNets).

All parts of the system must support inbound and outbound traffic restrictions.

All service calls must be authenticated by using Azure AD.

User agreements -

When a user submits content, they must agree to a user agreement. The agreement allows employees of Contoso, Ltd. to review content, store cookies on user devices, and track user's IP addresses.

Information regarding agreements is used by multiple divisions within Contoso, Ltd.

User responses must not be lost and must be available to all parties regardless of individual service uptime. The volume of agreements is expected to be in the millions per hour.

Validation testing -

When a new version of the ContentAnalysisService is available the previous seven days of content must be processed with the new version to verify that the new version does not significantly deviate from the old version.

Issues -

Users of the ContentUploadService report that they occasionally see HTTP 502 responses on specific pages.

Code -

ContentUploadService -

```
CS01 apiVersion: '2018-10-01'
CS02 type: Microsoft.ContainerInstance/containerGroups
CS03 location: westus
CS04 name: contentUploadService
CS05 properties:
CS06   containers:
CS07     - name: service
CS08       properties:
CS09         image: contoso/contentUploadService:latest
CS10       ports:
CS11         - port: 80
CS12           protocol: TCP
CS13       resources:
CS14         requests:
CS15           cpu: 1.0
CS16           memoryInGB: 1.5
CS17
CS18 ipAddress:
CS19   ip: 10.23.121.112
CS20   ports:
CS21     - port: 80
CS22       protocol: TCP
CS23
CS24
CS25 networkProfile:
CS26
id: /subscriptions/98...19/resourceGroups/container/providers/Microsoft.Network/networkProfiles/subnet
```

**ApplicationManifest -**

```
AM01 {  
AM02     "id" : "2b079f03-9b06-2d44-98bb-e9182901fcb6",  
AM03     "appId" : "7118a7f0-b5c2-4c9d-833c-3d711396fe65",  
AM04  
AM05     "createdDateTime" : "2019-12-24T06:01:44Z",  
AM06     "logoUrl" : null,  
AM07     "logoutUrl" : null,  
AM08     "name" : "ContentAnalysisService",  
AM09  
AM10  
AM11     "orgRestrictions" : [],  
AM12     "parentalControlSettings" : {  
AM13         "countriesBlockedForMinors" : [],  
AM14         "legalAgeGroupRule" : "Allow"  
AM15     },  
AM16     "passwordCredentials" : []  
AM17 }
```

**Question**

DRAG DROP -

You need to add markup at line AM04 to implement the ContentReview role.

How should you complete the markup? To answer, drag the appropriate json segments to the correct locations. Each json 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:

**Json segments****Answer Area**

User

`"appRoles": [`

value

`{ "": [`

role

`"": "`

Application

`],`

allowedMemberTypes

`"displayName": "ContentReviewer",`

allowedAccountTypes

`"id": "e1c2ade8-98f8-45fd-aa4a-6d24b512c22a",``"isEnabled" : true,``"": "ContentReviewer"``}``],`

**Correct Answer:**

### Json segments

User  
value  
role  
Application  
allowedMemberTypes  
allowedAccountTypes

### Answer Area

```
"appRoles": [
{
    "allowedMemberTypes": [
        "User"
    ],
    "displayName": "ContentReviewer",
    "id": "e1c2ade8-98f8-45fd-aa4a-6d24b512c22a",
    "isEnabled": true,
    "value": "ContentReviewer"
},
]
```

Box 1: allowedMemberTypes -

allowedMemberTypes specifies whether this app role definition can be assigned to users and groups by setting to "User", or to other applications (that are accessing this application in daemon service scenarios) by setting to "Application", or to both.

Note: The following example shows the appRoles that you can assign to users.

```
"appId": "8763f1c4-f988-489c-a51e-158e9ef97d6a",
"appRoles": [
{
    "allowedMemberTypes": [
        "User"
    ],
    "displayName": "Writer",
    "id": "d1c2ade8-98f8-45fd-aa4a-6d06b947c66f",
    "isEnabled": true,
    "description": "Writers Have the ability to create tasks.",
    "value": "Writer"
},
],
"availableToOtherTenants": false,
```

Box 2: User -

Scenario: In order to review content a user must be part of a ContentReviewer role.

Box 3: value -

value specifies the value which will be included in the roles claim in authentication and access tokens.

Reference:

<https://docs.microsoft.com/en-us/graph/api/resources/approle>

Question #2

Topic 8

## Introductory Info

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

Overview -

You are a developer for Contoso, Ltd. The company has a social networking website that is developed as a Single Page Application (SPA). The main web application for the social networking website loads user uploaded content from blob storage.

You are developing a solution to monitor uploaded data for inappropriate content. The following process occurs when users upload content by using the SPA:

- \* Messages are sent to ContentUploadService.
- \* Content is processed by ContentAnalysisService.
- \* After processing is complete, the content is posted to the social network or a rejection message is posted in its place.

The ContentAnalysisService is deployed with Azure Container Instances from a private Azure Container Registry named contosoimages.

The solution will use eight CPU cores.

Azure Active Directory -

Contoso, Ltd. uses Azure Active Directory (Azure AD) for both internal and guest accounts.

Requirements -

ContentAnalysisService -

The company's data science group built ContentAnalysisService which accepts user generated content as a string and returns a probable value for inappropriate content. Any values over a specific threshold must be reviewed by an employee of Contoso, Ltd.

You must create an Azure Function named CheckUserContent to perform the content checks.

Costs -

You must minimize costs for all Azure services.

Manual review -

To review content, the user must authenticate to the website portion of the ContentAnalysisService using their Azure AD credentials. The website is built using

React and all pages and API endpoints require authentication. In order to review content a user must be part of a ContentReviewer role. All completed reviews must include the reviewer's email address for auditing purposes.

High availability -

All services must run in multiple regions. The failure of any service in a region must not impact overall application availability.

Monitoring -

An alert must be raised if the ContentUploadService uses more than 80 percent of available CPU cores.

Security -

You have the following security requirements:

Any web service accessible over the Internet must be protected from cross site scripting attacks.

All websites and services must use SSL from a valid root certificate authority.

Azure Storage access keys must only be stored in memory and must be available only to the service.

All Internal services must only be accessible from internal Virtual Networks (VNets).

All parts of the system must support inbound and outbound traffic restrictions.

All service calls must be authenticated by using Azure AD.

User agreements -

When a user submits content, they must agree to a user agreement. The agreement allows employees of Contoso, Ltd. to review content, store cookies on user devices, and track user's IP addresses.

Information regarding agreements is used by multiple divisions within Contoso, Ltd.

User responses must not be lost and must be available to all parties regardless of individual service uptime. The volume of agreements is expected to be in the millions per hour.

Validation testing -

When a new version of the ContentAnalysisService is available the previous seven days of content must be processed with the new version to verify that the new version does not significantly deviate from the old version.

Issues -

Users of the ContentUploadService report that they occasionally see HTTP 502 responses on specific pages.

Code -

ContentUploadService -

```
CS01 apiVersion: '2018-10-01'
CS02 type: Microsoft.ContainerInstance/containerGroups
CS03 location: westus
CS04 name: contentUploadService
CS05 properties:
CS06   containers:
CS07     - name: service
CS08       properties:
CS09         image: contoso/contentUploadService:latest
CS10       ports:
CS11         - port: 80
CS12           protocol: TCP
CS13       resources:
CS14         requests:
CS15           cpu: 1.0
CS16           memoryInGB: 1.5
CS17
CS18 ipAddress:
CS19   ip: 10.23.121.112
CS20   ports:
CS21     - port: 80
CS22       protocol: TCP
CS23
CS24
CS25 networkProfile:
CS26
id: /subscriptions/98...19/resourceGroups/container/providers/Microsoft.Network/networkProfiles/subnet
```

**ApplicationManifest -**

```
AM01 {  
AM02     "id" : "2b079f03-9b06-2d44-98bb-e9182901fcb6",  
AM03     "appId" : "7118a7f0-b5c2-4c9d-833c-3d711396fe65",  
AM04  
AM05     "createdDateTime" : "2019-12-24T06:01:44Z",  
AM06     "logoUrl" : null,  
AM07     "logoutUrl" : null,  
AM08     "name" : "ContentAnalysisService",  
AM09  
AM10  
AM11     "orgRestrictions" : [],  
AM12     "parentalControlSettings" : {  
AM13         "countriesBlockedForMinors" : [],  
AM14         "legalAgeGroupRule" : "Allow"  
AM15     },  
AM16     "passwordCredentials" : []  
AM17 }
```

**Question****HOTSPOT -**

You need to add code at line AM09 to ensure that users can review content using ContentAnalysisService.

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

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

"allowPublicClient":true  
"oauth2Permissions": ["login"]  
"oauth2AllowUrlPathMatching":true  
"oauth2AllowIdTokenImplicitFlow":true

"oauth2AllowImplicitFlow": true  
"oauth2RequiredPostResponse":true  
"preAuthorizedApplications":["SPA"]  
"knownClientApplications":["ContentAnalysisService"]

## Answer Area

Correct Answer:

```
"allowPublicClient":true  
"oauth2Permissions": ["login"]  
"oauth2AllowUrlPathMatching":true  
"oauth2AllowIdTokenImplicitFlow":true
```

```
"oauth2AllowImplicitFlow": true  
"oauth2RequiredPostResponse":true  
"preAuthorizedApplications":["SPA"]  
"knownClientApplications":["ContentAnalysisService"]
```

Box 1: "oauth2Permissions": ["login"]

oauth2Permissions specifies the collection of OAuth 2.0 permission scopes that the web API (resource) app exposes to client apps. These permission scopes may be granted to client apps during consent.

Box 2: "oauth2AllowImplicitFlow":true

For applications (Angular, Ember.js, React.js, and so on), Microsoft identity platform supports the OAuth 2.0 Implicit Grant flow.

Reference:

<https://docs.microsoft.com/en-us/azure/active-directory/develop/reference-app-manifest>

Question #3

Topic 8

## Introductory Info

### 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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

### Background -

#### Overview -

You are a developer for Contoso, Ltd. The company has a social networking website that is developed as a Single Page Application (SPA). The main web application for the social networking website loads user uploaded content from blob storage.

You are developing a solution to monitor uploaded data for inappropriate content. The following process occurs when users upload content by using the SPA:

- \* Messages are sent to ContentUploadService.
- \* Content is processed by ContentAnalysisService.
- \* After processing is complete, the content is posted to the social network or a rejection message is posted in its place.

The ContentAnalysisService is deployed with Azure Container Instances from a private Azure Container Registry named contosoimages.

The solution will use eight CPU cores.

### Azure Active Directory -

Contoso, Ltd. uses Azure Active Directory (Azure AD) for both internal and guest accounts.

### Requirements -

#### ContentAnalysisService -

The company's data science group built ContentAnalysisService which accepts user generated content as a string and returns a probable value for inappropriate content. Any values over a specific threshold must be reviewed by an employee of Contoso, Ltd.

You must create an Azure Function named CheckUserContent to perform the content checks.

### Costs -

You must minimize costs for all Azure services.

### Manual review -

To review content, the user must authenticate to the website portion of the ContentAnalysisService using their Azure AD credentials. The website is built using

React and all pages and API endpoints require authentication. In order to review content a user must be part of a ContentReviewer role. All completed reviews must include the reviewer's email address for auditing purposes.

### High availability -

All services must run in multiple regions. The failure of any service in a region must not impact overall application availability.

### Monitoring -

An alert must be raised if the ContentUploadService uses more than 80 percent of available CPU cores.

### Security -

You have the following security requirements:

- Any web service accessible over the Internet must be protected from cross site scripting attacks.
- All websites and services must use SSL from a valid root certificate authority.
- Azure Storage access keys must only be stored in memory and must be available only to the service.
- All Internal services must only be accessible from internal Virtual Networks (VNets).
- All parts of the system must support inbound and outbound traffic restrictions.
- All service calls must be authenticated by using Azure AD.

User agreements -

When a user submits content, they must agree to a user agreement. The agreement allows employees of Contoso, Ltd. to review content, store cookies on user devices, and track user's IP addresses.

Information regarding agreements is used by multiple divisions within Contoso, Ltd.

User responses must not be lost and must be available to all parties regardless of individual service uptime. The volume of agreements is expected to be in the millions per hour.

Validation testing -

When a new version of the ContentAnalysisService is available the previous seven days of content must be processed with the new version to verify that the new version does not significantly deviate from the old version.

Issues -

Users of the ContentUploadService report that they occasionally see HTTP 502 responses on specific pages.

Code -

ContentUploadService -

```
CS01 apiVersion: '2018-10-01'
CS02 type: Microsoft.ContainerInstance/containerGroups
CS03 location: westus
CS04 name: contentUploadService
CS05 properties:
CS06   containers:
CS07     - name: service
CS08       properties:
CS09         image: contoso/contentUploadService:latest
CS10       ports:
CS11         - port: 80
CS12           protocol: TCP
CS13       resources:
CS14         requests:
CS15           cpu: 1.0
CS16           memoryInGB: 1.5
CS17
CS18 ipAddress:
CS19   ip: 10.23.121.112
CS20   ports:
CS21     - port: 80
CS22       protocol: TCP
CS23
CS24
CS25 networkProfile:
CS26
id: /subscriptions/98...19/resourceGroups/container/providers/Microsoft.Network/networkProfiles/subnet
```

**ApplicationManifest -**

```
AM01 {  
AM02     "id" : "2b079f03-9b06-2d44-98bb-e9182901fcb6",  
AM03     "appId" : "7118a7f0-b5c2-4c9d-833c-3d711396fe65",  
AM04  
AM05     "createdDateTime" : "2019-12-24T06:01:44Z",  
AM06     "logoUrl" : null,  
AM07     "logoutUrl" : null,  
AM08     "name" : "ContentAnalysisService",  
AM09  
AM10  
AM11     "orgRestrictions" : [],  
AM12     "parentalControlSettings" : {  
AM13         "countriesBlockedForMinors" : [],  
AM14         "legalAgeGroupRule" : "Allow"  
AM15     },  
AM16     "passwordCredentials" : []  
AM17 }
```

**Question****HOTSPOT -**

You need to ensure that network security policies are met.

How should you configure network security? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

Technology	Value
SSL certificate	<input type="checkbox"/> Valid root certificate <input type="checkbox"/> Self-signed certificate
Proxy type	<input type="checkbox"/> nginx <input type="checkbox"/> Azure Application Gateway

**Answer Area**

Technology	Value
Correct Answer: SSL certificate	<input checked="" type="checkbox"/> Valid root certificate <input type="checkbox"/> Self-signed certificate
Proxy type	<input type="checkbox"/> nginx <input checked="" type="checkbox"/> Azure Application Gateway

Box 1: Valid root certificate -

Scenario: All websites and services must use SSL from a valid root certificate authority.

Box 2: Azure Application Gateway

Scenario:

□ Any web service accessible over the Internet must be protected from cross site scripting attacks.

□ All Internal services must only be accessible from Internal Virtual Networks (VNets)

All parts of the system must support inbound and outbound traffic restrictions.

Azure Web Application Firewall (WAF) on Azure Application Gateway provides centralized protection of your web applications from common exploits and vulnerabilities. Web applications are increasingly targeted by malicious attacks that exploit commonly known vulnerabilities. SQL injection and cross-site scripting are among the most common attacks.

Application Gateway supports autoscaling, SSL offloading, and end-to-end SSL, a web application firewall (WAF), cookie-based session affinity, URL path-based routing, multisite hosting, redirection, rewrite HTTP headers and other features.

Note: Both Nginx and Azure Application Gateway act as a reverse proxy with Layer 7 load-balancing features plus a WAF to ensure strong protection against common web vulnerabilities and exploits.

You can modify Nginx web server configuration/SSL for X-XSS protection. This helps to prevent cross-site scripting exploits by forcing the injection of HTTP headers with X-XSS protection.

Reference:

<https://docs.microsoft.com/en-us/azure/web-application-firewall/ag/ag-overview> <https://www.upguard.com/articles/10-tips-for-securing-your-nginx-deployment>

Question #4

Topic 8

## Introductory Info

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

Overview -

You are a developer for Contoso, Ltd. The company has a social networking website that is developed as a Single Page Application (SPA). The main web application for the social networking website loads user uploaded content from blob storage.

You are developing a solution to monitor uploaded data for inappropriate content. The following process occurs when users upload content by using the SPA:

- \* Messages are sent to ContentUploadService.
- \* Content is processed by ContentAnalysisService.
- \* After processing is complete, the content is posted to the social network or a rejection message is posted in its place.

The ContentAnalysisService is deployed with Azure Container Instances from a private Azure Container Registry named contosoimages.

The solution will use eight CPU cores.

Azure Active Directory -

Contoso, Ltd. uses Azure Active Directory (Azure AD) for both internal and guest accounts.

Requirements -

ContentAnalysisService -

The company's data science group built ContentAnalysisService which accepts user generated content as a string and returns a probable value for inappropriate content. Any values over a specific threshold must be reviewed by an employee of Contoso, Ltd.

You must create an Azure Function named CheckUserContent to perform the content checks.

Costs -

You must minimize costs for all Azure services.

Manual review -

To review content, the user must authenticate to the website portion of the ContentAnalysisService using their Azure AD credentials. The website is built using

React and all pages and API endpoints require authentication. In order to review content a user must be part of a ContentReviewer role. All completed reviews must include the reviewer's email address for auditing purposes.

High availability -

All services must run in multiple regions. The failure of any service in a region must not impact overall application availability.

Monitoring -

An alert must be raised if the ContentUploadService uses more than 80 percent of available CPU cores.

Security -

You have the following security requirements:

Any web service accessible over the Internet must be protected from cross site scripting attacks.

All websites and services must use SSL from a valid root certificate authority.

Azure Storage access keys must only be stored in memory and must be available only to the service.

All Internal services must only be accessible from internal Virtual Networks (VNets).

All parts of the system must support inbound and outbound traffic restrictions.

All service calls must be authenticated by using Azure AD.

#### User agreements -

When a user submits content, they must agree to a user agreement. The agreement allows employees of Contoso, Ltd. to review content, store cookies on user devices, and track user's IP addresses.

Information regarding agreements is used by multiple divisions within Contoso, Ltd.

User responses must not be lost and must be available to all parties regardless of individual service uptime. The volume of agreements is expected to be in the millions per hour.

#### Validation testing -

When a new version of the ContentAnalysisService is available the previous seven days of content must be processed with the new version to verify that the new version does not significantly deviate from the old version.

#### Issues -

Users of the ContentUploadService report that they occasionally see HTTP 502 responses on specific pages.

#### Code -

##### ContentUploadService -

```
CS01 apiVersion: '2018-10-01'
CS02 type: Microsoft.ContainerInstance/containerGroups
CS03 location: westus
CS04 name: contentUploadService
CS05 properties:
CS06   containers:
CS07     - name: service
CS08       properties:
CS09         image: contoso/contentUploadService:latest
CS10       ports:
CS11         - port: 80
CS12           protocol: TCP
CS13       resources:
CS14         requests:
CS15           cpu: 1.0
CS16           memoryInGB: 1.5
CS17
CS18 ipAddress:
CS19   ip: 10.23.121.112
CS20   ports:
CS21     - port: 80
CS22       protocol: TCP
CS23
CS24
CS25 networkProfile:
CS26
id: /subscriptions/98...19/resourceGroups/container/providers/Microsoft.Network/networkProfiles/subnet
```

## ApplicationManifest -

```

AM01 {
AM02     "id" : "2b079f03-9b06-2d44-98bb-e9182901fcb6",
AM03     "appId" : "7118a7f0-b5c2-4c9d-833c-3d711396fe65",
AM04
AM05     "createdDateTime" : "2019-12-24T06:01:44Z",
AM06     "logoUrl" : null,
AM07     "logoutUrl" : null,
AM08     "name" : "ContentAnalysisService",
AM09
AM10
AM11     "orgRestrictions" : [],
AM12     "parentalControlSettings" : {
AM13         "countriesBlockedForMinors" : [],
AM14         "legalAgeGroupRule" : "Allow"
AM15     },
AM16     "passwordCredentials" : []
AM17 }

```

**Question**

DRAG DROP -

You need to add YAML markup at line CS17 to ensure that the ContentUploadService can access Azure Storage access keys.

How should you complete the YAML markup? To answer, drag the appropriate YAML segments to the correct locations. Each YAML 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:

**YAML segments**

secret

envVar

secretValues

volumes

volumeMounts

environmentVariables

**Answer Area**

YAML segment :

- mountPath: /mnt/secrets
- name: accesskey

YAML segment :

- name: accesskey

YAML segment :

key: TXkgZmlyc3Qgc2VjcmV0IEZPTwo=

**Correct Answer:****YAML segments**

envVar

secretValues

environmentVariables

**Answer Area**

volumeMounts :

- mountPath: /mnt/secrets
- name: accesskey

volumes :

- name: accesskey

secret :

key: TXkgZmlyc3Qgc2VjcmV0IEZPTwo=

Box 1: volumeMounts -

Example:

```
volumeMounts:  
- mountPath: /mnt/secrets  
name: secretvolume1  
volumes:  
- name: secretvolume1  
secret:  
mysecret1: TXkgZmlyc3Qgc2VjcmV0IEZPTwo=
```

Box 2: volumes -

Box 3: secret -

Reference:

<https://docs.microsoft.com/en-us/azure/container-instances/container-instances-volume-secret>

Question #5

Topic 8

## Introductory Info

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

Overview -

You are a developer for Contoso, Ltd. The company has a social networking website that is developed as a Single Page Application (SPA). The main web application for the social networking website loads user uploaded content from blob storage.

You are developing a solution to monitor uploaded data for inappropriate content. The following process occurs when users upload content by using the SPA:

- \* Messages are sent to ContentUploadService.
- \* Content is processed by ContentAnalysisService.
- \* After processing is complete, the content is posted to the social network or a rejection message is posted in its place.

The ContentAnalysisService is deployed with Azure Container Instances from a private Azure Container Registry named contosoimages.

The solution will use eight CPU cores.

Azure Active Directory -

Contoso, Ltd. uses Azure Active Directory (Azure AD) for both internal and guest accounts.

Requirements -

ContentAnalysisService -

The company's data science group built ContentAnalysisService which accepts user generated content as a string and returns a probable value for inappropriate content. Any values over a specific threshold must be reviewed by an employee of Contoso, Ltd.

You must create an Azure Function named CheckUserContent to perform the content checks.

Costs -

You must minimize costs for all Azure services.

Manual review -

To review content, the user must authenticate to the website portion of the ContentAnalysisService using their Azure AD credentials. The website is built using

React and all pages and API endpoints require authentication. In order to review content a user must be part of a ContentReviewer role. All completed reviews must include the reviewer's email address for auditing purposes.

High availability -

All services must run in multiple regions. The failure of any service in a region must not impact overall application availability.

Monitoring -

An alert must be raised if the ContentUploadService uses more than 80 percent of available CPU cores.

Security -

You have the following security requirements:

Any web service accessible over the Internet must be protected from cross site scripting attacks.

All websites and services must use SSL from a valid root certificate authority.

Azure Storage access keys must only be stored in memory and must be available only to the service.

All Internal services must only be accessible from internal Virtual Networks (VNets).

All parts of the system must support inbound and outbound traffic restrictions.

All service calls must be authenticated by using Azure AD.

User agreements -

When a user submits content, they must agree to a user agreement. The agreement allows employees of Contoso, Ltd. to review content, store cookies on user devices, and track user's IP addresses.

Information regarding agreements is used by multiple divisions within Contoso, Ltd.

User responses must not be lost and must be available to all parties regardless of individual service uptime. The volume of agreements is expected to be in the millions per hour.

Validation testing -

When a new version of the ContentAnalysisService is available the previous seven days of content must be processed with the new version to verify that the new version does not significantly deviate from the old version.

Issues -

Users of the ContentUploadService report that they occasionally see HTTP 502 responses on specific pages.

Code -

ContentUploadService -

```
CS01 apiVersion: '2018-10-01'
CS02 type: Microsoft.ContainerInstance/containerGroups
CS03 location: westus
CS04 name: contentUploadService
CS05 properties:
CS06   containers:
CS07     - name: service
CS08       properties:
CS09         image: contoso/contentUploadService:latest
CS10       ports:
CS11         - port: 80
CS12           protocol: TCP
CS13       resources:
CS14         requests:
CS15           cpu: 1.0
CS16           memoryInGB: 1.5
CS17
CS18 ipAddress:
CS19   ip: 10.23.121.112
CS20   ports:
CS21     - port: 80
CS22       protocol: TCP
CS23
CS24
CS25 networkProfile:
CS26
id: /subscriptions/98...19/resourceGroups/container/providers/Microsoft.Network/networkProfiles/subnet
```

**ApplicationManifest -**

```
AM01 {  
AM02     "id" : "2b079f03-9b06-2d44-98bb-e9182901fcb6",  
AM03     "appId" : "7118a7f0-b5c2-4c9d-833c-3d711396fe65",  
AM04  
AM05     "createdDateTime" : "2019-12-24T06:01:44Z",  
AM06     "logoUrl" : null,  
AM07     "logoutUrl" : null,  
AM08     "name" : "ContentAnalysisService",  
AM09  
AM10  
AM11     "orgRestrictions" : [],  
AM12     "parentalControlSettings" : {  
AM13         "countriesBlockedForMinors" : [],  
AM14         "legalAgeGroupRule" : "Allow"  
AM15     },  
AM16     "passwordCredentials" : []  
AM17 }
```

**Question****HOTSPOT -**

You need to add code at line AM10 of the application manifest to ensure that the requirement for manually reviewing content can be met.

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

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

```
"optionalClaims": [  
    "acct",  
    "pltf",  
    "sid",  
    "tenant_ctry",  
    "upn",  
    "email",  
    "enfpolids"],
```

## Answer Area

"optionalClaims": [  
    "  
        acct  
        platf  
        **sid**  
        tenant\_ctry  
    ",  
    "  
        **sid**  
        upn  
        **email**  
        enfpolids  
    "  
]

Correct Answer:

### Topic 9 - Testlet 11

Box 1: sid -

Sid: Session ID, used for per-session user sign-out. Personal and Azure AD accounts.

Scenario: Manual review -

To review content, the user must authenticate to the website portion of the ContentAnalysisService using their Azure AD credentials. The website is built using

React and all pages and API endpoints require authentication. In order to review content a user must be part of a ContentReviewer role.

Box 2: email -

Scenario: All completed reviews must include the reviewer's email address for auditing purposes.

Question #1

Topic 9

## Introductory Info

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Current environment -

Windows Server 2016 virtual machine

This virtual machine (VM) runs BizTalk Server 2016. The VM runs the following workflows:

Ocean Transport `" This workflow gathers and validates container information including container contents and arrival notices at various shipping ports.

Inland Transport `" This workflow gathers and validates trucking information including fuel usage, number of stops, and routes.

The VM supports the following REST API calls:

Container API `" This API provides container information including weight, contents, and other attributes.

Location API `" This API provides location information regarding shipping ports of call and trucking stops.

Shipping REST API `" This API provides shipping information for use and display on the shipping website.

Shipping Data -

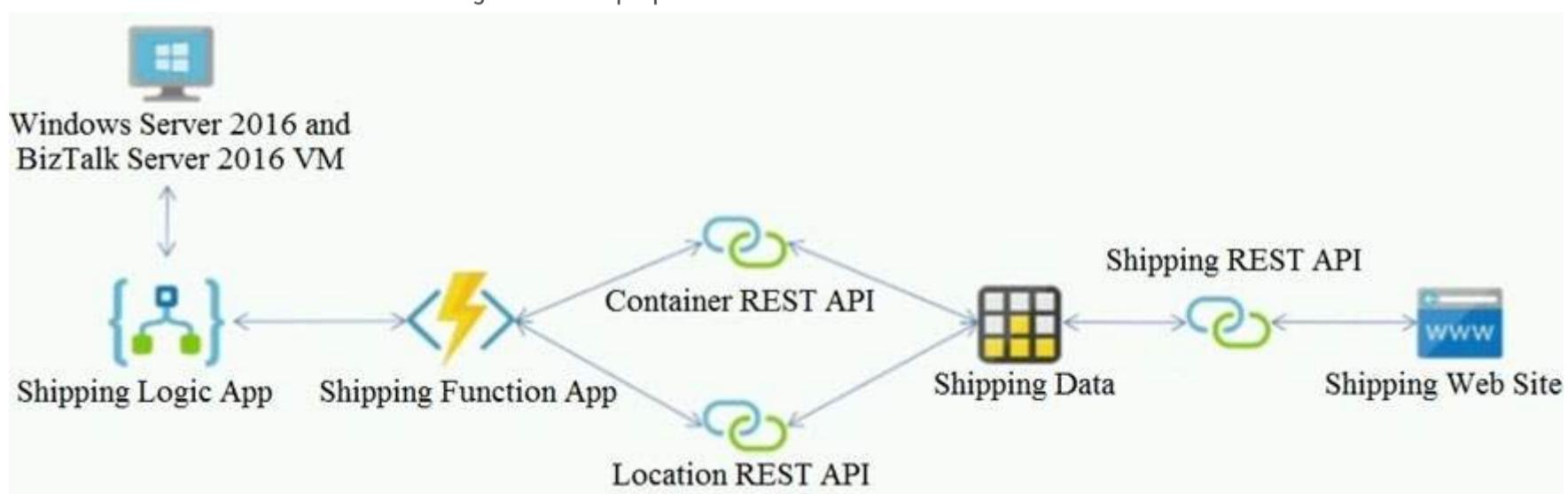
The application uses MongoDB JSON document storage database for all container and transport information.

Shipping Web Site -

The site displays shipping container tracking information and container contents. The site is located at <http://shipping.wideworldimporters.com/>

Proposed solution -

The on-premises shipping application must be moved to Azure. The VM has been migrated to a new Standard\_D16s\_v3 Azure VM by using Azure Site Recovery and must remain running in Azure to complete the BizTalk component migrations. You create a Standard\_D16s\_v3 Azure VM to host BizTalk Server. The Azure architecture diagram for the proposed solution is shown below:



Requirements -

Shipping Logic app -

The Shipping Logic app must meet the following requirements:

Support the ocean transport and inland transport workflows by using a Logic App.

Support industry-standard protocol X12 message format for various messages including vessel content details and arrival notices.

Secure resources to the corporate VNet and use dedicated storage resources with a fixed costing model.

Maintain on-premises connectivity to support legacy applications and final BizTalk migrations.

Shipping Function app -

Implement secure function endpoints by using app-level security and include Azure Active Directory (Azure AD).

REST APIs -

The REST API's that support the solution must meet the following requirements:

Secure resources to the corporate VNet.

Allow deployment to a testing location within Azure while not incurring additional costs.

Automatically scale to double capacity during peak shipping times while not causing application downtime.

Minimize costs when selecting an Azure payment model.

Shipping data -

Data migration from on-premises to Azure must minimize costs and downtime.

Shipping website -

Use Azure Content Delivery Network (CDN) and ensure maximum performance for dynamic content while minimizing latency and costs.

Issues -

Windows Server 2016 VM -

The VM shows high network latency, jitter, and high CPU utilization. The VM is critical and has not been backed up in the past. The VM must enable a quick restore from a 7-day snapshot to include in-place restore of disks in case of failure.

Shipping website and REST APIs -

The following error message displays while you are testing the website:

Failed to load http://test-shippingapi.wideworldimporters.com/: No 'Access-Control-Allow-Origin' header is present on the requested resource. Origin 'http://test.wideworldimporters.com/' is therefore not allowed access.

### Question

HOTSPOT -

You need to secure the Shipping Function app.

How should you configure the app? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

## Answer Area

Setting	Value
Authorization level	<div style="border: 1px solid black; padding: 5px; width: fit-content;"><p>Function</p><p>Anonymous</p><p>Admin</p></div>
User claims	<div style="border: 1px solid black; padding: 5px; width: fit-content;"><p>JSON Web Token (JWT)</p><p>Shared Access Signature (SAS) token</p><p>API Key</p></div>
Trigger type	<div style="border: 1px solid black; padding: 5px; width: fit-content;"><p>blob</p><p>HTTP</p><p>queue</p><p>timer</p></div>

## Answer Area

Setting	Value
---------	-------

Authorization level

Function
Anonymous
Admin

Correct Answer:

User claims

JSON Web Token (JWT)
Shared Access Signature (SAS) token
API Key

Trigger type

blob
HTTP
queue
timer

Scenario: Shipping Function app: Implement secure function endpoints by using app-level security and include Azure Active Directory (Azure AD).

Box 1: Function -

Box 2: JSON based Token (JWT)

Azure AD uses JSON based tokens (JWTs) that contain claims

Box 3: HTTP -

How a web app delegates sign-in to Azure AD and obtains a token

User authentication happens via the browser. The OpenID protocol uses standard HTTP protocol messages.

Reference:

<https://docs.microsoft.com/en-us/azure/active-directory/develop/authentication-scenarios>

Question #2

Topic 9

## Introductory Info

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Current environment -

Windows Server 2016 virtual machine

This virtual machine (VM) runs BizTalk Server 2016. The VM runs the following workflows:

Ocean Transport `" This workflow gathers and validates container information including container contents and arrival notices at various shipping ports.

Inland Transport `" This workflow gathers and validates trucking information including fuel usage, number of stops, and routes.

The VM supports the following REST API calls:

Container API `" This API provides container information including weight, contents, and other attributes.

Location API `" This API provides location information regarding shipping ports of call and trucking stops.

Shipping REST API `" This API provides shipping information for use and display on the shipping website.

Shipping Data -

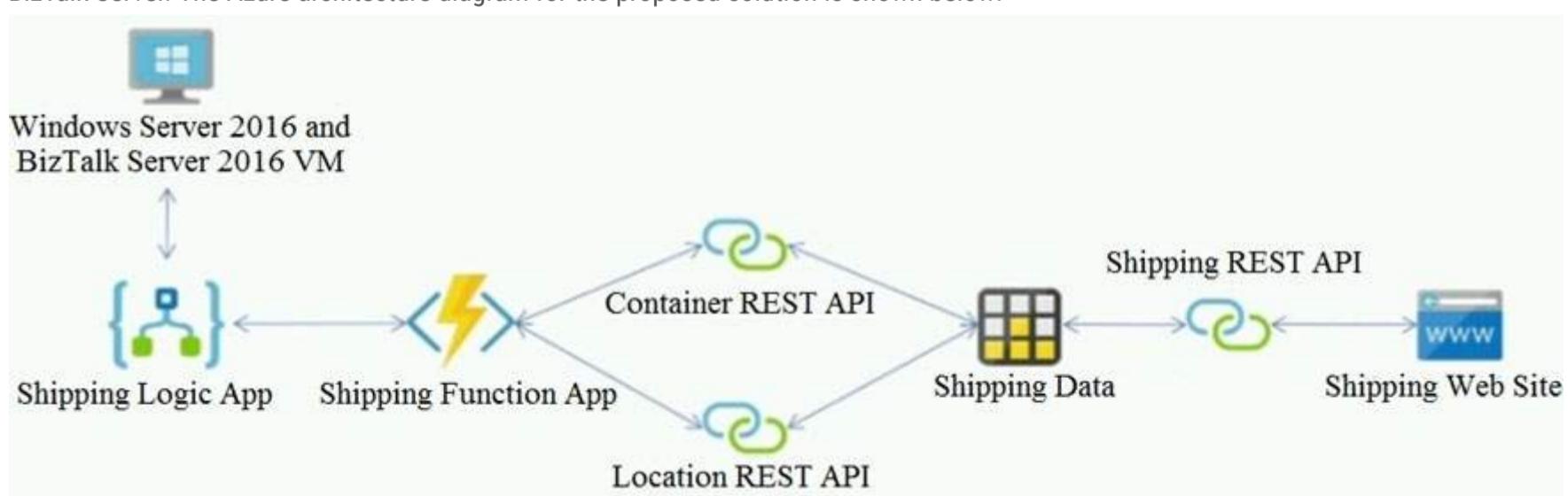
The application uses MongoDB JSON document storage database for all container and transport information.

Shipping Web Site -

The site displays shipping container tracking information and container contents. The site is located at <http://shipping.wideworldimporters.com/>

Proposed solution -

The on-premises shipping application must be moved to Azure. The VM has been migrated to a new Standard\_D16s\_v3 Azure VM by using Azure Site Recovery and must remain running in Azure to complete the BizTalk component migrations. You create a Standard\_D16s\_v3 Azure VM to host BizTalk Server. The Azure architecture diagram for the proposed solution is shown below:



Requirements -

Shipping Logic app -

The Shipping Logic app must meet the following requirements:

Support the ocean transport and inland transport workflows by using a Logic App.

Support industry-standard protocol X12 message format for various messages including vessel content details and arrival notices.

Secure resources to the corporate VNet and use dedicated storage resources with a fixed costing model.

Maintain on-premises connectivity to support legacy applications and final BizTalk migrations.

#### Shipping Function app -

Implement secure function endpoints by using app-level security and include Azure Active Directory (Azure AD).

#### REST APIs -

The REST API's that support the solution must meet the following requirements:

Secure resources to the corporate VNet.

Allow deployment to a testing location within Azure while not incurring additional costs.

- Automatically scale to double capacity during peak shipping times while not causing application downtime.

- Minimize costs when selecting an Azure payment model.

#### Shipping data -

Data migration from on-premises to Azure must minimize costs and downtime.

#### Shipping website -

Use Azure Content Delivery Network (CDN) and ensure maximum performance for dynamic content while minimizing latency and costs.

#### Issues -

##### Windows Server 2016 VM -

The VM shows high network latency, jitter, and high CPU utilization. The VM is critical and has not been backed up in the past. The VM must enable a quick restore from a 7-day snapshot to include in-place restore of disks in case of failure.

##### Shipping website and REST APIs -

The following error message displays while you are testing the website:

Failed to load http://test-shippingapi.wideworldimporters.com/: No 'Access-Control-Allow-Origin' header is present on the requested resource. Origin 'http://test.wideworldimporters.com/' is therefore not allowed access.

#### Question

You need to secure the Shipping Logic App.

What should you use?

- A. Azure App Service Environment (ASE)
- B. Integration Service Environment (ISE)
- C. VNet service endpoint
- D. Azure AD B2B integration

#### Correct Answer: B

Scenario: The Shipping Logic App requires secure resources to the corporate VNet and use dedicated storage resources with a fixed costing model.

You can access to Azure Virtual Network resources from Azure Logic Apps by using integration service environments (ISEs).

Sometimes, your logic apps and integration accounts need access to secured resources, such as virtual machines (VMs) and other systems or services, that are inside an Azure virtual network. To set up this access, you can create an integration service environment (ISE) where you can run your logic apps and create your integration accounts.

Reference:

<https://docs.microsoft.com/en-us/azure/logic-apps/connect-virtual-network-vnet-isolated-environment-overview>

#### Topic 10 - Testlet 12

Question #1

**Introductory Info**

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

City Power & Light company provides electrical infrastructure monitoring solutions for homes and businesses. The company is migrating solutions to Azure.

Current environment -

Architecture overview -

The company has a public website located at <http://www.cpndl.com/>. The site is a single-page web application that runs in Azure App Service on Linux. The website uses files stored in Azure Storage and cached in Azure Content Delivery Network (CDN) to serve static content.

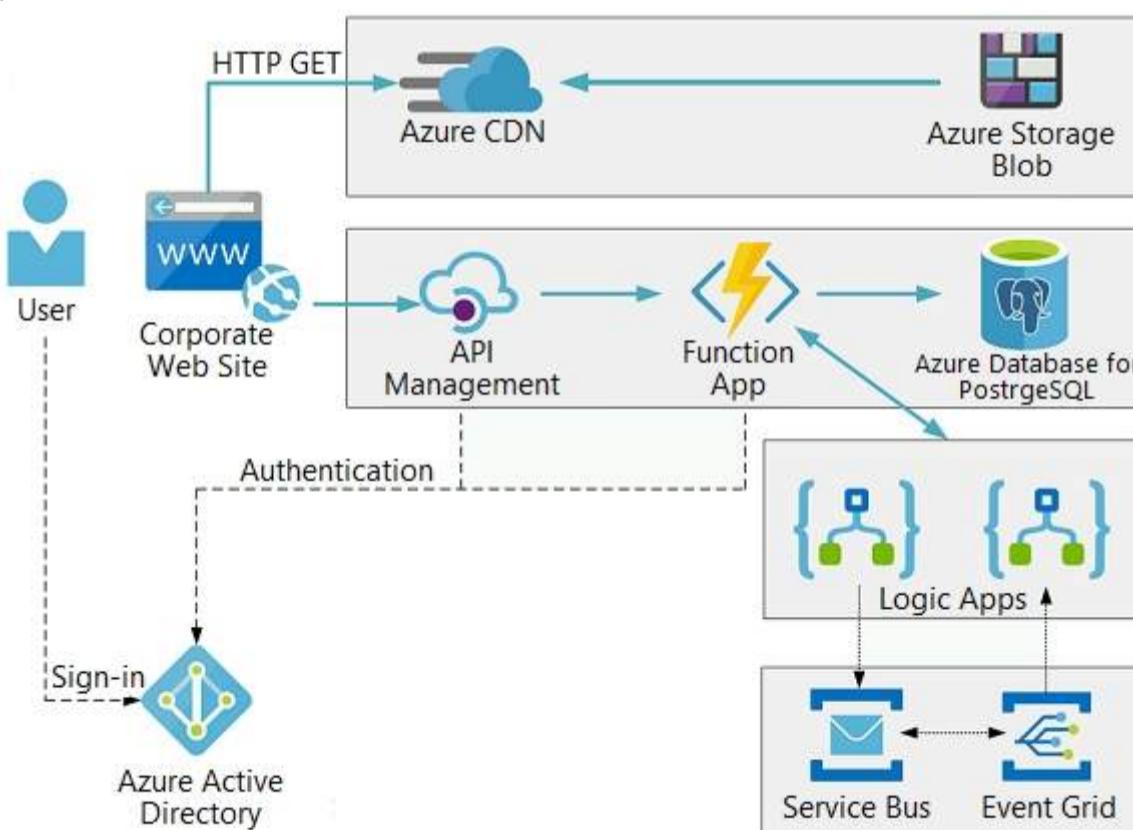
API Management and Azure Function App functions are used to process and store data in Azure Database for PostgreSQL. API Management is used to broker communications to the Azure Function app functions for Logic app integration. Logic apps are used to orchestrate the data processing while Service Bus and

Event Grid handle messaging and events.

The solution uses Application Insights, Azure Monitor, and Azure Key Vault.

Architecture diagram -

The company has several applications and services that support their business. The company plans to implement serverless computing where possible. The overall architecture is shown below.



User authentication -

The following steps detail the user authentication process:

1. The user selects Sign in in the website.
2. The browser redirects the user to the Azure Active Directory (Azure AD) sign in page.

3. The user signs in.
4. Azure AD redirects the user's session back to the web application. The URL includes an access token.
5. The web application calls an API and includes the access token in the authentication header. The application ID is sent as the audience ('`aud`') claim in the access token.
6. The back-end API validates the access token.

Requirements -

Corporate website -

Communications and content must be secured by using SSL.

Communications must use HTTPS.

Data must be replicated to a secondary region and three availability zones.

Data storage costs must be minimized.

Azure Database for PostgreSQL -

The database connection string is stored in Azure Key Vault with the following attributes:

Azure Key Vault name: cpandlkeyvault

Secret name: PostgreSQLConn

Id: 80df3e46ffcd4f1cb187f79905e9a1e8

The connection information is updated frequently. The application must always use the latest information to connect to the database.

Azure Service Bus and Azure Event Grid

Azure Event Grid must use Azure Service Bus for queue-based load leveling.

Events in Azure Event Grid must be routed directly to Service Bus queues for use in buffering.

Events from Azure Service Bus and other Azure services must continue to be routed to Azure Event Grid for processing.

Security -

All SSL certificates and credentials must be stored in Azure Key Vault.

File access must restrict access by IP, protocol, and Azure AD rights.

All user accounts and processes must receive only those privileges which are essential to perform their intended function.

Compliance -

Auditing of the file updates and transfers must be enabled to comply with General Data Protection Regulation (GDPR). The file updates must be read-only, stored in the order in which they occurred, include only create, update, delete, and copy operations, and be retained for compliance reasons.

Issues -

Corporate website -

While testing the site, the following error message displays:

CryptographicException: The system cannot find the file specified.

Function app -

You perform local testing for the RequestUserApproval function. The following error message displays:

'Timeout value of 00:10:00 exceeded by function: RequestUserApproval'

The same error message displays when you test the function in an Azure development environment when you run the following Kusto query:

FunctionAppLogs -

```
| where FunctionName == "RequestUserApproval"
```

Logic app -

You test the Logic app in a development environment. The following error message displays:

'400 Bad Request'

Troubleshooting of the error shows an HttpTrigger action to call the RequestUserApproval function.

Code -

Corporate website -

Security.cs:

```

SC01 public class Security
SC02 {
SC03 var bytes = System.IO.File.ReadAllBytes("~/var/ssl/private");
SC04 var cert = new System.Security.Cryptography.X509Certificate2(bytes);
SC05 var certName = cert.FriendlyName;
SC06 }

```

Function app -

RequestUserApproval.cs:

```

RA01 public static class RequestUserApproval
RA02 {
RA03 [FunctionName("RequestUserApproval")]
RA04 public static async Task<IActionResult> Run(
RA05 [HttpTrigger(AuthorizationLevel.Function, "get", "post", Route = null)] HttpRequest req,
RA06 ILogger log)
RA06 {
RA07     log.LogInformation("RequestUserApproval function processed a request.");
RA08 ...
RA09     return ProcessRequest(req)
RA10     ? (ActionResult)new OkObjectResult($"User approval processed")
RA11     : new BadRequestObjectResult("Failed to process user approval");
RA12 }
RA13 private static bool ProcessRequest(HttpContext req)
RA14 {
RA15     ...
RA16 }
RA17 }

```

### Question

HOTSPOT -

You need to retrieve the database connection string.

Which values should you use? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

### Answer Area

REST API Endpoint:

https://	<input type="text"/>	▼	.vault.azure.net/secrets/	<input type="text"/>	▼ /
	<input type="checkbox"/> cpandlkeyvault <input type="checkbox"/> PostgreSQLConn <input type="checkbox"/> 80df3e46ffcd4f1cb187f79905e9a1e8			<input type="checkbox"/> cpandlkeyvault <input type="checkbox"/> PostgreSQLConn <input type="checkbox"/> 80df3e46ffcd4f1cb187f79905e9a1e8	

Variable type to access Azure Key Vault secret values:

<input type="checkbox"/> Environment
<input type="checkbox"/> Session
<input type="checkbox"/> ViewState
<input type="checkbox"/> Querystring

Correct Answer:

### Answer Area

REST API Endpoint:

https://	<input type="text"/>	▼	.vault.azure.net/secrets/	<input type="text"/>	▼ /
	<input checked="" type="checkbox"/> cpandlkeyvault <input type="checkbox"/> PostgreSQLConn <input type="checkbox"/> 80df3e46ffcd4f1cb187f79905e9a1e8			<input checked="" type="checkbox"/> cpandlkeyvault <input checked="" type="checkbox"/> PostgreSQLConn <input checked="" type="checkbox"/> 80df3e46ffcd4f1cb187f79905e9a1e8	

Variable type to access Azure Key Vault secret values:

<input checked="" type="checkbox"/> Environment
<input type="checkbox"/> Session
<input type="checkbox"/> ViewState
<input checked="" type="checkbox"/> Querystring

Azure database connection string retrieve REST API vault.azure.net/secrets/

Box 1: cpndlkeyvault -

We specify the key vault, cpndlkeyvault.

Scenario: The database connection string is stored in Azure Key Vault with the following attributes:

Azure Key Vault name: cpndlkeyvault

Secret name: PostgreSQLConn -

Id: 80df3e46ffcd4f1cb187f79905e9a1e8

Box 2: PostgreSQLConn -

We specify the secret, PostgreSQLConn

Example, sample request:

<https://myvault.vault.azure.net/secrets/mysecretname/4387e9f3d6e14c459867679a90fd0f79?api-version=7.1>

Box 3: Querystring -

Reference:

<https://docs.microsoft.com/en-us/rest/api/keyvault/getsecret/getsecret>

Question #2

**Introductory Info**

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

City Power & Light company provides electrical infrastructure monitoring solutions for homes and businesses. The company is migrating solutions to Azure.

Current environment -

Architecture overview -

The company has a public website located at <http://www.cpndl.com/>. The site is a single-page web application that runs in Azure App Service on Linux. The website uses files stored in Azure Storage and cached in Azure Content Delivery Network (CDN) to serve static content.

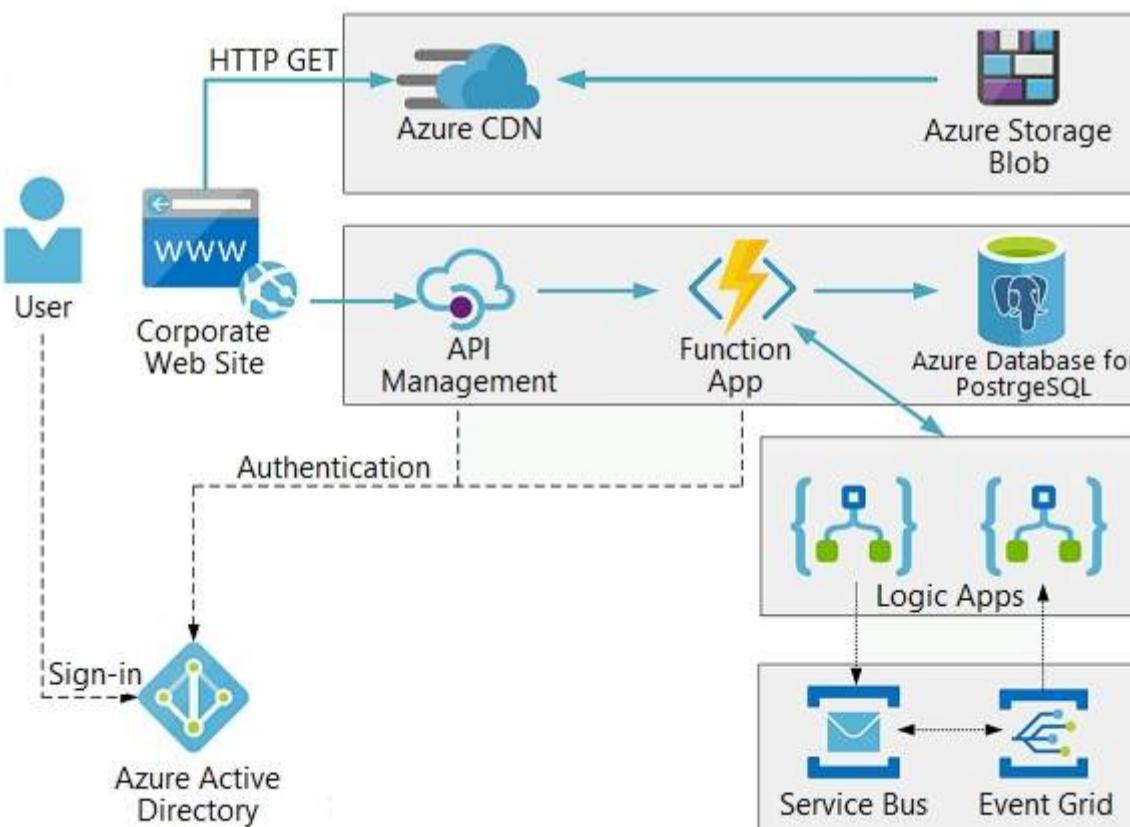
API Management and Azure Function App functions are used to process and store data in Azure Database for PostgreSQL. API Management is used to broker communications to the Azure Function app functions for Logic app integration. Logic apps are used to orchestrate the data processing while Service Bus and

Event Grid handle messaging and events.

The solution uses Application Insights, Azure Monitor, and Azure Key Vault.

Architecture diagram -

The company has several applications and services that support their business. The company plans to implement serverless computing where possible. The overall architecture is shown below.



User authentication -

The following steps detail the user authentication process:

1. The user selects Sign in in the website.
2. The browser redirects the user to the Azure Active Directory (Azure AD) sign in page.

3. The user signs in.
4. Azure AD redirects the user's session back to the web application. The URL includes an access token.
5. The web application calls an API and includes the access token in the authentication header. The application ID is sent as the audience ('`aud`') claim in the access token.
6. The back-end API validates the access token.

Requirements -

Corporate website -

Communications and content must be secured by using SSL.

Communications must use HTTPS.

Data must be replicated to a secondary region and three availability zones.

Data storage costs must be minimized.

Azure Database for PostgreSQL -

The database connection string is stored in Azure Key Vault with the following attributes:

Azure Key Vault name: cpandlkeyvault

Secret name: PostgreSQLConn

Id: 80df3e46ffcd4f1cb187f79905e9a1e8

The connection information is updated frequently. The application must always use the latest information to connect to the database.

Azure Service Bus and Azure Event Grid

Azure Event Grid must use Azure Service Bus for queue-based load leveling.

Events in Azure Event Grid must be routed directly to Service Bus queues for use in buffering.

Events from Azure Service Bus and other Azure services must continue to be routed to Azure Event Grid for processing.

Security -

All SSL certificates and credentials must be stored in Azure Key Vault.

File access must restrict access by IP, protocol, and Azure AD rights.

All user accounts and processes must receive only those privileges which are essential to perform their intended function.

Compliance -

Auditing of the file updates and transfers must be enabled to comply with General Data Protection Regulation (GDPR). The file updates must be read-only, stored in the order in which they occurred, include only create, update, delete, and copy operations, and be retained for compliance reasons.

Issues -

Corporate website -

While testing the site, the following error message displays:

CryptographicException: The system cannot find the file specified.

Function app -

You perform local testing for the RequestUserApproval function. The following error message displays:

'Timeout value of 00:10:00 exceeded by function: RequestUserApproval'

The same error message displays when you test the function in an Azure development environment when you run the following Kusto query:

FunctionAppLogs -

```
| where FunctionName == "RequestUserApproval"
```

Logic app -

You test the Logic app in a development environment. The following error message displays:

'400 Bad Request'

Troubleshooting of the error shows an HttpTrigger action to call the RequestUserApproval function.

Code -

Corporate website -

Security.cs:

```
SC01 public class Security
SC02 {
SC03     var bytes = System.IO.File.ReadAllBytes("~/var/ssl/private");
SC04     var cert = new System.Security.Cryptography.X509Certificate2(bytes);
SC05     var certName = cert.FriendlyName;
SC06 }
```

Function app -

RequestUserApproval.cs:

```
RA01 public static class RequestUserApproval
RA02 {
RA03     [FunctionName("RequestUserApproval")]
RA04     public static async Task<IActionResult> Run(
RA05         [HttpTrigger(AuthorizationLevel.Function, "get", "post", Route = null)] HttpRequest req,
RA06         ILogger log)
RA07     {
RA08         log.LogInformation("RequestUserApproval function processed a request.");
RA09         ...
RA10         return ProcessRequest(req)
RA11             ? (ActionResult)new OkObjectResult($"User approval processed")
RA12             : new BadRequestObjectResult("Failed to process user approval");
RA13     }
RA14     private static bool ProcessRequest(HttpContext req)
RA15     {
RA16         ...
RA17     }
```

### Question

DRAG DROP -

You need to correct the corporate website error.

Which four actions should you recommend be performed 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
Upload the certificate to Azure Key Vault.	
Update line SC05 of Security.cs to include error handling and then redeploy the code.	
Update line SC03 of Security.cs to include a using statement and then re-deploy the code.	
Add the certificate thumbprint to the WEBSITE_LOAD_CERTIFICATES app setting.	 
Upload the certificate to source control.	
Import the certificate to Azure App Service.	
Generate a certificate.	

Actions	Answer Area
Upload the certificate to Azure Key Vault.	Generate a certificate.
Update line SC05 of Security.cs to include error handling and then redeploy the code.	Upload the certificate to Azure Key Vault.
Update line SC03 of Security.cs to include a using statement and then re-deploy the code.	Import the certificate to Azure App Service.
Add the certificate thumbprint to the WEBSITE_LOAD_CERTIFICATES app setting.	Update line SC05 of Security.cs to include error handling and then redeploy the code.
Upload the certificate to source control.	
Import the certificate to Azure App Service.	
Generate a certificate.	

**Correct Answer:** Add the certificate thumbprint to the WEBSITE\_LOAD\_CERTIFICATES app setting.

Scenario: Corporate website -  
While testing the site, the following error message displays:  
CryptographicException: The system cannot find the file specified.

Step 1: Generate a certificate -  
Step 2: Upload the certificate to Azure Key Vault  
Scenario: All SSL certificates and credentials must be stored in Azure Key Vault.  
Step 3: Import the certificate to Azure App Service  
Step 4: Update line SC05 of Security.cs to include error handling and then redeploy the code  
Reference:  
<https://docs.microsoft.com/en-us/azure/app-service/configure-ssl-certificate>

Question #3

**Introductory Info**

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

City Power & Light company provides electrical infrastructure monitoring solutions for homes and businesses. The company is migrating solutions to Azure.

Current environment -

Architecture overview -

The company has a public website located at <http://www.cpndl.com/>. The site is a single-page web application that runs in Azure App Service on Linux. The website uses files stored in Azure Storage and cached in Azure Content Delivery Network (CDN) to serve static content.

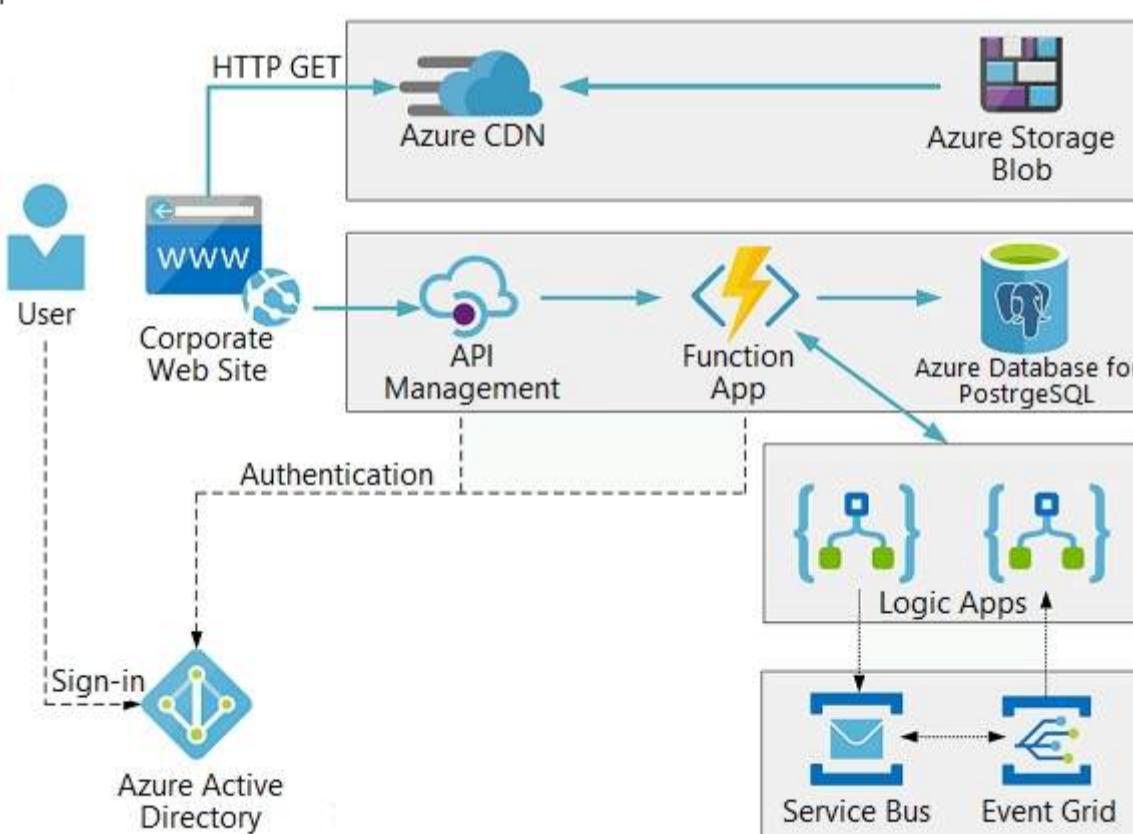
API Management and Azure Function App functions are used to process and store data in Azure Database for PostgreSQL. API Management is used to broker communications to the Azure Function app functions for Logic app integration. Logic apps are used to orchestrate the data processing while Service Bus and

Event Grid handle messaging and events.

The solution uses Application Insights, Azure Monitor, and Azure Key Vault.

Architecture diagram -

The company has several applications and services that support their business. The company plans to implement serverless computing where possible. The overall architecture is shown below.



User authentication -

The following steps detail the user authentication process:

1. The user selects Sign in in the website.
2. The browser redirects the user to the Azure Active Directory (Azure AD) sign in page.

3. The user signs in.
4. Azure AD redirects the user's session back to the web application. The URL includes an access token.
5. The web application calls an API and includes the access token in the authentication header. The application ID is sent as the audience ('`aud`') claim in the access token.
6. The back-end API validates the access token.

Requirements -

Corporate website -

Communications and content must be secured by using SSL.

Communications must use HTTPS.

Data must be replicated to a secondary region and three availability zones.

Data storage costs must be minimized.

Azure Database for PostgreSQL -

The database connection string is stored in Azure Key Vault with the following attributes:

Azure Key Vault name: cpandlkeyvault

Secret name: PostgreSQLConn

Id: 80df3e46ffcd4f1cb187f79905e9a1e8

The connection information is updated frequently. The application must always use the latest information to connect to the database.

Azure Service Bus and Azure Event Grid

Azure Event Grid must use Azure Service Bus for queue-based load leveling.

Events in Azure Event Grid must be routed directly to Service Bus queues for use in buffering.

Events from Azure Service Bus and other Azure services must continue to be routed to Azure Event Grid for processing.

Security -

All SSL certificates and credentials must be stored in Azure Key Vault.

File access must restrict access by IP, protocol, and Azure AD rights.

All user accounts and processes must receive only those privileges which are essential to perform their intended function.

Compliance -

Auditing of the file updates and transfers must be enabled to comply with General Data Protection Regulation (GDPR). The file updates must be read-only, stored in the order in which they occurred, include only create, update, delete, and copy operations, and be retained for compliance reasons.

Issues -

Corporate website -

While testing the site, the following error message displays:

CryptographicException: The system cannot find the file specified.

Function app -

You perform local testing for the RequestUserApproval function. The following error message displays:

'Timeout value of 00:10:00 exceeded by function: RequestUserApproval'

The same error message displays when you test the function in an Azure development environment when you run the following Kusto query:

FunctionAppLogs -

```
| where FunctionName == "RequestUserApproval"
```

Logic app -

You test the Logic app in a development environment. The following error message displays:

'400 Bad Request'

Troubleshooting of the error shows an HttpTrigger action to call the RequestUserApproval function.

Code -

Corporate website -

Security.cs:

```

SC01 public class Security
SC02 {
SC03 var bytes = System.IO.File.ReadAllBytes("~/var/ssl/private");
SC04 var cert = new System.Security.Cryptography.X509Certificate2(bytes);
SC05 var certName = cert.FriendlyName;
SC06 }

```

Function app -

RequestUserApproval.cs:

```

RA01 public static class RequestUserApproval
RA02 {
RA03 [FunctionName("RequestUserApproval")]
RA04 public static async Task<IActionResult> Run(
RA05 [HttpTrigger(AuthorizationLevel.Function, "get", "post", Route = null)] HttpRequest req,
ILogger log)
RA06 {
RA07     log.LogInformation("RequestUserApproval function processed a request.");
RA08 ...
RA09     return ProcessRequest(req)
RA10     ? (ActionResult)new OkObjectResult($"User approval processed")
RA11     : new BadRequestObjectResult("Failed to process user approval");
RA12 }
RA13 private static bool ProcessRequest(HttpContext req)
RA14 {
RA15     ...
RA16 }
RA17 }

```

### Question

HOTSPOT -

You need to configure API Management for authentication.

Which policy values should you use? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

### Answer Area

Setting	Value
Policy	<div style="border: 1px solid #ccc; padding: 5px; width: 150px;"> Check HTTP header  Restrict caller IPs  Limit call rate by key  Validate JWT </div>
Policy section	<div style="border: 1px solid #ccc; padding: 5px; width: 150px;"> Inbound  Outbound </div>

### Answer Area

Setting	Value
Policy	<div style="border: 1px solid #ccc; padding: 5px; width: 150px;"> Check HTTP header  Restrict caller IPs  Limit call rate by key  <span style="background-color: #e0f2e0;">Validate JWT</span> </div>
Policy section	<div style="border: 1px solid #ccc; padding: 5px; width: 150px;"> <span style="background-color: #e0f2e0;">Inbound</span>  <span style="background-color: #e0f2e0;">Outbound</span> </div>

Box 1: Validate JWT -

The validate-jwt policy enforces existence and validity of a JWT extracted from either a specified HTTP Header or a specified query parameter.

Scenario: User authentication (see step 5 below)

The following steps detail the user authentication process:

1. The user selects Sign in in the website.
2. The browser redirects the user to the Azure Active Directory (Azure AD) sign in page.
3. The user signs in.
4. Azure AD redirects the user's session back to the web application. The URL includes an access token.
5. The web application calls an API and includes the access token in the authentication header. The application ID is sent as the audience (`aud`) claim in the access token.
6. The back-end API validates the access token.

Incorrect Answers:

- Limit call rate by key - Prevents API usage spikes by limiting call rate, on a per key basis.
- Restrict caller IPs - Filters (allows/denies) calls from specific IP addresses and/or address ranges.
- Check HTTP header - Enforces existence and/or value of a HTTP Header.

Box 2: Outbound -

Reference:

<https://docs.microsoft.com/en-us/azure/api-management/api-management-access-restriction-policies>

Question #4

**Introductory Info**

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

City Power & Light company provides electrical infrastructure monitoring solutions for homes and businesses. The company is migrating solutions to Azure.

Current environment -

Architecture overview -

The company has a public website located at <http://www.cpndl.com/>. The site is a single-page web application that runs in Azure App Service on Linux. The website uses files stored in Azure Storage and cached in Azure Content Delivery Network (CDN) to serve static content.

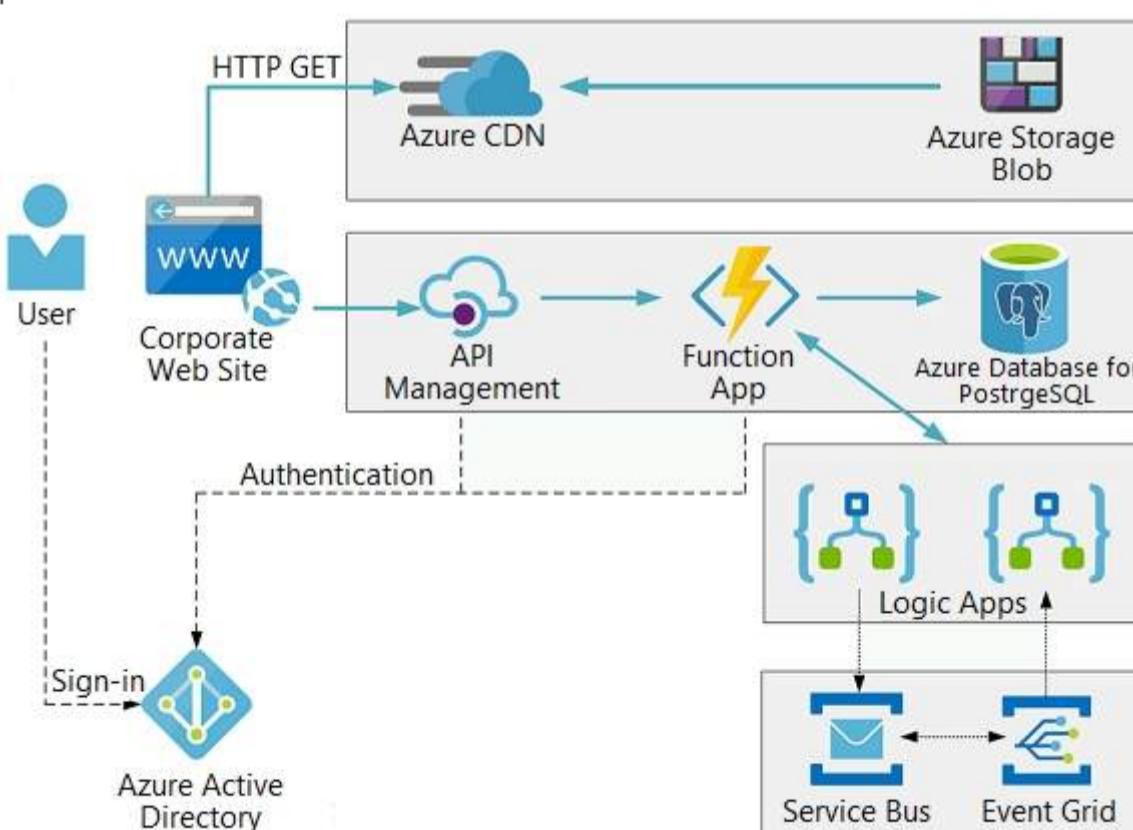
API Management and Azure Function App functions are used to process and store data in Azure Database for PostgreSQL. API Management is used to broker communications to the Azure Function app functions for Logic app integration. Logic apps are used to orchestrate the data processing while Service Bus and

Event Grid handle messaging and events.

The solution uses Application Insights, Azure Monitor, and Azure Key Vault.

Architecture diagram -

The company has several applications and services that support their business. The company plans to implement serverless computing where possible. The overall architecture is shown below.



User authentication -

The following steps detail the user authentication process:

1. The user selects Sign in in the website.
2. The browser redirects the user to the Azure Active Directory (Azure AD) sign in page.

3. The user signs in.
4. Azure AD redirects the user's session back to the web application. The URL includes an access token.
5. The web application calls an API and includes the access token in the authentication header. The application ID is sent as the audience ('`aud`') claim in the access token.
6. The back-end API validates the access token.

Requirements -

Corporate website -

Communications and content must be secured by using SSL.

Communications must use HTTPS.

Data must be replicated to a secondary region and three availability zones.

Data storage costs must be minimized.

Azure Database for PostgreSQL -

The database connection string is stored in Azure Key Vault with the following attributes:

Azure Key Vault name: cpandlkeyvault

Secret name: PostgreSQLConn

Id: 80df3e46ffcd4f1cb187f79905e9a1e8

The connection information is updated frequently. The application must always use the latest information to connect to the database.

Azure Service Bus and Azure Event Grid

Azure Event Grid must use Azure Service Bus for queue-based load leveling.

Events in Azure Event Grid must be routed directly to Service Bus queues for use in buffering.

Events from Azure Service Bus and other Azure services must continue to be routed to Azure Event Grid for processing.

Security -

All SSL certificates and credentials must be stored in Azure Key Vault.

File access must restrict access by IP, protocol, and Azure AD rights.

All user accounts and processes must receive only those privileges which are essential to perform their intended function.

Compliance -

Auditing of the file updates and transfers must be enabled to comply with General Data Protection Regulation (GDPR). The file updates must be read-only, stored in the order in which they occurred, include only create, update, delete, and copy operations, and be retained for compliance reasons.

Issues -

Corporate website -

While testing the site, the following error message displays:

CryptographicException: The system cannot find the file specified.

Function app -

You perform local testing for the RequestUserApproval function. The following error message displays:

'Timeout value of 00:10:00 exceeded by function: RequestUserApproval'

The same error message displays when you test the function in an Azure development environment when you run the following Kusto query:

FunctionAppLogs -

```
| where FunctionName == "RequestUserApproval"
```

Logic app -

You test the Logic app in a development environment. The following error message displays:

'400 Bad Request'

Troubleshooting of the error shows an HttpTrigger action to call the RequestUserApproval function.

Code -

Corporate website -

Security.cs:

```

SC01 public class Security
SC02 {
SC03 var bytes = System.IO.File.ReadAllBytes("~/var/ssl/private");
SC04 var cert = new System.Security.Cryptography.X509Certificate2(bytes);
SC05 var certName = cert.FriendlyName;
SC06 }

```

Function app -

RequestUserApproval.cs:

```

RA01 public static class RequestUserApproval
RA02 {
RA03 [FunctionName("RequestUserApproval")]
RA04 public static async Task<IActionResult> Run(
RA05 [HttpTrigger(AuthorizationLevel.Function, "get", "post", Route = null)] HttpRequest req,
RA06 ILogger log)
RA06 {
RA07     log.LogInformation("RequestUserApproval function processed a request.");
RA08 ...
RA09     return ProcessRequest(req)
RA10     ? (ActionResult)new OkObjectResult($"User approval processed")
RA11     : new BadRequestObjectResult("Failed to process user approval");
RA12 }
RA13 private static bool ProcessRequest(HttpRequest req)
RA14 {
RA15     ...
RA16 }
RA17 }

```

### Question

You need to authenticate the user to the corporate website as indicated by the architectural diagram.

Which two values should you use? Each correct answer presents part of the solution.

NOTE: Each correct selection is worth one point.

- A. ID token signature
- B. ID token claims
- C. HTTP response code
- D. Azure AD endpoint URI
- E. Azure AD tenant ID

#### Correct Answer: AD

A: Claims in access tokens -

JWTs (JSON Web Tokens) are split into three pieces:

- ☞ Header - Provides information about how to validate the token including information about the type of token and how it was signed.
- ☞ Payload - Contains all of the important data about the user or app that is attempting to call your service.
- ☞ Signature - Is the raw material used to validate the token.

E: Your client can get an access token from either the v1.0 endpoint or the v2.0 endpoint using a variety of protocols.

Scenario: User authentication (see step 5 below)

The following steps detail the user authentication process:

1. The user selects Sign in in the website.
2. The browser redirects the user to the Azure Active Directory (Azure AD) sign in page.
3. The user signs in.
4. Azure AD redirects the user's session back to the web application. The URL includes an access token.
5. The web application calls an API and includes the access token in the authentication header. The application ID is sent as the audience (`aud`) claim in the access token.
6. The back-end API validates the access token.

Reference:

<https://docs.microsoft.com/en-us/azure/api-management/api-management-access-restriction-policies>

Question #5

**Introductory Info**

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

City Power & Light company provides electrical infrastructure monitoring solutions for homes and businesses. The company is migrating solutions to Azure.

Current environment -

Architecture overview -

The company has a public website located at <http://www.cpndl.com/>. The site is a single-page web application that runs in Azure App Service on Linux. The website uses files stored in Azure Storage and cached in Azure Content Delivery Network (CDN) to serve static content.

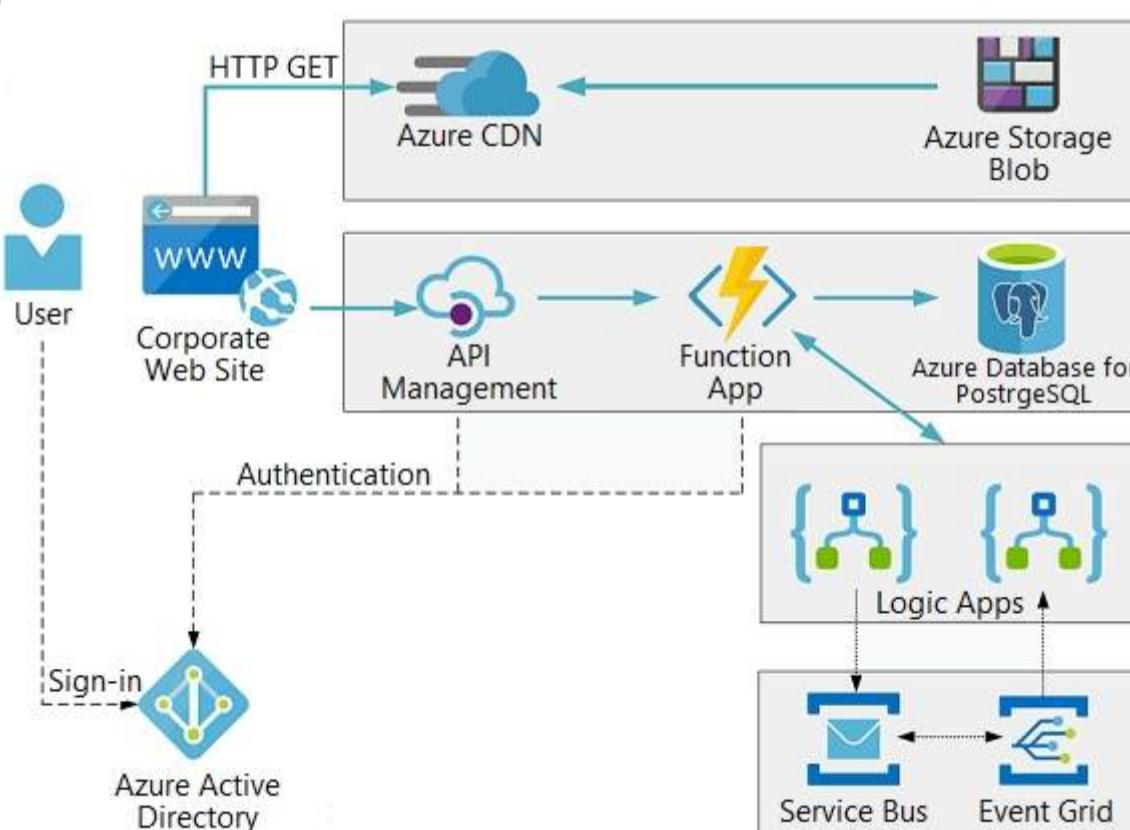
API Management and Azure Function App functions are used to process and store data in Azure Database for PostgreSQL. API Management is used to broker communications to the Azure Function app functions for Logic app integration. Logic apps are used to orchestrate the data processing while Service Bus and

Event Grid handle messaging and events.

The solution uses Application Insights, Azure Monitor, and Azure Key Vault.

Architecture diagram -

The company has several applications and services that support their business. The company plans to implement serverless computing where possible. The overall architecture is shown below.



User authentication -

The following steps detail the user authentication process:

1. The user selects Sign in in the website.
2. The browser redirects the user to the Azure Active Directory (Azure AD) sign in page.

3. The user signs in.
4. Azure AD redirects the user's session back to the web application. The URL includes an access token.
5. The web application calls an API and includes the access token in the authentication header. The application ID is sent as the audience ('`aud`') claim in the access token.
6. The back-end API validates the access token.

Requirements -

Corporate website -

Communications and content must be secured by using SSL.

Communications must use HTTPS.

Data must be replicated to a secondary region and three availability zones.

Data storage costs must be minimized.

Azure Database for PostgreSQL -

The database connection string is stored in Azure Key Vault with the following attributes:

Azure Key Vault name: cpandlkeyvault

Secret name: PostgreSQLConn

Id: 80df3e46ffcd4f1cb187f79905e9a1e8

The connection information is updated frequently. The application must always use the latest information to connect to the database.

Azure Service Bus and Azure Event Grid

Azure Event Grid must use Azure Service Bus for queue-based load leveling.

Events in Azure Event Grid must be routed directly to Service Bus queues for use in buffering.

Events from Azure Service Bus and other Azure services must continue to be routed to Azure Event Grid for processing.

Security -

All SSL certificates and credentials must be stored in Azure Key Vault.

File access must restrict access by IP, protocol, and Azure AD rights.

All user accounts and processes must receive only those privileges which are essential to perform their intended function.

Compliance -

Auditing of the file updates and transfers must be enabled to comply with General Data Protection Regulation (GDPR). The file updates must be read-only, stored in the order in which they occurred, include only create, update, delete, and copy operations, and be retained for compliance reasons.

Issues -

Corporate website -

While testing the site, the following error message displays:

CryptographicException: The system cannot find the file specified.

Function app -

You perform local testing for the RequestUserApproval function. The following error message displays:

'Timeout value of 00:10:00 exceeded by function: RequestUserApproval'

The same error message displays when you test the function in an Azure development environment when you run the following Kusto query:

FunctionAppLogs -

```
| where FunctionName == "RequestUserApproval"
```

Logic app -

You test the Logic app in a development environment. The following error message displays:

'400 Bad Request'

Troubleshooting of the error shows an HttpTrigger action to call the RequestUserApproval function.

Code -

Corporate website -

Security.cs:

```

SC01 public class Security
SC02 {
SC03 var bytes = System.IO.File.ReadAllBytes("~/var/ssl/private");
SC04 var cert = new System.Security.Cryptography.X509Certificate2(bytes);
SC05 var certName = cert.FriendlyName;
SC06 }

```

Function app -

RequestUserApproval.cs:

```

RA01 public static class RequestUserApproval
RA02 {
RA03 [FunctionName("RequestUserApproval")]
RA04 public static async Task<IActionResult> Run(
RA05 [HttpTrigger(AuthorizationLevel.Function, "get", "post", Route = null)] HttpRequest req,
RA06 ILogger log)
RA06 {
RA07     log.LogInformation("RequestUserApproval function processed a request.");
RA08 ...
RA09     return ProcessRequest(req)
RA10     ? (ActionResult)new OkObjectResult($"User approval processed")
RA11     : new BadRequestObjectResult("Failed to process user approval");
RA12 }
RA13 private static bool ProcessRequest(HttpRequest req)
RA14 {
RA15     ...
RA16 }
RA17 }

```

### Question

HOTSPOT -

You need to correct the Azure Logic app error message.

Which configuration values should you use? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

### Answer Area

Setting	Value
authentication level	<div style="border: 1px solid #ccc; padding: 5px; width: fit-content;"> anonymous  function  admin </div>
managed identity	<div style="border: 1px solid #ccc; padding: 5px; width: fit-content;"> system-assigned  user-assigned </div>

### Answer Area

Setting	Value
authentication level	<div style="border: 1px solid #ccc; padding: 5px; width: fit-content;"> anonymous  <span style="background-color: #c8e6c9; border: 1px solid #ccc; padding: 2px;">function</span>  admin </div>
managed identity	<div style="border: 1px solid #ccc; padding: 5px; width: fit-content;"> <span style="background-color: #c8e6c9; border: 1px solid #ccc; padding: 2px;">system-assigned</span>  user-assigned </div>

Scenario: You test the Logic app in a development environment. The following error message displays:

'400 Bad Request'

Troubleshooting of the error shows an HttpTrigger action to call the RequestUserApproval function.

Note: If the inbound call's request body doesn't match your schema, the trigger returns an HTTP 400 Bad Request error.

**Box 1: function -**

If you have an Azure function where you want to use the system-assigned identity, first enable authentication for Azure functions.

**Box 2: system-assigned -**

Your logic app or individual connections can use either the system-assigned identity or a single user-assigned identity, which you can share across a group of logic apps, but not both.

Reference:

<https://docs.microsoft.com/en-us/azure/logic-apps/create-managed-service-identity>

Question #6

**Introductory Info**

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

City Power & Light company provides electrical infrastructure monitoring solutions for homes and businesses. The company is migrating solutions to Azure.

Current environment -

Architecture overview -

The company has a public website located at <http://www.cpndl.com/>. The site is a single-page web application that runs in Azure App Service on Linux. The website uses files stored in Azure Storage and cached in Azure Content Delivery Network (CDN) to serve static content.

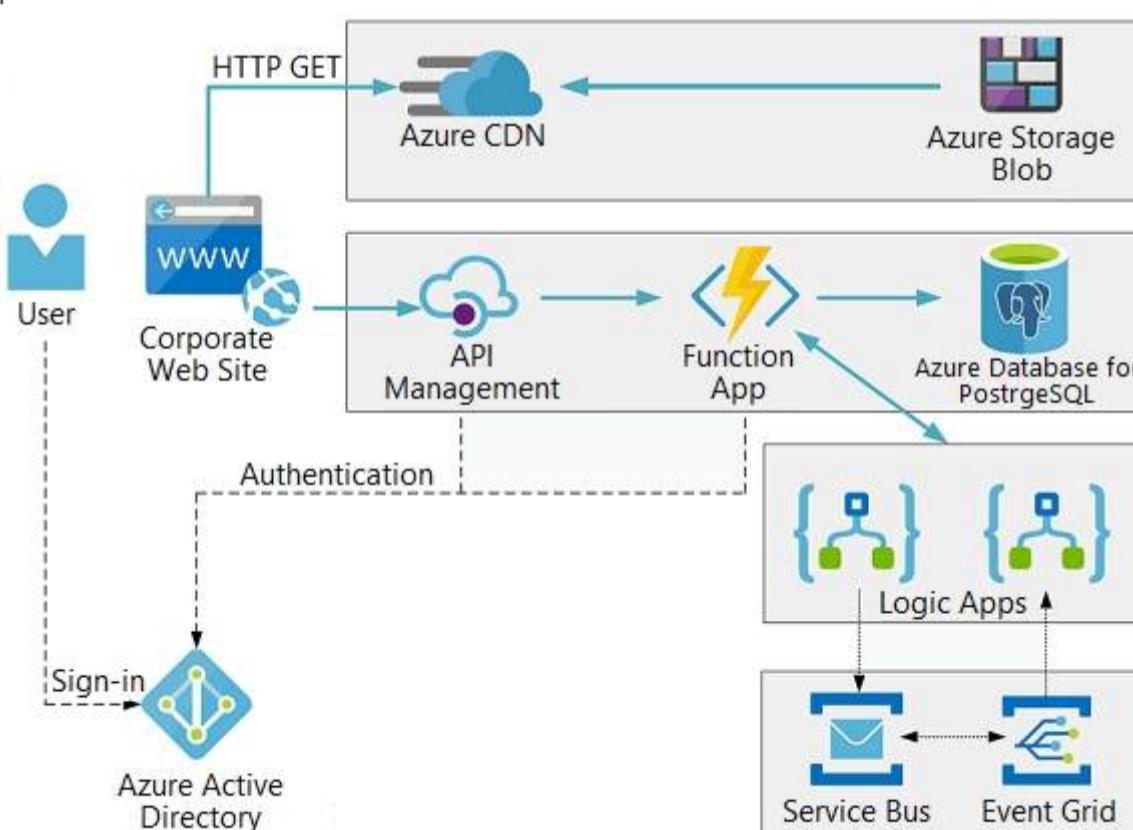
API Management and Azure Function App functions are used to process and store data in Azure Database for PostgreSQL. API Management is used to broker communications to the Azure Function app functions for Logic app integration. Logic apps are used to orchestrate the data processing while Service Bus and

Event Grid handle messaging and events.

The solution uses Application Insights, Azure Monitor, and Azure Key Vault.

Architecture diagram -

The company has several applications and services that support their business. The company plans to implement serverless computing where possible. The overall architecture is shown below.



User authentication -

The following steps detail the user authentication process:

1. The user selects Sign in in the website.
2. The browser redirects the user to the Azure Active Directory (Azure AD) sign in page.

3. The user signs in.
4. Azure AD redirects the user's session back to the web application. The URL includes an access token.
5. The web application calls an API and includes the access token in the authentication header. The application ID is sent as the audience ('`aud`') claim in the access token.
6. The back-end API validates the access token.

Requirements -

Corporate website -

Communications and content must be secured by using SSL.

Communications must use HTTPS.

Data must be replicated to a secondary region and three availability zones.

Data storage costs must be minimized.

Azure Database for PostgreSQL -

The database connection string is stored in Azure Key Vault with the following attributes:

Azure Key Vault name: cpandlkeyvault

Secret name: PostgreSQLConn

Id: 80df3e46ffcd4f1cb187f79905e9a1e8

The connection information is updated frequently. The application must always use the latest information to connect to the database.

Azure Service Bus and Azure Event Grid

Azure Event Grid must use Azure Service Bus for queue-based load leveling.

Events in Azure Event Grid must be routed directly to Service Bus queues for use in buffering.

Events from Azure Service Bus and other Azure services must continue to be routed to Azure Event Grid for processing.

Security -

All SSL certificates and credentials must be stored in Azure Key Vault.

File access must restrict access by IP, protocol, and Azure AD rights.

All user accounts and processes must receive only those privileges which are essential to perform their intended function.

Compliance -

Auditing of the file updates and transfers must be enabled to comply with General Data Protection Regulation (GDPR). The file updates must be read-only, stored in the order in which they occurred, include only create, update, delete, and copy operations, and be retained for compliance reasons.

Issues -

Corporate website -

While testing the site, the following error message displays:

CryptographicException: The system cannot find the file specified.

Function app -

You perform local testing for the RequestUserApproval function. The following error message displays:

'Timeout value of 00:10:00 exceeded by function: RequestUserApproval'

The same error message displays when you test the function in an Azure development environment when you run the following Kusto query:

FunctionAppLogs -

```
| where FunctionName == "RequestUserApproval"
```

Logic app -

You test the Logic app in a development environment. The following error message displays:

'400 Bad Request'

Troubleshooting of the error shows an HttpTrigger action to call the RequestUserApproval function.

Code -

Corporate website -

Security.cs:

```

SC01 public class Security
SC02 {
SC03 var bytes = System.IO.File.ReadAllBytes("~/var/ssl/private");
SC04 var cert = new System.Security.Cryptography.X509Certificate2(bytes);
SC05 var certName = cert.FriendlyName;
SC06 }

```

Function app -

RequestUserApproval.cs:

```

RA01 public static class RequestUserApproval
RA02 {
RA03 [FunctionName("RequestUserApproval")]
RA04 public static async Task<IActionResult> Run(
RA05 [HttpTrigger(AuthorizationLevel.Function, "get", "post", Route = null)] HttpRequest req,
RA06 ILogger log)
RA06 {
RA07     log.LogInformation("RequestUserApproval function processed a request.");
RA08 ...
RA09     return ProcessRequest(req)
RA10     ? (ActionResult)new OkObjectResult($"User approval processed")
RA11     : new BadRequestObjectResult("Failed to process user approval");
RA12 }
RA13 private static bool ProcessRequest(HttpRequest req)
RA14 {
RA15     ...
RA16 }
RA17 }

```

### Question

HOTSPOT -

You need to configure Azure Service Bus to Event Grid integration.

Which Azure Service Bus settings should you use? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

### Answer Area

Setting	Value
Tier	<input type="checkbox"/> Basic <input type="checkbox"/> Standard <input type="checkbox"/> Premium
RBAC role	<input type="checkbox"/> Owner <input type="checkbox"/> Contributor <input type="checkbox"/> Azure Service Bus Data Owner <input type="checkbox"/> Azure Service Bus Data Receiver

### Answer Area

Setting	Value
Tier	<input type="checkbox"/> Basic <input type="checkbox"/> Standard <input checked="" type="checkbox"/> Premium
RBAC role	<input type="checkbox"/> Owner <input checked="" type="checkbox"/> Contributor <input type="checkbox"/> Azure Service Bus Data Owner <input type="checkbox"/> Azure Service Bus Data Receiver

Box 1: Premium -

Service Bus can now emit events to Event Grid when there are messages in a queue or a subscription when no receivers are present. You can create Event Grid subscriptions to your Service Bus namespaces, listen to these events, and then react to the events by starting a receiver. With

this feature, you can use Service Bus in reactive programming models.

To enable the feature, you need the following items:

A Service Bus Premium namespace with at least one Service Bus queue or a Service Bus topic with at least one subscription.

Contributor access to the Service Bus namespace.

Box 2: Contributor -

Reference:

<https://docs.microsoft.com/en-us/azure/service-bus-messaging/service-bus-to-event-grid-integration-concept>

### Topic 11 - Testlet 13

Question #1

Topic 11

## Introductory Info

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 questions in 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 section 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 in 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. 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 SaaS company that provides a solution for managing employee expenses. The solution consists of an ASP.NET Core Web API project that is deployed as an Azure Web App.

Overall architecture -

Employees upload receipts for the system to process. When processing is complete, the employee receives a summary report email that details the processing results. Employees then use a web application to manage their receipts and perform any additional tasks needed for reimbursement.

Receipt processing -

Employees may upload receipts in two ways:

Uploading using an Azure Files mounted folder

Uploading using the web application

Data Storage -

Receipt and employee information is stored in an Azure SQL database.

Documentation -

Employees are provided with a getting started document when they first use the solution. The documentation includes details on supported operating systems for

Azure File upload, and instructions on how to configure the mounted folder.

Solution details -

Users table -

Column	Description
UserId	unique identifier for an employee
ExpenseAccount	employees expense account number in the format 1234-123-1234
AllowedAmount	limit of allowed expenses before approval is needed
SupervisorId	unique identifier for employee's supervisor
SecurityPin	value used to validate user identity

Web Application -

You enable MSI for the Web App and configure the Web App to use the security principal name WebAppIdentity.

Processing -

Processing is performed by an Azure Function that uses version 2 of the Azure Function runtime. Once processing is completed, results are stored in Azure Blob

Storage and an Azure SQL database. Then, an email summary is sent to the user with a link to the processing report. The link to the report must

remain valid if the email is forwarded to another user.

#### Logging -

Azure Application Insights is used for telemetry and logging in both the processor and the web application. The processor also has TraceWriter logging enabled.

Application Insights must always contain all log messages.

#### Requirements -

##### Receipt processing -

Concurrent processing of a receipt must be prevented.

##### Disaster recovery -

Regional outage must not impact application availability. All DR operations must not be dependent on application running and must ensure that data in the DR region is up to date.

#### Security -

User's SecurityPin must be stored in such a way that access to the database does not allow the viewing of SecurityPins. The web application is the only system that should have access to SecurityPins.

All certificates and secrets used to secure data must be stored in Azure Key Vault.

You must adhere to the principle of least privilege and provide privileges which are essential to perform the intended function.

All access to Azure Storage and Azure SQL database must use the application's Managed Service Identity (MSI).

Receipt data must always be encrypted at rest.

All data must be protected in transit.

User's expense account number must be visible only to logged in users. All other views of the expense account number should include only the last segment, with the remaining parts obscured.

In the case of a security breach, access to all summary reports must be revoked without impacting other parts of the system.

#### Issues -

##### Upload format issue -

Employees occasionally report an issue with uploading a receipt using the web application. They report that when they upload a receipt using the Azure File Share, the receipt does not appear in their profile. When this occurs, they delete the file in the file share and use the web application, which returns a 500 Internal Server error page.

##### Capacity issue -

During busy periods, employees report long delays between the time they upload the receipt and when it appears in the web application.

##### Log capacity issue -

Developers report that the number of log messages in the trace output for the processor is too high, resulting in lost log messages.

#### Application code -

##### Processing.cs -

```

PC01 public static class Processing
PC02 {
PC03     public static class Function
PC04     {
PC05         [FunctionName("IssueWork")]
PC06         public static async Task Run([TimerTrigger("0 */5 * * *")] TimerInfo timer, ILogger
log)
PC07         {
PC08             var container = await GetCloudBlobContainer();
PC09             foreach (var fileItem in await ListFiles())
PC10             {
PC11                 var file = new CloudFile(fileItem.StorageUri.PrimaryUri);
PC12                 var ms = new MemoryStream();
PC13                 await file.DownloadToStreamAsync(ms);
PC14                 var blob = container.GetBlockBlobReference(fileItem.Uri.ToString());
PC15                 await blob.UploadFromStreamAsync(ms);
PC16             }
PC17         }
PC18     }
PC19     private static CloudBlockBlob GetDRBlob(CloudBlockBlob sourceBlob)
PC20     {
PC21         . .
PC22     }
PC23     private static async Task<CloudBlobContainer> GetCloudBlobContainer()
PC24     {
PC25         var cloudBlobClient = new CloudBlobClient(new Uri(" . . ."), await GetCredentials());
PC26
PC27         await cloudBlobClient.GetRootContainerReference().CreateIfNotExistsAsync();
PC28         return cloudBlobClient.GetRootContainerReference();
PC29     }
PC30     private static async Task<StorageCredentials> GetCredentials()
PC31     {
PC32         . .
PC33     }
PC34     private static async Task<List<IListFileItem>> ListFiles()
PC35     {
PC36         . .
PC37     }
PC38     private KeyVaultClient _keyVaultClient = new KeyVaultClient(" . . .");
PC39 }

```

Database.cs -

```

DB01 public class Database
DB02 {
DB03     private string ConnectionString =
DB04
DB05     public async Task<object> LoadUserDetails(string userId)
DB06     {
DB07
DB08         return await policy.ExecuteAsync(async () =>
DB09         {
DB10             using (var connection = new SqlConnection(ConnectionString))
DB11             {
DB12                 await connection.OpenAsync();
DB13                 using (var command = new SqlCommand(" . . .", connection))
DB14                     using (var reader = command.ExecuteReader())
DB15                     {
DB16                         . .
DB17                     }
DB18                 }
DB19             });
DB20         }
DB21     }

```

ReceiptUploader.cs -

```

RU01 public class ReceiptUploader
RU02 {
RU03     public async Task UploadFile(string file, byte[] binary)
RU04     {
RU05         var httpClient = new HttpClient();
RU06         var response = await httpClient.PutAsync("...", new ByteArrayContent(binary));
RU07         while (ShouldRetry(response))
RU08         {
RU09             response = await httpClient.PutAsync("...", new ByteArrayContent(binary));
RU10         }
RU11     }
RU12     private bool ShouldRetry(HttpStatusCode response)
RU13     {
RU14     }
RU15 }
RU16 }
```

ConfigureSSE.ps1 -

```

CS01 $storageAccount = Get-AzureRmStorageAccount -ResourceGroupName "..." -AccountName "..."
CS02 $keyVault = Get-AzureRmKeyVault -VaultName "..."
CS03 $key = Get-AzureKeyVaultKey -VaultName $keyVault.VaultName -Name "..."
CS04 Set-AzureRmKeyVaultAccessPolicy ` 
CS05 -VaultName $keyVault.VaultName ` 
CS06 -ObjectId $storageAccount.Identity.PrincipalId ` 
CS07 
CS08 
CS09 Set-AzureRmStorageAccount ` 
CS10 -ResourceGroupName $storageAccount.ResourceGroupName ` 
CS11 -AccountName $storageAccount.StorageAccountName ` 
CS12 -EnableEncryptionService File ` 
CS13 -KeyvaultEncryption ` 
CS14 -KeyName $key.Name 
CS15 -KeyVersion $key.Version ` 
CS16 -KeyVaultUri $keyVault.VaultUri
```

## Question

HOTSPOT -

You need to add code at line PC26 of Processing.cs to ensure that security policies are met.

How should you complete the code that you will add at line PC26? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

## Answer Area

```
var resolver = new KeyVaultKeyResolver(_keyVaultClient);
var keyBundle = await _keyVaultClient.GetKeyAsync("...", "...");
```

```
var key = keyBundle.Key;
var key = keyBundle.KeyIdentifier.Identifier;
var key = await resolver.ResolveKeyAsync("encrypt", null);
var key = await resolver.ResolveKeyAsync(keyBundle.KeyIdentifier.Identifier, CancellationToken.None);
```

```
var x = keyBundle.Managed;
var x = AuthenticationScheme.SharedKey;
var x = new BlobEncryptionPolicy(key, resolver);
var x = new DeleteRetentionPolicy {Enabled = key.Kid != null};
```

```
cloudBlobClient.AuthenticationScheme = x;
cloudBlobClient.DefaultRequestOptions.RequireEncryption = x;
cloudBlobClient.DefaultRequestOptions.EncryptionPolicy = x;
cloudBlobClient.SetServiceProperties(new ServiceProperties(deleteRetentionPolicy:x));
```

Correct Answer:

## Answer Area

```
var resolver = new KeyVaultKeyResolver(_keyVaultClient);
var keyBundle = await _keyVaultClient.GetKeyAsync("...", "...");
```

```
var key = keyBundle.Key;
var key = keyBundle.KeyIdentifier.Identifier;
var key = await resolver.ResolveKeyAsync("encrypt", null);
var key = await resolver.ResolveKeyAsync(keyBundle.KeyIdentifier.Identifier, CancellationToken.None);
```

```
var x = keyBundle.Managed;
var x = AuthenticationScheme.SharedKey;
var x = new BlobEncryptionPolicy(key, resolver);
var x = new DeleteRetentionPolicy {Enabled = key.Kid != null};
```

```
cloudBlobClient.AuthenticationScheme = x;
cloudBlobClient.DefaultRequestOptions.RequireEncryption = x;
cloudBlobClient.DefaultRequestOptions.EncryptionPolicy = x;
cloudBlobClient.SetServiceProperties(new ServiceProperties(deleteRetentionPolicy:x));
```

Box 1: var key = await Resolver.ResolveKeyAsyn(keyBundle,KeyIdentifier.CancellationToken.None);

Box 2: var x = new BlobEncryptionPolicy(key,resolver);

Example:

```
// We begin with cloudKey1, and a resolver capable of resolving and caching Key Vault secrets.
```

```
BlobEncryptionPolicy encryptionPolicy = new BlobEncryptionPolicy(cloudKey1, cachingResolver);
```

```
client.DefaultRequestOptions.EncryptionPolicy = encryptionPolicy;
```

Box 3: cloudblobClient. DefaultRequestOptions.EncryptionPolicy = x;

Reference:

<https://github.com/Azure/azure-storage-net/blob/master/Samples/GettingStarted/EncryptionSamples/KeyRotation/Program.cs>

Question #2

## Introductory Info

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 questions in 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 section 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 in 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. 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 SaaS company that provides a solution for managing employee expenses. The solution consists of an ASP.NET Core Web API project that is deployed as an Azure Web App.

Overall architecture -

Employees upload receipts for the system to process. When processing is complete, the employee receives a summary report email that details the processing results. Employees then use a web application to manage their receipts and perform any additional tasks needed for reimbursement.

Receipt processing -

Employees may upload receipts in two ways:

Uploading using an Azure Files mounted folder

Uploading using the web application

Data Storage -

Receipt and employee information is stored in an Azure SQL database.

Documentation -

Employees are provided with a getting started document when they first use the solution. The documentation includes details on supported operating systems for

Azure File upload, and instructions on how to configure the mounted folder.

Solution details -

Users table -

Column	Description
UserId	unique identifier for an employee
ExpenseAccount	employees expense account number in the format 1234-123-1234
AllowedAmount	limit of allowed expenses before approval is needed
SupervisorId	unique identifier for employee's supervisor
SecurityPin	value used to validate user identity

Web Application -

You enable MSI for the Web App and configure the Web App to use the security principal name WebAppIdentity.

Processing -

Processing is performed by an Azure Function that uses version 2 of the Azure Function runtime. Once processing is completed, results are stored in Azure Blob

Storage and an Azure SQL database. Then, an email summary is sent to the user with a link to the processing report. The link to the report must

remain valid if the email is forwarded to another user.

#### Logging -

Azure Application Insights is used for telemetry and logging in both the processor and the web application. The processor also has TraceWriter logging enabled.

Application Insights must always contain all log messages.

#### Requirements -

##### Receipt processing -

Concurrent processing of a receipt must be prevented.

##### Disaster recovery -

Regional outage must not impact application availability. All DR operations must not be dependent on application running and must ensure that data in the DR region is up to date.

#### Security -

User's SecurityPin must be stored in such a way that access to the database does not allow the viewing of SecurityPins. The web application is the only system that should have access to SecurityPins.

All certificates and secrets used to secure data must be stored in Azure Key Vault.

You must adhere to the principle of least privilege and provide privileges which are essential to perform the intended function.

All access to Azure Storage and Azure SQL database must use the application's Managed Service Identity (MSI).

Receipt data must always be encrypted at rest.

All data must be protected in transit.

User's expense account number must be visible only to logged in users. All other views of the expense account number should include only the last segment, with the remaining parts obscured.

In the case of a security breach, access to all summary reports must be revoked without impacting other parts of the system.

#### Issues -

##### Upload format issue -

Employees occasionally report an issue with uploading a receipt using the web application. They report that when they upload a receipt using the Azure File Share, the receipt does not appear in their profile. When this occurs, they delete the file in the file share and use the web application, which returns a 500 Internal Server error page.

##### Capacity issue -

During busy periods, employees report long delays between the time they upload the receipt and when it appears in the web application.

##### Log capacity issue -

Developers report that the number of log messages in the trace output for the processor is too high, resulting in lost log messages.

#### Application code -

##### Processing.cs -

```

PC01 public static class Processing
PC02 {
PC03     public static class Function
PC04     {
PC05         [FunctionName("IssueWork")]
PC06         public static async Task Run([TimerTrigger("0 */5 * * *")] TimerInfo timer, ILogger
log)
PC07         {
PC08             var container = await GetCloudBlobContainer();
PC09             foreach (var fileItem in await ListFiles())
PC10             {
PC11                 var file = new CloudFile(fileItem.StorageUri.PrimaryUri);
PC12                 var ms = new MemoryStream();
PC13                 await file.DownloadToStreamAsync(ms);
PC14                 var blob = container.GetBlockBlobReference(fileItem.Uri.ToString());
PC15                 await blob.UploadFromStreamAsync(ms);
PC16             }
PC17         }
PC18     }
PC19     private static CloudBlockBlob GetDRBlob(CloudBlockBlob sourceBlob)
PC20     {
PC21         . .
PC22     }
PC23     private static async Task<CloudBlobContainer> GetCloudBlobContainer()
PC24     {
PC25         var cloudBlobClient = new CloudBlobClient(new Uri(" . . ."), await GetCredentials());
PC26
PC27         await cloudBlobClient.GetRootContainerReference().CreateIfNotExistsAsync();
PC28         return cloudBlobClient.GetRootContainerReference();
PC29     }
PC30     private static async Task<StorageCredentials> GetCredentials()
PC31     {
PC32         . .
PC33     }
PC34     private static async Task<List<IListFileItem>> ListFiles()
PC35     {
PC36         . .
PC37     }
PC38     private KeyVaultClient _keyVaultClient = new KeyVaultClient(" . . .");
PC39 }

```

Database.cs -

```

DB01 public class Database
DB02 {
DB03     private string ConnectionString =
DB04
DB05     public async Task<object> LoadUserDetails(string userId)
DB06     {
DB07
DB08         return await policy.ExecuteAsync(async () =>
DB09         {
DB10             using (var connection = new SqlConnection(ConnectionString))
DB11             {
DB12                 await connection.OpenAsync();
DB13                 using (var command = new SqlCommand(" . . .", connection))
DB14                     using (var reader = command.ExecuteReader())
DB15                     {
DB16                         . .
DB17                     }
DB18                 }
DB19             });
DB20         }
DB21     }

```

ReceiptUploader.cs -

```

RU01 public class ReceiptUploader
RU02 {
RU03     public async Task UploadFile(string file, byte[] binary)
RU04     {
RU05         var httpClient = new HttpClient();
RU06         var response = await httpClient.PutAsync("...", new ByteArrayContent(binary));
RU07         while (ShouldRetry(response))
RU08         {
RU09             response = await httpClient.PutAsync("...", new ByteArrayContent(binary));
RU10         }
RU11     }
RU12     private bool ShouldRetry(HttpStatusCode response)
RU13     {
RU14     }
RU15 }
RU16 }
```

ConfigureSSE.ps1 -

```

CS01 $storageAccount = Get-AzureRmStorageAccount -ResourceGroupName "..." -AccountName "..."
CS02 $keyVault = Get-AzureRmKeyVault -VaultName "..."
CS03 $key = Get-AzureKeyVaultKey -VaultName $keyVault.VaultName -Name "..."
CS04 Set-AzureRmKeyVaultAccessPolicy ` 
CS05     -VaultName $keyVault.VaultName ` 
CS06     -ObjectId $storageAccount.Identity.PrincipalId ` 
CS07
CS08
CS09 Set-AzureRmStorageAccount ` 
CS10     -ResourceGroupName $storageAccount.ResourceGroupName ` 
CS11     -AccountName $storageAccount.StorageAccountName ` 
CS12     -EnableEncryptionService File ` 
CS13     -KeyvaultEncryption ` 
CS14     -KeyName $key.Name 
CS15     -KeyVersion $key.Version ` 
CS16     -KeyVaultUri $keyVault.VaultUri
```

### Question

You need to ensure the security policies are met.

What code do you add at line CS07 of ConfigureSSE.ps1?

- A. "PermissionsToKeys create, encrypt, decrypt
- B. "PermissionsToCertificates create, encrypt, decrypt
- C. "PermissionsToCertificates wrapkey, unwrapkey, get

### Topic 12 - Testlet 14

#### Correct Answer: B

Scenario: All certificates and secrets used to secure data must be stored in Azure Key Vault.

You must adhere to the principle of least privilege and provide privileges which are essential to perform the intended function.

The Set-AzureRmKeyValutAccessPolicy parameter -PermissionsToKeys specifies an array of key operation permissions to grant to a user or service principal.

The acceptable values for this parameter: decrypt, encrypt, unwrapKey, wrapKey, verify, sign, get, list, update, create, import, delete, backup, restore, recover, purge

Incorrect Answers:

A, C: The Set-AzureRmKeyValutAccessPolicy parameter -PermissionsToCertificates specifies an array of certificate permissions to grant to a user or service principal. The acceptable values for this parameter: get, list, delete, create, import, update, managecontacts, getissuers, listissuers, setissuers, deleteissuers, manageissuers, recover, purge, backup, restore

Reference:

<https://docs.microsoft.com/en-us/powershell/module/azurerm.keyvault/set-azurermkeyvaultaccesspolicy>

Question #1

## Introductory Info

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

Overview -

You are a developer for Contoso, Ltd. The company has a social networking website that is developed as a Single Page Application (SPA). The main web application for the social networking website loads user uploaded content from blob storage.

You are developing a solution to monitor uploaded data for inappropriate content. The following process occurs when users upload content by using the SPA:

- \* Messages are sent to ContentUploadService.
- \* Content is processed by ContentAnalysisService.
- \* After processing is complete, the content is posted to the social network or a rejection message is posted in its place.

The ContentAnalysisService is deployed with Azure Container Instances from a private Azure Container Registry named contosoimages.

The solution will use eight CPU cores.

Azure Active Directory -

Contoso, Ltd. uses Azure Active Directory (Azure AD) for both internal and guest accounts.

Requirements -

ContentAnalysisService -

The company's data science group built ContentAnalysisService which accepts user generated content as a string and returns a probable value for inappropriate content. Any values over a specific threshold must be reviewed by an employee of Contoso, Ltd.

You must create an Azure Function named CheckUserContent to perform the content checks.

Costs -

You must minimize costs for all Azure services.

Manual review -

To review content, the user must authenticate to the website portion of the ContentAnalysisService using their Azure AD credentials. The website is built using

React and all pages and API endpoints require authentication. In order to review content a user must be part of a ContentReviewer role. All completed reviews must include the reviewer's email address for auditing purposes.

High availability -

All services must run in multiple regions. The failure of any service in a region must not impact overall application availability.

Monitoring -

An alert must be raised if the ContentUploadService uses more than 80 percent of available CPU cores.

Security -

You have the following security requirements:

Any web service accessible over the Internet must be protected from cross site scripting attacks.

All websites and services must use SSL from a valid root certificate authority.

Azure Storage access keys must only be stored in memory and must be available only to the service.

All Internal services must only be accessible from internal Virtual Networks (VNets).

All parts of the system must support inbound and outbound traffic restrictions.

All service calls must be authenticated by using Azure AD.

User agreements -

When a user submits content, they must agree to a user agreement. The agreement allows employees of Contoso, Ltd. to review content, store cookies on user devices, and track user's IP addresses.

Information regarding agreements is used by multiple divisions within Contoso, Ltd.

User responses must not be lost and must be available to all parties regardless of individual service uptime. The volume of agreements is expected to be in the millions per hour.

Validation testing -

When a new version of the ContentAnalysisService is available the previous seven days of content must be processed with the new version to verify that the new version does not significantly deviate from the old version.

Issues -

Users of the ContentUploadService report that they occasionally see HTTP 502 responses on specific pages.

Code -

ContentUploadService -

```
CS01 apiVersion: '2018-10-01'
CS02 type: Microsoft.ContainerInstance/containerGroups
CS03 location: westus
CS04 name: contentUploadService
CS05 properties:
CS06   containers:
CS07     - name: service
CS08       properties:
CS09         image: contoso/contentUploadService:latest
CS10       ports:
CS11         - port: 80
CS12           protocol: TCP
CS13       resources:
CS14         requests:
CS15           cpu: 1.0
CS16           memoryInGB: 1.5
CS17
CS18 ipAddress:
CS19   ip: 10.23.121.112
CS20   ports:
CS21     - port: 80
CS22       protocol: TCP
CS23
CS24
CS25 networkProfile:
CS26
id: /subscriptions/98...19/resourceGroups/container/providers/Microsoft.Network/networkProfiles/subnet
```

**ApplicationManifest -**

```
AM01 {  
AM02     "id" : "2b079f03-9b06-2d44-98bb-e9182901fcb6",  
AM03     "appId" : "7118a7f0-b5c2-4c9d-833c-3d711396fe65",  
AM04  
AM05     "createdDateTime" : "2019-12-24T06:01:44Z",  
AM06     "logoUrl" : null,  
AM07     "logoutUrl" : null,  
AM08     "name" : "ContentAnalysisService",  
AM09  
AM10  
AM11     "orgRestrictions" : [],  
AM12     "parentalControlSettings" : {  
AM13         "countriesBlockedForMinors" : [],  
AM14         "legalAgeGroupRule" : "Allow"  
AM15     },  
AM16     "passwordCredentials" : []  
AM17 }
```

**Question**

You need to monitor ContentUploadService according to the requirements.

Which command should you use?

- A. az monitor metrics alert create &gt;g &gt; -scopes &gt; -condition "avg Percentage CPU > 8"
- B. az monitor metrics alert create &gt;g &gt; --scopes &gt; --condition "avg Percentage CPU > 800"
- C. az monitor metrics alert create &gt;g &gt; -scopes &gt; -condition "CPU Usage > 800"
- D. az monitor metrics alert create &gt;g &gt; -scopes &gt; -condition "CPU Usage > 8"

**Correct Answer:** B 

Scenario: An alert must be raised if the ContentUploadService uses more than 80 percent of available CPU cores

Reference:

<https://docs.microsoft.com/en-us/cli/azure/monitor/metrics/alert>

Question #2

## Introductory Info

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

Overview -

You are a developer for Contoso, Ltd. The company has a social networking website that is developed as a Single Page Application (SPA). The main web application for the social networking website loads user uploaded content from blob storage.

You are developing a solution to monitor uploaded data for inappropriate content. The following process occurs when users upload content by using the SPA:

- \* Messages are sent to ContentUploadService.
- \* Content is processed by ContentAnalysisService.
- \* After processing is complete, the content is posted to the social network or a rejection message is posted in its place.

The ContentAnalysisService is deployed with Azure Container Instances from a private Azure Container Registry named contosoimages.

The solution will use eight CPU cores.

Azure Active Directory -

Contoso, Ltd. uses Azure Active Directory (Azure AD) for both internal and guest accounts.

Requirements -

ContentAnalysisService -

The company's data science group built ContentAnalysisService which accepts user generated content as a string and returns a probable value for inappropriate content. Any values over a specific threshold must be reviewed by an employee of Contoso, Ltd.

You must create an Azure Function named CheckUserContent to perform the content checks.

Costs -

You must minimize costs for all Azure services.

Manual review -

To review content, the user must authenticate to the website portion of the ContentAnalysisService using their Azure AD credentials. The website is built using

React and all pages and API endpoints require authentication. In order to review content a user must be part of a ContentReviewer role. All completed reviews must include the reviewer's email address for auditing purposes.

High availability -

All services must run in multiple regions. The failure of any service in a region must not impact overall application availability.

Monitoring -

An alert must be raised if the ContentUploadService uses more than 80 percent of available CPU cores.

Security -

You have the following security requirements:

Any web service accessible over the Internet must be protected from cross site scripting attacks.

All websites and services must use SSL from a valid root certificate authority.

Azure Storage access keys must only be stored in memory and must be available only to the service.

All Internal services must only be accessible from internal Virtual Networks (VNets).

All parts of the system must support inbound and outbound traffic restrictions.

All service calls must be authenticated by using Azure AD.

User agreements -

When a user submits content, they must agree to a user agreement. The agreement allows employees of Contoso, Ltd. to review content, store cookies on user devices, and track user's IP addresses.

Information regarding agreements is used by multiple divisions within Contoso, Ltd.

User responses must not be lost and must be available to all parties regardless of individual service uptime. The volume of agreements is expected to be in the millions per hour.

Validation testing -

When a new version of the ContentAnalysisService is available the previous seven days of content must be processed with the new version to verify that the new version does not significantly deviate from the old version.

Issues -

Users of the ContentUploadService report that they occasionally see HTTP 502 responses on specific pages.

Code -

ContentUploadService -

```
CS01 apiVersion: '2018-10-01'
CS02 type: Microsoft.ContainerInstance/containerGroups
CS03 location: westus
CS04 name: contentUploadService
CS05 properties:
CS06   containers:
CS07     - name: service
CS08       properties:
CS09         image: contoso/contentUploadService:latest
CS10       ports:
CS11         - port: 80
CS12           protocol: TCP
CS13       resources:
CS14         requests:
CS15           cpu: 1.0
CS16           memoryInGB: 1.5
CS17
CS18 ipAddress:
CS19   ip: 10.23.121.112
CS20   ports:
CS21     - port: 80
CS22       protocol: TCP
CS23
CS24
CS25 networkProfile:
CS26
id: /subscriptions/98...19/resourceGroups/container/providers/Microsoft.Network/networkProfiles/subnet
```

**ApplicationManifest -**

```
AM01 {  
AM02     "id" : "2b079f03-9b06-2d44-98bb-e9182901fcb6",  
AM03     "appId" : "7118a7f0-b5c2-4c9d-833c-3d711396fe65",  
AM04  
AM05     "createdDateTime" : "2019-12-24T06:01:44Z",  
AM06     "logoUrl" : null,  
AM07     "logoutUrl" : null,  
AM08     "name" : "ContentAnalysisService",  
AM09  
AM10  
AM11     "orgRestrictions" : [],  
AM12     "parentalControlSettings" : {  
AM13         "countriesBlockedForMinors" : [],  
AM14         "legalAgeGroupRule" : "Allow"  
AM15     },  
AM16     "passwordCredentials" : []  
AM17 }
```

**Question**

You need to investigate the http server log output to resolve the issue with the ContentUploadService.

Which command should you use first?

- A. az webapp log
- B. az ams live-output
- C. az monitor activity-log
- D. az container attach

**Correct Answer:** C 

Scenario: Users of the ContentUploadService report that they occasionally see HTTP 502 responses on specific pages.

"502 bad gateway" and "503 service unavailable" are common errors in your app hosted in Azure App Service.

Microsoft Azure publicizes each time there is a service interruption or performance degradation.

The az monitor activity-log command manages activity logs.

Note: Troubleshooting can be divided into three distinct tasks, in sequential order:

1. Observe and monitor application behavior
2. Collect data
3. Mitigate the issue

Reference:

<https://docs.microsoft.com/en-us/cli/azure/monitor/activity-log>

**Topic 13 - Testlet 15**

Question #1

**Introductory Info**

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

City Power & Light company provides electrical infrastructure monitoring solutions for homes and businesses. The company is migrating solutions to Azure.

Current environment -

Architecture overview -

The company has a public website located at <http://www.cpndl.com/>. The site is a single-page web application that runs in Azure App Service on Linux. The website uses files stored in Azure Storage and cached in Azure Content Delivery Network (CDN) to serve static content.

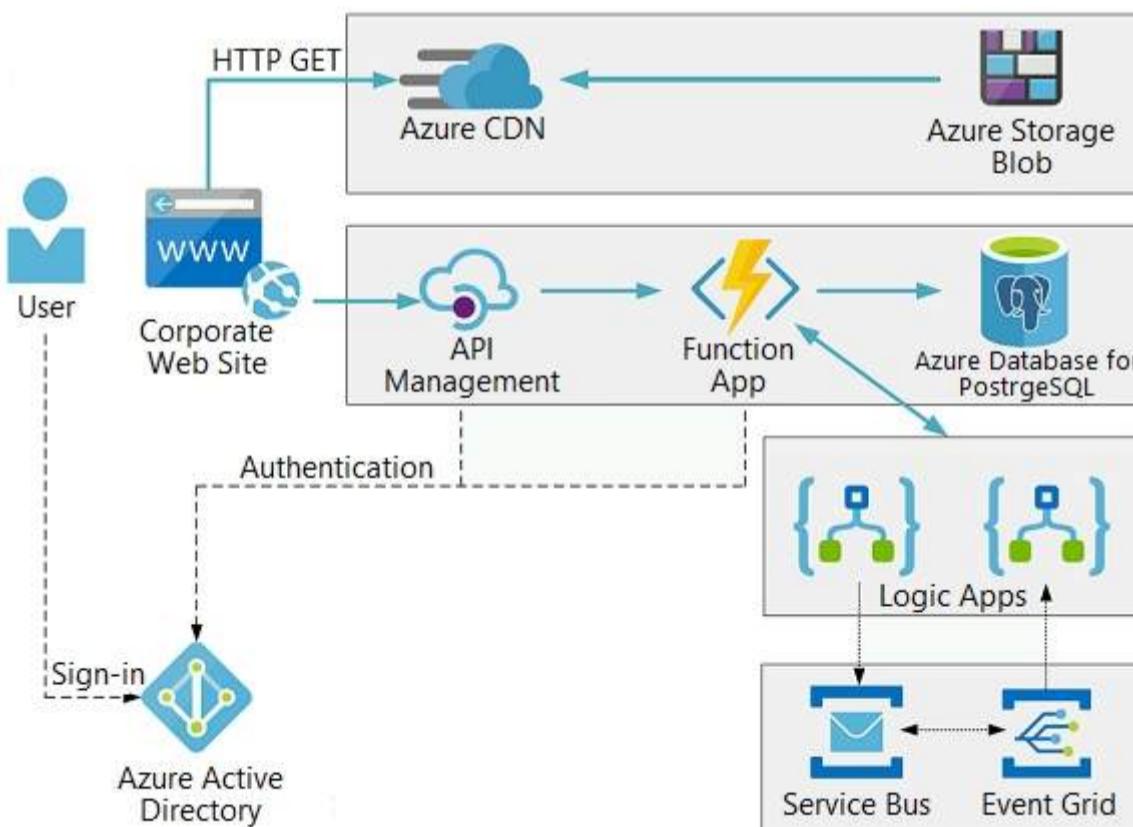
API Management and Azure Function App functions are used to process and store data in Azure Database for PostgreSQL. API Management is used to broker communications to the Azure Function app functions for Logic app integration. Logic apps are used to orchestrate the data processing while Service Bus and

Event Grid handle messaging and events.

The solution uses Application Insights, Azure Monitor, and Azure Key Vault.

Architecture diagram -

The company has several applications and services that support their business. The company plans to implement serverless computing where possible. The overall architecture is shown below.



User authentication -

The following steps detail the user authentication process:

1. The user selects Sign in in the website.
2. The browser redirects the user to the Azure Active Directory (Azure AD) sign in page.

3. The user signs in.
4. Azure AD redirects the user's session back to the web application. The URL includes an access token.
5. The web application calls an API and includes the access token in the authentication header. The application ID is sent as the audience ('`aud`') claim in the access token.
6. The back-end API validates the access token.

Requirements -

Corporate website -

Communications and content must be secured by using SSL.

Communications must use HTTPS.

Data must be replicated to a secondary region and three availability zones.

Data storage costs must be minimized.

Azure Database for PostgreSQL -

The database connection string is stored in Azure Key Vault with the following attributes:

Azure Key Vault name: cpandlkeyvault

Secret name: PostgreSQLConn

Id: 80df3e46ffcd4f1cb187f79905e9a1e8

The connection information is updated frequently. The application must always use the latest information to connect to the database.

Azure Service Bus and Azure Event Grid

Azure Event Grid must use Azure Service Bus for queue-based load leveling.

Events in Azure Event Grid must be routed directly to Service Bus queues for use in buffering.

Events from Azure Service Bus and other Azure services must continue to be routed to Azure Event Grid for processing.

Security -

All SSL certificates and credentials must be stored in Azure Key Vault.

File access must restrict access by IP, protocol, and Azure AD rights.

All user accounts and processes must receive only those privileges which are essential to perform their intended function.

Compliance -

Auditing of the file updates and transfers must be enabled to comply with General Data Protection Regulation (GDPR). The file updates must be read-only, stored in the order in which they occurred, include only create, update, delete, and copy operations, and be retained for compliance reasons.

Issues -

Corporate website -

While testing the site, the following error message displays:

CryptographicException: The system cannot find the file specified.

Function app -

You perform local testing for the RequestUserApproval function. The following error message displays:

'Timeout value of 00:10:00 exceeded by function: RequestUserApproval'

The same error message displays when you test the function in an Azure development environment when you run the following Kusto query:

FunctionAppLogs -

```
| where FunctionName == "RequestUserApproval"
```

Logic app -

You test the Logic app in a development environment. The following error message displays:

'400 Bad Request'

Troubleshooting of the error shows an HttpTrigger action to call the RequestUserApproval function.

Code -

Corporate website -

Security.cs:

```
SC01 public class Security
SC02 {
SC03 var bytes = System.IO.File.ReadAllBytes("~/var/ssl/private");
SC04 var cert = new System.Security.Cryptography.X509Certificate2(bytes);
SC05 var certName = cert.FriendlyName;
SC06 }
```

Function app -

RequestUserApproval.cs:

```
RA01 public static class RequestUserApproval
RA02 {
RA03 [FunctionName("RequestUserApproval")]
RA04 public static async Task<IActionResult> Run(
RA05 [HttpTrigger(AuthorizationLevel.Function, "get", "post", Route = null)] HttpRequest req,
RA06 ILogger log)
RA06 {
RA07     log.LogInformation("RequestUserApproval function processed a request.");
RA08 ...
RA09     return ProcessRequest(req)
RA10     ? (ActionResult)new OkObjectResult($"User approval processed")
RA11     : new BadRequestObjectResult("Failed to process user approval");
RA12 }
RA13 private static bool ProcessRequest(HttpRequest req)
RA14 {
RA15     ...
RA16 }
RA17 }
```

### Question

You need to investigate the Azure Function app error message in the development environment.

What should you do?

- A. Connect Live Metrics Stream from Application Insights to the Azure Function app and filter the metrics.
- B. Create a new Azure Log Analytics workspace and instrument the Azure Function app with Application Insights.
- C. Update the Azure Function app with extension methods from Microsoft.Extensions.Logging to log events by using the log instance.
- D. Add a new diagnostic setting to the Azure Function app to send logs to Log Analytics.

#### Correct Answer: A

Azure Functions offers built-in integration with Azure Application Insights to monitor functions.

The following areas of Application Insights can be helpful when evaluating the behavior, performance, and errors in your functions:

Live Metrics: View metrics data as it's created in near real-time.

Failures -

Performance -

Metrics -

Reference:

<https://docs.microsoft.com/en-us/azure/azure-functions/functions-monitoring>

Question #2

**Introductory Info**

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

City Power & Light company provides electrical infrastructure monitoring solutions for homes and businesses. The company is migrating solutions to Azure.

Current environment -

Architecture overview -

The company has a public website located at <http://www.cpndl.com/>. The site is a single-page web application that runs in Azure App Service on Linux. The website uses files stored in Azure Storage and cached in Azure Content Delivery Network (CDN) to serve static content.

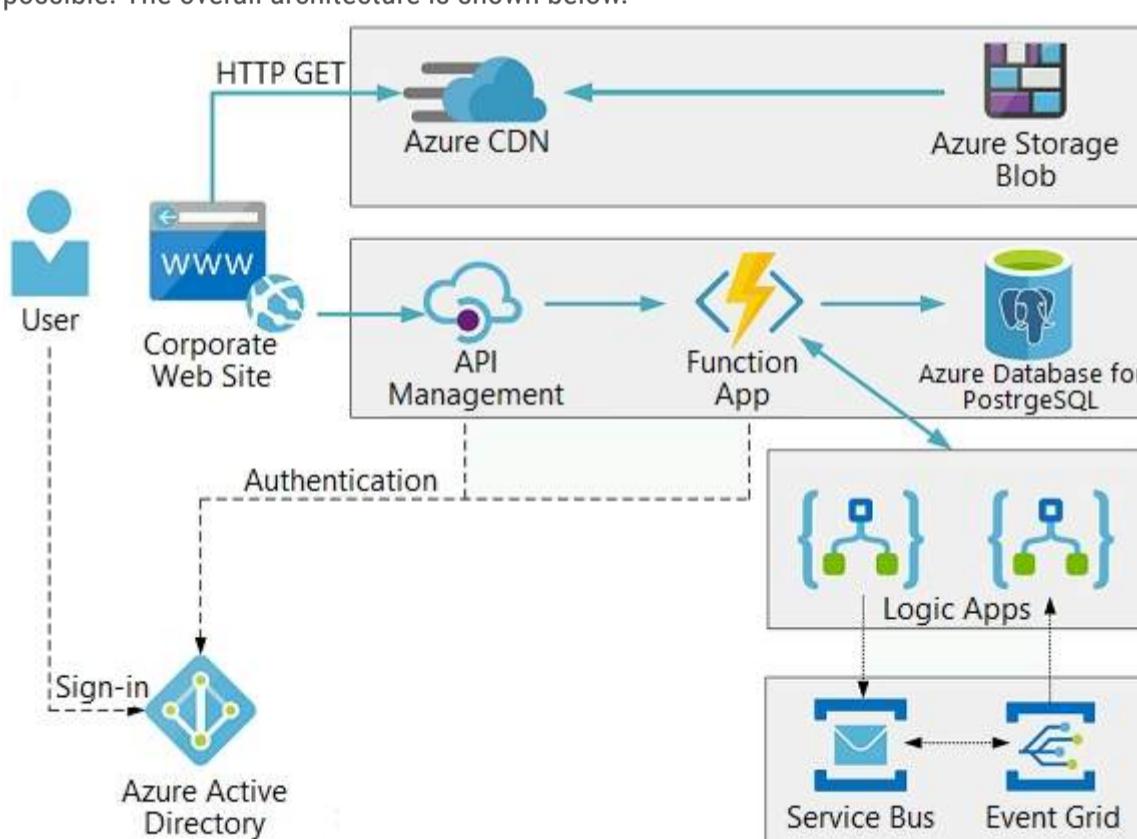
API Management and Azure Function App functions are used to process and store data in Azure Database for PostgreSQL. API Management is used to broker communications to the Azure Function app functions for Logic app integration. Logic apps are used to orchestrate the data processing while Service Bus and

Event Grid handle messaging and events.

The solution uses Application Insights, Azure Monitor, and Azure Key Vault.

Architecture diagram -

The company has several applications and services that support their business. The company plans to implement serverless computing where possible. The overall architecture is shown below.



User authentication -

The following steps detail the user authentication process:

1. The user selects Sign in in the website.

2. The browser redirects the user to the Azure Active Directory (Azure AD) sign in page.
3. The user signs in.
4. Azure AD redirects the user's session back to the web application. The URL includes an access token.
5. The web application calls an API and includes the access token in the authentication header. The application ID is sent as the audience ('`aud`') claim in the access token.
6. The back-end API validates the access token.

Requirements -

Corporate website -

Communications and content must be secured by using SSL.

Communications must use HTTPS.

Data must be replicated to a secondary region and three availability zones.

Data storage costs must be minimized.

Azure Database for PostgreSQL -

The database connection string is stored in Azure Key Vault with the following attributes:

Azure Key Vault name: cpndlkeyvault

Secret name: PostgreSQLConn

Id: 80df3e46ffcd4f1cb187f79905e9a1e8

The connection information is updated frequently. The application must always use the latest information to connect to the database.

Azure Service Bus and Azure Event Grid

Azure Event Grid must use Azure Service Bus for queue-based load leveling.

Events in Azure Event Grid must be routed directly to Service Bus queues for use in buffering.

Events from Azure Service Bus and other Azure services must continue to be routed to Azure Event Grid for processing.

Security -

All SSL certificates and credentials must be stored in Azure Key Vault.

File access must restrict access by IP, protocol, and Azure AD rights.

All user accounts and processes must receive only those privileges which are essential to perform their intended function.

Compliance -

Auditing of the file updates and transfers must be enabled to comply with General Data Protection Regulation (GDPR). The file updates must be read-only, stored in the order in which they occurred, include only create, update, delete, and copy operations, and be retained for compliance reasons.

Issues -

Corporate website -

While testing the site, the following error message displays:

CryptographicException: The system cannot find the file specified.

Function app -

You perform local testing for the RequestUserApproval function. The following error message displays:

'Timeout value of 00:10:00 exceeded by function: RequestUserApproval'

The same error message displays when you test the function in an Azure development environment when you run the following Kusto query:

FunctionAppLogs -

| where FunctionName == "RequestUserApproval"

Logic app -

You test the Logic app in a development environment. The following error message displays:

'400 Bad Request'

Troubleshooting of the error shows an HttpTrigger action to call the RequestUserApproval function.

Code -

Corporate website -

Security.cs:

```
SC01 public class Security
SC02 {
SC03     var bytes = System.IO.File.ReadAllBytes("~/var/ssl/private");
SC04     var cert = new System.Security.Cryptography.X509Certificate2(bytes);
SC05     var certName = cert.FriendlyName;
SC06 }
```

Function app -

RequestUserApproval.cs:

```
RA01 public static class RequestUserApproval
RA02 {
RA03     [FunctionName("RequestUserApproval")]
RA04     public static async Task<IActionResult> Run(
RA05         [HttpTrigger(AuthorizationLevel.Function, "get", "post", Route = null)] HttpRequest req,
RA06         ILogger log)
RA06 {
RA07     log.LogInformation("RequestUserApproval function processed a request.");
RA08     ...
RA09     return ProcessRequest(req)
RA10     ? (ActionResult)new OkObjectResult($"User approval processed")
RA11     : new BadRequestObjectResult("Failed to process user approval");
RA12 }
RA13 private static bool ProcessRequest(HttpRequest req)
RA14 {
RA15     ...
RA16 }
RA17 }
```

### Question

HOTSPOT -

You need to configure security and compliance for the corporate website files.

Which Azure Blob storage settings should you use? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

### Answer Area

Action	Setting
Restrict file access	<input type="checkbox"/> role-based access control (RBAC) <input type="checkbox"/> managed identity <input type="checkbox"/> shared access signature (SAS) token <input type="checkbox"/> connection string
Enable file auditing	<input type="checkbox"/> access tier <input type="checkbox"/> change feed <input type="checkbox"/> blob indexer <input type="checkbox"/> storage account type

## Answer Area

Action	Setting
Restrict file access	role-based access control (RBAC) managed identity shared access signature (SAS) token connection string
Enable file auditing	access tier change feed blob indexer storage account type

Correct Answer:

Box 1: role-based access control (RBAC)

Azure Storage supports authentication and authorization with Azure AD for the Blob and Queue services via Azure role-based access control (Azure RBAC).

Scenario: File access must restrict access by IP, protocol, and Azure AD rights.

Box 2: storage account type -

Scenario: The website uses files stored in Azure Storage

Auditing of the file updates and transfers must be enabled to comply with General Data Protection Regulation (GDPR).

Creating a diagnostic setting:

1. Sign in to the Azure portal.
2. Navigate to your storage account.
3. In the Monitoring section, click Diagnostic settings (preview).

The screenshot shows the Azure portal interface for managing diagnostic settings. The left sidebar has a navigation tree with 'Storage accounts' selected. Under 'Monitoring', 'Diagnostic settings (preview)' is highlighted with a red box. The main content area shows a table of resources:

NAME	RESOURCE TYPE	RESOURCE GROUP	DIAGNOSTICS STATUS
mystorageaccount	Storage account	my-resource-group	Disabled
blob	Storage account	my-resource-group	Disabled
queue	Storage account	my-resource-group	Disabled
table	Storage account	my-resource-group	Disabled
file	Storage account	my-resource-group	Disabled

4. Choose file as the type of storage that you want to enable logs for.

5. Click Add diagnostic setting.

Reference:

<https://docs.microsoft.com/en-us/azure/storage/common/storage-introduction> <https://docs.microsoft.com/en-us/azure/storage/files/storage-files-monitoring>

Question #1

## Introductory Info

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

You are a developer for Proseware, Inc. You are developing an application that applies a set of governance policies for Proseware's internal services, external services, and applications. The application will also provide a shared library for common functionality.

Requirements -

Policy service -

You develop and deploy a stateful ASP.NET Core 2.1 web application named Policy service to an Azure App Service Web App. The application reacts to events from Azure Event Grid and performs policy actions based on those events.

The application must include the Event Grid Event ID field in all Application Insights telemetry.

Policy service must use Application Insights to automatically scale with the number of policy actions that it is performing.

Policies -

Log policy -

All Azure App Service Web Apps must write logs to Azure Blob storage. All log files should be saved to a container named logdrop. Logs must remain in the container for 15 days.

Authentication events -

Authentication events are used to monitor users signing in and signing out. All authentication events must be processed by Policy service. Sign outs must be processed as quickly as possible.

PolicyLib -

You have a shared library named PolicyLib that contains functionality common to all ASP.NET Core web services and applications. The PolicyLib library must:

Exclude non-user actions from Application Insights telemetry.

Provide methods that allow a web service to scale itself.

Ensure that scaling actions do not disrupt application usage.

Other -

Anomaly detection service -

You have an anomaly detection service that analyzes log information for anomalies. It is implemented as an Azure Machine Learning model. The model is deployed as a web service. If an anomaly is detected, an Azure Function that emails administrators is called by using an HTTP WebHook.

Health monitoring -

All web applications and services have health monitoring at the /health service endpoint.

Issues -

**Policy loss -**

When you deploy Policy service, policies may not be applied if they were in the process of being applied during the deployment.

**Performance issue -**

When under heavy load, the anomaly detection service undergoes slowdowns and rejects connections.

**Notification latency -**

Users report that anomaly detection emails can sometimes arrive several minutes after an anomaly is detected.

**App code -****EventGridController.cs -**

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.

```
EventGridController.cs
EG01 public class EventGridController : Controller
EG02 {
EG03     public static AsyncLocal<string> EventId = new AsyncLocal<string>();
EG04     public IActionResult Process([FromBody] string eventsJson)
EG05     {
EG06         var events = JArray.Parse(eventsJson);
EG07
EG08         foreach (var @event in events)
EG09         {
EG10             EventId.Value = @event["id"].ToString();
EG11             if (@event["topic"].ToString().Contains("providers/Microsoft.Storage"))
EG12             {
EG13                 SendToAnomalyDetectionService(@event["data"]["url"].ToString());
EG14             }
EG15
EG16             {
EG17                 EnsureLogging(@event["subject"].ToString());
EG18             }
EG19         }
EG20         return null;
EG21     }
EG22     private void EnsureLogging(string resource)
EG23     {
EG24     . .
EG25     }
EG26     private async Task SendToAnomalyDetectionService(string uri)
EG27     {
EG28         var content = GetLogData(uri);
EG29         var scoreRequest = new
EG30         {
EG31             Inputs = new Dictionary<string, List<Dictionary<string, string>>>()
EG32             {
EG33                 {
EG34                     "input1",
EG35                     new List<Dictionary<string, string>>()
EG36                     {
EG37                         new Dictionary<string, string>()
EG38                         {
EG39                             {
EG40                                 "logcontent", content
EG41                             }
EG42                         }
EG43                     }
EG44                 },
EG45             },
EG46             GlobalParameters = new Dictionary<string, string>() { }
EG47         };
EG48         var result = await (new HttpClient()).PostAsJsonAsync("...", scoreRequest);
EG49         var rawModelResult = await result.Content.ReadAsStringAsync();
EG50         var modelResult = JObject.Parse(rawModelResult);
EG51         if (modelResult["notify"].HasValues)
EG52         {
EG53             . .
EG54         }
EG55     }
EG56     private (string name, string resourceGroup) ParseResourceId(string resourceId)
EG57     {
EG58     . .
EG59     }
EG60     private string GetLogData(string uri)
EG61     {
EG62     . .
EG63     }
EG64     static string BlobStoreAccountSAS(string containerName)
EG65     {
EG66     . .
EG67     }
EG68 }
```

LoginEvent.cs -

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.

```

LoginEvent.cs
LE01 public class LoginEvent
LE02 {
LE03
LE04     public string subject { get; set; }
LE05     public DateTime eventTime { get; set; }
LE06     public Dictionary<string, string> data { get; set; }
LE07     public string Serialize()
LE08     {
LE09         return JsonConvert.SerializeObject(this);
LE10     }
LE11 }
```

### Question

DRAG DROP -

You need to implement the Log policy.

How should you complete the Azure Event Grid subscription? To answer, drag the appropriate JSON segments to the correct locations. Each JSON segment may be used once, more than once, or not at all. You may need to drag the split bar between panes to view content.

NOTE: Each correct selection is worth one point.

Select and Place:

#### Code segment

- All
- WebHook
- EventHub
- subjectEndsWith
- Mictosoft.Storage
- subjectBeginsWith
- Microsoft.Storage.BlobCreated

#### Answer Area

```
{
    "name": "newlogs",
    "properties": {
        "topic": "/subscriptions/. . ./providers/Microsoft.EventGrid/topics/. . .",
        "destination": {
            "endpointType" : " [ code segment ] ",
            "filter": {
                "code segment" : "/blobServices/default/containers/logdrop/",
                "includedEventTypes": [ " [ code segment ] " ],
            },
            "labels": [],
            "eventDeliverySchema": "EventGridSchema"
        }
    }
}
```

#### Correct Answer:

#### Code segment

- All
- WebHook
- EventHub
- subjectEndsWith
- Mictosoft.Storage
- subjectBeginsWith
- Microsoft.Storage.BlobCreated

#### Answer Area

```
{
    "name": "newlogs",
    "properties": {
        "topic": "/subscriptions/. . ./providers/Microsoft.EventGrid/topics/. . .",
        "destination": {
            "endpointType" : " [ WebHook ] ",
            "filter": {
                "subjectBeginsWith" : "/blobServices/default/containers/logdrop/",
                "includedEventTypes": [ " [ Microsoft.Storage.BlobCreated ] " ],
            },
            "labels": [],
            "eventDeliverySchema": "EventGridSchema"
        }
    }
}
```

Box 1:WebHook -

Scenario: If an anomaly is detected, an Azure Function that emails administrators is called by using an HTTP WebHook. endpointType: The type of endpoint for the subscription (webhook/HTTP, Event Hub, or queue).

Box 2: SubjectBeginsWith -

Box 3: Microsoft.Storage.BlobCreated

Scenario: Log Policy -

All Azure App Service Web Apps must write logs to Azure Blob storage. All log files should be saved to a container named logdrop. Logs must remain in the container for 15 days.

Example subscription schema -

```
{
    "properties": {
        "destination": {
```

```
"endpointType": "webhook",
"properties": {
  "endpointUrl": "https://example.azurewebsites.net/api/HttpTriggerCSharp1?
code=VXbGWce53I48Mt8wuotr0GPmyJ/nDT4hgdFj9DpBiRt38qqnnm50Fg=="
},
},
"filter": {
  "includedEventTypes": [ "Microsoft.Storage.BlobCreated", "Microsoft.Storage.BlobDeleted" ],
  "subjectBeginsWith": "blobServices/default/containers/mycontainer/log",
  [1]
  "isSubjectCaseSensitive": "true"
}
}
```

Reference:

<https://docs.microsoft.com/en-us/azure/event-grid/subscription-creation-schema>

Question #2

## Introductory Info

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

You are a developer for Proseware, Inc. You are developing an application that applies a set of governance policies for Proseware's internal services, external services, and applications. The application will also provide a shared library for common functionality.

Requirements -

Policy service -

You develop and deploy a stateful ASP.NET Core 2.1 web application named Policy service to an Azure App Service Web App. The application reacts to events from Azure Event Grid and performs policy actions based on those events.

The application must include the Event Grid Event ID field in all Application Insights telemetry.

Policy service must use Application Insights to automatically scale with the number of policy actions that it is performing.

Policies -

Log policy -

All Azure App Service Web Apps must write logs to Azure Blob storage. All log files should be saved to a container named logdrop. Logs must remain in the container for 15 days.

Authentication events -

Authentication events are used to monitor users signing in and signing out. All authentication events must be processed by Policy service. Sign outs must be processed as quickly as possible.

PolicyLib -

You have a shared library named PolicyLib that contains functionality common to all ASP.NET Core web services and applications. The PolicyLib library must:

Exclude non-user actions from Application Insights telemetry.

Provide methods that allow a web service to scale itself.

Ensure that scaling actions do not disrupt application usage.

Other -

Anomaly detection service -

You have an anomaly detection service that analyzes log information for anomalies. It is implemented as an Azure Machine Learning model. The model is deployed as a web service. If an anomaly is detected, an Azure Function that emails administrators is called by using an HTTP WebHook.

Health monitoring -

All web applications and services have health monitoring at the /health service endpoint.

Issues -

**Policy loss -**

When you deploy Policy service, policies may not be applied if they were in the process of being applied during the deployment.

**Performance issue -**

When under heavy load, the anomaly detection service undergoes slowdowns and rejects connections.

**Notification latency -**

Users report that anomaly detection emails can sometimes arrive several minutes after an anomaly is detected.

**App code -****EventGridController.cs -**

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.

```
EventGridController.cs
EG01 public class EventGridController : Controller
EG02 {
EG03     public static AsyncLocal<string> EventId = new AsyncLocal<string>();
EG04     public IActionResult Process([FromBody] string eventsJson)
EG05     {
EG06         var events = JArray.Parse(eventsJson);
EG07
EG08         foreach (var @event in events)
EG09         {
EG10             EventId.Value = @event["id"].ToString();
EG11             if (@event["topic"].ToString().Contains("providers/Microsoft.Storage"))
EG12             {
EG13                 SendToAnomalyDetectionService(@event["data"]["url"].ToString());
EG14             }
EG15
EG16             {
EG17                 EnsureLogging(@event["subject"].ToString());
EG18             }
EG19         }
EG20         return null;
EG21     }
EG22     private void EnsureLogging(string resource)
EG23     {
EG24     . .
EG25     }
EG26     private async Task SendToAnomalyDetectionService(string uri)
EG27     {
EG28         var content = GetLogData(uri);
EG29         var scoreRequest = new
EG30         {
EG31             Inputs = new Dictionary<string, List<Dictionary<string, string>>>()
EG32             {
EG33                 {
EG34                     "input1",
EG35                     new List<Dictionary<string, string>>()
EG36                     {
EG37                         new Dictionary<string, string>()
EG38                         {
EG39                             {
EG40                                 "logcontent", content
EG41                             }
EG42                         }
EG43                     }
EG44                 },
EG45             },
EG46             GlobalParameters = new Dictionary<string, string>() { }
EG47         };
EG48         var result = await (new HttpClient()).PostAsJsonAsync("...", scoreRequest);
EG49         var rawModelResult = await result.Content.ReadAsStringAsync();
EG50         var modelResult = JObject.Parse(rawModelResult);
EG51         if (modelResult["notify"].HasValues)
EG52         {
EG53             . .
EG54         }
EG55     }
EG56     private (string name, string resourceGroup) ParseResourceId(string resourceId)
EG57     {
EG58     . .
EG59     }
EG60     private string GetLogData(string uri)
EG61     {
EG62     . .
EG63     }
EG64     static string BlobStoreAccountSAS(string containerName)
EG65     {
EG66     . .
EG67     }
EG68 }
```

LoginEvent.cs -

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.

```
LoginEvent.cs
LE01 public class LoginEvent
LE02 {
LE03
LE04     public string subject { get; set; }
LE05     public DateTime eventTime { get; set; }
LE06     public Dictionary<string, string> data { get; set; }
LE07     public string Serialize()
LE08     {
LE09         return JsonConvert.SerializeObject(this);
LE10     }
LE11 }
```

### Question

You need to ensure that the solution can meet the scaling requirements for Policy Service.

Which Azure Application Insights data model should you use?

- A. an Application Insights dependency
- B. an Application Insights event
- C. an Application Insights trace
- D. an Application Insights metric

#### Correct Answer: D

Application Insights provides three additional data types for custom telemetry:

Trace - used either directly, or through an adapter to implement diagnostics logging using an instrumentation framework that is familiar to you, such as Log4Net or

System.Diagnostics.

Event - typically used to capture user interaction with your service, to analyze usage patterns.

Metric - used to report periodic scalar measurements.

Scenario:

Policy service must use Application Insights to automatically scale with the number of policy actions that it is performing.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-monitor/app/data-model>

Question #3

## Introductory Info

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

You are a developer for Proseware, Inc. You are developing an application that applies a set of governance policies for Proseware's internal services, external services, and applications. The application will also provide a shared library for common functionality.

Requirements -

Policy service -

You develop and deploy a stateful ASP.NET Core 2.1 web application named Policy service to an Azure App Service Web App. The application reacts to events from Azure Event Grid and performs policy actions based on those events.

The application must include the Event Grid Event ID field in all Application Insights telemetry.

Policy service must use Application Insights to automatically scale with the number of policy actions that it is performing.

Policies -

Log policy -

All Azure App Service Web Apps must write logs to Azure Blob storage. All log files should be saved to a container named logdrop. Logs must remain in the container for 15 days.

Authentication events -

Authentication events are used to monitor users signing in and signing out. All authentication events must be processed by Policy service. Sign outs must be processed as quickly as possible.

PolicyLib -

You have a shared library named PolicyLib that contains functionality common to all ASP.NET Core web services and applications. The PolicyLib library must:

Exclude non-user actions from Application Insights telemetry.

Provide methods that allow a web service to scale itself.

Ensure that scaling actions do not disrupt application usage.

Other -

Anomaly detection service -

You have an anomaly detection service that analyzes log information for anomalies. It is implemented as an Azure Machine Learning model. The model is deployed as a web service. If an anomaly is detected, an Azure Function that emails administrators is called by using an HTTP WebHook.

Health monitoring -

All web applications and services have health monitoring at the /health service endpoint.

Issues -

**Policy loss -**

When you deploy Policy service, policies may not be applied if they were in the process of being applied during the deployment.

**Performance issue -**

When under heavy load, the anomaly detection service undergoes slowdowns and rejects connections.

**Notification latency -**

Users report that anomaly detection emails can sometimes arrive several minutes after an anomaly is detected.

**App code -****EventGridController.cs -**

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.

```
EventGridController.cs
EG01 public class EventGridController : Controller
EG02 {
EG03     public static AsyncLocal<string> EventId = new AsyncLocal<string>();
EG04     public IActionResult Process([FromBody] string eventsJson)
EG05     {
EG06         var events = JArray.Parse(eventsJson);
EG07
EG08         foreach (var @event in events)
EG09         {
EG10             EventId.Value = @event["id"].ToString();
EG11             if (@event["topic"].ToString().Contains("providers/Microsoft.Storage"))
EG12             {
EG13                 SendToAnomalyDetectionService(@event["data"]["url"].ToString());
EG14             }
EG15
EG16             {
EG17                 EnsureLogging(@event["subject"].ToString());
EG18             }
EG19         }
EG20         return null;
EG21     }
EG22     private void EnsureLogging(string resource)
EG23     {
EG24     . .
EG25     }
EG26     private async Task SendToAnomalyDetectionService(string uri)
EG27     {
EG28         var content = GetLogData(uri);
EG29         var scoreRequest = new
EG30         {
EG31             Inputs = new Dictionary<string, List<Dictionary<string, string>>>()
EG32             {
EG33                 {
EG34                     "input1",
EG35                     new List<Dictionary<string, string>>()
EG36                     {
EG37                         new Dictionary<string, string>()
EG38                         {
EG39                             {
EG40                                 "logcontent", content
EG41                             }
EG42                         }
EG43                     }
EG44                 },
EG45             },
EG46             GlobalParameters = new Dictionary<string, string>() { }
EG47         };
EG48         var result = await (new HttpClient()).PostAsJsonAsync("...", scoreRequest);
EG49         var rawModelResult = await result.Content.ReadAsStringAsync();
EG50         var modelResult = JObject.Parse(rawModelResult);
EG51         if (modelResult["notify"].HasValues)
EG52         {
EG53             . .
EG54         }
EG55     }
EG56     private (string name, string resourceGroup) ParseResourceId(string resourceId)
EG57     {
EG58     . .
EG59     }
EG60     private string GetLogData(string uri)
EG61     {
EG62     . .
EG63     }
EG64     static string BlobStoreAccountSAS(string containerName)
EG65     {
EG66     . .
EG67     }
EG68 }
```

LoginEvent.cs -

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.

```
>LoginEvent.cs  
LE01 public class LoginEvent  
LE02 {  
LE03  
LE04     public string subject { get; set; }  
LE05     public DateTime eventTime { get; set; }  
LE06     public Dictionary<string, string> data { get; set; }  
LE07     public string Serialize()  
LE08     {  
LE09         return JsonConvert.SerializeObject(this);  
LE10     }  
LE11 }
```

### Question

DRAG DROP -

You need to implement telemetry for non-user actions.

How should you complete the Filter class? 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 one point.

Select and Place:

#### Code segments

```
/health  
/status  
RequestTelemetry  
PageViewTelemetry  
ITelemetryProcessor  
ITelemetryInitializer
```

#### Answer Area

```
public class Filter : code segment  
{  
    private readonly code segment _next;  
    public (Filter code segment next)  
    {  
        _next = next;  
    }  
    public void Process(ITelemetry item)  
    {  
        var x = item as code segment;  
        if (x?.Url.AbsolutePath == "code segment")  
        {  
            return;  
        }  
        _next.Process(item);  
    }  
}
```

Correct Answer:

### Code segments

/health  
/status  
RequestTelemetry  
PageViewTelemetry  
ITelemetryProcessor  
ITelemetryInitializer

### Answer Area

```
public class Filter : ITelemetryProcessor
{
    private readonly ITelemetryProcessor _next;
    public (Filter ITelemetryProcessor next)
    {
        _next = next;
    }
    public void Process(ITelemetry item)
    {
        var x = item as RequestTelemetry ;
        if (x?.Url.AbsolutePath == "/health" )
        {
            return;
        }
        _next.Process(item);
    }
}
```

Scenario: Exclude non-user actions from Application Insights telemetry.

Box 1: ITelemetryProcessor -

To create a filter, implement ITelemetryProcessor. This technique gives you more direct control over what is included or excluded from the telemetry stream.

Box 2: ITelemetryProcessor -

Box 3: ITelemetryProcessor -

Box 4: RequestTelemetry -

Box 5: /health -

To filter out an item, just terminate the chain.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-monitor/app/api-filtering-sampling>

Question #4

## Introductory Info

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

You are a developer for Proseware, Inc. You are developing an application that applies a set of governance policies for Proseware's internal services, external services, and applications. The application will also provide a shared library for common functionality.

Requirements -

Policy service -

You develop and deploy a stateful ASP.NET Core 2.1 web application named Policy service to an Azure App Service Web App. The application reacts to events from Azure Event Grid and performs policy actions based on those events.

The application must include the Event Grid Event ID field in all Application Insights telemetry.

Policy service must use Application Insights to automatically scale with the number of policy actions that it is performing.

Policies -

Log policy -

All Azure App Service Web Apps must write logs to Azure Blob storage. All log files should be saved to a container named logdrop. Logs must remain in the container for 15 days.

Authentication events -

Authentication events are used to monitor users signing in and signing out. All authentication events must be processed by Policy service. Sign outs must be processed as quickly as possible.

PolicyLib -

You have a shared library named PolicyLib that contains functionality common to all ASP.NET Core web services and applications. The PolicyLib library must:

Exclude non-user actions from Application Insights telemetry.

Provide methods that allow a web service to scale itself.

Ensure that scaling actions do not disrupt application usage.

Other -

Anomaly detection service -

You have an anomaly detection service that analyzes log information for anomalies. It is implemented as an Azure Machine Learning model. The model is deployed as a web service. If an anomaly is detected, an Azure Function that emails administrators is called by using an HTTP WebHook.

Health monitoring -

All web applications and services have health monitoring at the /health service endpoint.

Issues -

**Policy loss -**

When you deploy Policy service, policies may not be applied if they were in the process of being applied during the deployment.

**Performance issue -**

When under heavy load, the anomaly detection service undergoes slowdowns and rejects connections.

**Notification latency -**

Users report that anomaly detection emails can sometimes arrive several minutes after an anomaly is detected.

**App code -****EventGridController.cs -**

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.

```
EventGridController.cs
EG01 public class EventGridController : Controller
EG02 {
EG03     public static AsyncLocal<string> EventId = new AsyncLocal<string>();
EG04     public IActionResult Process([FromBody] string eventsJson)
EG05     {
EG06         var events = JArray.Parse(eventsJson);
EG07
EG08         foreach (var @event in events)
EG09         {
EG10             EventId.Value = @event["id"].ToString();
EG11             if (@event["topic"].ToString().Contains("providers/Microsoft.Storage"))
EG12             {
EG13                 SendToAnomalyDetectionService(@event["data"]["url"].ToString());
EG14             }
EG15
EG16             {
EG17                 EnsureLogging(@event["subject"].ToString());
EG18             }
EG19         }
EG20         return null;
EG21     }
EG22     private void EnsureLogging(string resource)
EG23     {
EG24     . .
EG25     }
EG26     private async Task SendToAnomalyDetectionService(string uri)
EG27     {
EG28         var content = GetLogData(uri);
EG29         var scoreRequest = new
EG30         {
EG31             Inputs = new Dictionary<string, List<Dictionary<string, string>>>()
EG32             {
EG33                 {
EG34                     "input1",
EG35                     new List<Dictionary<string, string>>()
EG36                     {
EG37                         new Dictionary<string, string>()
EG38                         {
EG39                             {
EG40                                 "logcontent", content
EG41                             }
EG42                         }
EG43                     }
EG44                 },
EG45             },
EG46             GlobalParameters = new Dictionary<string, string>() { }
EG47         };
EG48         var result = await (new HttpClient()).PostAsJsonAsync("...", scoreRequest);
EG49         var rawModelResult = await result.Content.ReadAsStringAsync();
EG50         var modelResult = JObject.Parse(rawModelResult);
EG51         if (modelResult["notify"].HasValues)
EG52         {
EG53             . .
EG54         }
EG55     }
EG56     private (string name, string resourceGroup) ParseResourceId(string resourceId)
EG57     {
EG58     . .
EG59     }
EG60     private string GetLogData(string uri)
EG61     {
EG62     . .
EG63     }
EG64     static string BlobStoreAccountSAS(string containerName)
EG65     {
EG66     . .
EG67     }
EG68 }
```

LoginEvent.cs -

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.

#### LoginEvent.cs

```
LE01 public class LoginEvent
LE02 {
LE03
LE04     public string subject { get; set; }
LE05     public DateTime eventTime { get; set; }
LE06     public Dictionary<string, string> data { get; set; }
LE07     public string Serialize()
LE08     {
LE09         return JsonConvert.SerializeObject(this);
LE10     }
LE11 }
```

#### Question

DRAG DROP -

You need to ensure that PolicyLib requirements are met.

How should you complete the code segment? 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 one point.

Select and Place:

#### Code segments

```
Process
Initialize
telemetry.Sequence
ITelemetryProcessor
ITelemetryInitializer
telemetry.Context
EventGridController.EventId.Value
((EventTelemetry)telemetry).Properties["EventId"]
```

#### Answer Area

```
public class IncludeEventId : ITelemetryInitializer
{
    public void Initialize(ITelemetry telemetry)
    {
        ((EventTelemetry)telemetry).Properties["EventId"] =
    }
}
```

#### Correct Answer:

#### Code segments

```
Process
Initialize
telemetry.Sequence
ITelemetryProcessor
```

#### Answer Area

```
public class IncludeEventId : ITelemetryInitializer
{
    public void Initialize(ITelemetry telemetry)
    {
    }
}
```

#### Topic 15 - Testlet 17

```
EventGridController.EventId.Value
((EventTelemetry)telemetry).Properties["EventId"] }
```

Scenario: You have a shared library named PolicyLib that contains functionality common to all ASP.NET Core web services and applications.

The PolicyLib library must:

- Exclude non-user actions from Application Insights telemetry.
- Provide methods that allow a web service to scale itself.
- Ensure that scaling actions do not disrupt application usage.

Box 1: ITelemetryInitializer -

Use telemetry initializers to define global properties that are sent with all telemetry; and to override selected behavior of the standard telemetry modules.

Box 2: Initialize -

Box 3: Telemetry.Context -

Box 4: ((EventTelemetry)telemetry).Properties["EventID"]

Reference:

<https://docs.microsoft.com/en-us/azure/azure-monitor/app/api-filtering-sampling>

Question #1

## Introductory Info

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 questions in 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 section 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 in 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 SaaS company that provides a solution for managing employee expenses. The solution consists of an ASP.NET Core Web

API project that is deployed as an Azure Web App.

Overall architecture -

Employees upload receipts for the system to process. When processing is complete, the employee receives a summary report email that details the processing results. Employees then use a web application to manage their receipts and perform any additional tasks needed for reimbursement.

Receipt processing -

Employees may upload receipts in two ways:

Uploading using an Azure Files mounted folder

Uploading using the web application

Data Storage -

Receipt and employee information is stored in an Azure SQL database.

Documentation -

Employees are provided with a getting started document when they first use the solution. The documentation includes details on supported operating systems for

Azure File upload, and instructions on how to configure the mounted folder.

Solution details -

Users table -

Column	Description
UserId	unique identifier for an employee
ExpenseAccount	employees expense account number in the format 1234-123-1234
AllowedAmount	limit of allowed expenses before approval is needed
SupervisorId	unique identifier for employee's supervisor
SecurityPin	value used to validate user identity

Web Application -

You enable MSI for the Web App and configure the Web App to use the security principal name WebAppIdentity.

Processing -

Processing is performed by an Azure Function that uses version 2 of the Azure Function runtime. Once processing is completed, results are stored in Azure Blob

Storage and an Azure SQL database. Then, an email summary is sent to the user with a link to the processing report. The link to the report must remain valid if the email is forwarded to another user.

#### Logging -

Azure Application Insights is used for telemetry and logging in both the processor and the web application. The processor also has TraceWriter logging enabled.

Application Insights must always contain all log messages.

#### Requirements -

##### Receipt processing -

Concurrent processing of a receipt must be prevented.

##### Disaster recovery -

Regional outage must not impact application availability. All DR operations must not be dependent on application running and must ensure that data in the DR region is up to date.

#### Security -

User's SecurityPin must be stored in such a way that access to the database does not allow the viewing of SecurityPins. The web application is the only system that should have access to SecurityPins.

All certificates and secrets used to secure data must be stored in Azure Key Vault.

You must adhere to the principle of least privilege and provide privileges which are essential to perform the intended function.

All access to Azure Storage and Azure SQL database must use the application's Managed Service Identity (MSI).

Receipt data must always be encrypted at rest.

All data must be protected in transit.

User's expense account number must be visible only to logged in users. All other views of the expense account number should include only the last segment, with the remaining parts obscured.

In the case of a security breach, access to all summary reports must be revoked without impacting other parts of the system.

#### Issues -

##### Upload format issue -

Employees occasionally report an issue with uploading a receipt using the web application. They report that when they upload a receipt using the Azure File

Share, the receipt does not appear in their profile. When this occurs, they delete the file in the file share and use the web application, which returns a 500 Internal

Server error page.

##### Capacity issue -

During busy periods, employees report long delays between the time they upload the receipt and when it appears in the web application.

##### Log capacity issue -

Developers report that the number of log messages in the trace output for the processor is too high, resulting in lost log messages.

#### Application code -

##### Processing.cs -

```

PC01 public static class Processing
PC02 {
PC03     public static class Function
PC04     {
PC05         [FunctionName("IssueWork")]
PC06         public static async Task Run([TimerTrigger("0 */5 * * *")] TimerInfo timer, ILogger
log)
PC07         {
PC08             var container = await GetCloudBlobContainer();
PC09             foreach (var fileItem in await ListFiles())
PC10             {
PC11                 var file = new CloudFile(fileItem.StorageUri.PrimaryUri);
PC12                 var ms = new MemoryStream();
PC13                 await file.DownloadToStreamAsync(ms);
PC14                 var blob = container.GetBlockBlobReference(fileItem.Uri.ToString());
PC15                 await blob.UploadFromStreamAsync(ms);
PC16
PC17             }
PC18         }
PC19         private static CloudBlockBlob GetDRBlock(CloudBlockBlob sourceBlob)
PC20         {
PC21             . . .
PC22         }
PC23         private static async Task<CloudBlobContainer> GetCloudBlobContainer()
PC24         {
PC25             var cloudBlobClient = new CloudBlobClient(new Uri(" . . ."), await GetCredentials());
PC26
PC27             await cloudBlobClient.GetRootContainerReference().CreateIfNotExistsAsync();
PC28             return cloudBlobClient.GetRootContainerReference();
PC29         }
PC30         private static async Task<StorageCredentials> GetCredentials()
PC31         {
PC32             . . .
PC33         }
PC34         private static async Task<List<IListFileItem>> ListFiles()
PC35         {
PC36             . . .
PC37         }
PC38         private KeyVaultClient _keyVaultClient = new KeyVaultClient(" . . .");
PC39     }

```

Database.cs -

```

DB01 public class Database
DB02 {
DB03     private string ConnectionString =
DB04
DB05     public async Task<object> LoadUserDetails(string userId)
DB06     {
DB07
DB08         return await policy.ExecuteAsync(async () =>
DB09             {
DB10                 using (var connection = new SqlConnection(ConnectionString))
DB11                 {
DB12                     await connection.OpenAsync();
DB13                     using (var command = new SqlCommand(" . . .", connection))
DB14                     using (var reader = command.ExecuteReader())
DB15                     {
DB16                         . . .
DB17                     }
DB18                 }
DB19             });
DB20         }
DB21     }

```

## ReceiptUploader.cs -

```

RU01 public class ReceiptUploader
RU02 {
RU03     public async Task UploadFile(string file, byte[] binary)
RU04     {
RU05         var httpClient = new HttpClient();
RU06         var response = await httpClient.PutAsync("...", new ByteArrayContent(binary));
RU07         while (ShouldRetry(response))
RU08         {
RU09             response = await httpClient.PutAsync("...", new ByteArrayContent(binary));
RU10         }
RU11     }
RU12     private bool ShouldRetry(HttpStatusCode response)
RU13     {
RU14     }
RU15 }
RU16 }
```

## ConfigureSSE.ps1 -

```

CS01 $storageAccount = Get-AzureRmStorageAccount -ResourceGroupName "..." -AccountName "..."
CS02 $keyVault = Get-AzureRmKeyVault -VaultName "..."
CS03 $key = Get-AzureKeyVaultKey -VaultName $keyVault.VaultName -Name "..."
CS04 Set-AzureRmKeyVaultAccessPolicy ` 
CS05     -VaultName $keyVault.VaultName ` 
CS06     -ObjectId $storageAccount.Identity.PrincipalId ` 
CS07
CS08
CS09 Set-AzureRmStorageAccount ` 
CS10     -ResourceGroupName $storageAccount.ResourceGroupName ` 
CS11     -AccountName $storageAccount.StorageAccountName ` 
CS12     -EnableEncryptionService File ` 
CS13     -KeyvaultEncryption ` 
CS14     -KeyName $key.Name 
CS15     -KeyVersion $key.Version ` 
CS16     -KeyVaultUri $keyVault.VaultUri
```

**Question**

You need to ensure receipt processing occurs correctly.

What should you do?

- A. Use blob properties to prevent concurrency problems
- B. Use blob SnapshotTime to prevent concurrency problems
- C. Use blob metadata to prevent concurrency problems
- D. Use blob leases to prevent concurrency problems

**Correct Answer: B**

You can create a snapshot of a blob. A snapshot is a read-only version of a blob that's taken at a point in time. Once a snapshot has been created, it can be read, copied, or deleted, but not modified. Snapshots provide a way to back up a blob as it appears at a moment in time.

**Scenario:** Processing is performed by an Azure Function that uses version 2 of the Azure Function runtime. Once processing is completed, results are stored in

Azure Blob Storage and an Azure SQL database. Then, an email summary is sent to the user with a link to the processing report. The link to the report must remain valid if the email is forwarded to another user.

Reference:

<https://docs.microsoft.com/en-us/rest/api/storageservices/creating-a-snapshot-of-a-blob>

Question #2

## Introductory Info

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 questions in 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 section 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 in 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 SaaS company that provides a solution for managing employee expenses. The solution consists of an ASP.NET Core Web API project that is deployed as an Azure Web App.

Overall architecture -

Employees upload receipts for the system to process. When processing is complete, the employee receives a summary report email that details the processing results. Employees then use a web application to manage their receipts and perform any additional tasks needed for reimbursement.

Receipt processing -

Employees may upload receipts in two ways:

Uploading using an Azure Files mounted folder

Uploading using the web application

Data Storage -

Receipt and employee information is stored in an Azure SQL database.

Documentation -

Employees are provided with a getting started document when they first use the solution. The documentation includes details on supported operating systems for

Azure File upload, and instructions on how to configure the mounted folder.

Solution details -

Users table -

Column	Description
UserId	unique identifier for an employee
ExpenseAccount	employees expense account number in the format 1234-123-1234
AllowedAmount	limit of allowed expenses before approval is needed
SupervisorId	unique identifier for employee's supervisor
SecurityPin	value used to validate user identity

Web Application -

You enable MSI for the Web App and configure the Web App to use the security principal name WebAppIdentity.

Processing -

Processing is performed by an Azure Function that uses version 2 of the Azure Function runtime. Once processing is completed, results are stored in Azure Blob

Storage and an Azure SQL database. Then, an email summary is sent to the user with a link to the processing report. The link to the report must remain valid if the email is forwarded to another user.

#### Logging -

Azure Application Insights is used for telemetry and logging in both the processor and the web application. The processor also has TraceWriter logging enabled.

Application Insights must always contain all log messages.

#### Requirements -

##### Receipt processing -

Concurrent processing of a receipt must be prevented.

##### Disaster recovery -

Regional outage must not impact application availability. All DR operations must not be dependent on application running and must ensure that data in the DR region is up to date.

#### Security -

User's SecurityPin must be stored in such a way that access to the database does not allow the viewing of SecurityPins. The web application is the only system that should have access to SecurityPins.

All certificates and secrets used to secure data must be stored in Azure Key Vault.

You must adhere to the principle of least privilege and provide privileges which are essential to perform the intended function.

All access to Azure Storage and Azure SQL database must use the application's Managed Service Identity (MSI).

Receipt data must always be encrypted at rest.

All data must be protected in transit.

User's expense account number must be visible only to logged in users. All other views of the expense account number should include only the last segment, with the remaining parts obscured.

In the case of a security breach, access to all summary reports must be revoked without impacting other parts of the system.

#### Issues -

##### Upload format issue -

Employees occasionally report an issue with uploading a receipt using the web application. They report that when they upload a receipt using the Azure File

Share, the receipt does not appear in their profile. When this occurs, they delete the file in the file share and use the web application, which returns a 500 Internal

Server error page.

##### Capacity issue -

During busy periods, employees report long delays between the time they upload the receipt and when it appears in the web application.

##### Log capacity issue -

Developers report that the number of log messages in the trace output for the processor is too high, resulting in lost log messages.

#### Application code -

##### Processing.cs -

```

PC01 public static class Processing
PC02 {
PC03     public static class Function
PC04     {
PC05         [FunctionName("IssueWork")]
PC06         public static async Task Run([TimerTrigger("0 */5 * * *")] TimerInfo timer, ILogger
log)
PC07         {
PC08             var container = await GetCloudBlobContainer();
PC09             foreach (var fileItem in await ListFiles())
PC10             {
PC11                 var file = new CloudFile(fileItem.StorageUri.PrimaryUri);
PC12                 var ms = new MemoryStream();
PC13                 await file.DownloadToStreamAsync(ms);
PC14                 var blob = container.GetBlockBlobReference(fileItem.Uri.ToString());
PC15                 await blob.UploadFromStreamAsync(ms);
PC16
PC17             }
PC18         }
PC19         private static CloudBlockBlob GetDRBlock(CloudBlockBlob sourceBlob)
PC20         {
PC21             . . .
PC22         }
PC23         private static async Task<CloudBlobContainer> GetCloudBlobContainer()
PC24         {
PC25             var cloudBlobClient = new CloudBlobClient(new Uri(" . . ."), await GetCredentials());
PC26
PC27             await cloudBlobClient.GetRootContainerReference().CreateIfNotExistsAsync();
PC28             return cloudBlobClient.GetRootContainerReference();
PC29         }
PC30         private static async Task<StorageCredentials> GetCredentials()
PC31         {
PC32             . . .
PC33         }
PC34         private static async Task<List<IListFileItem>> ListFiles()
PC35         {
PC36             . . .
PC37         }
PC38         private KeyVaultClient _keyVaultClient = new KeyVaultClient(" . . .");
PC39     }

```

Database.cs -

```

DB01 public class Database
DB02 {
DB03     private string ConnectionString =
DB04
DB05     public async Task<object> LoadUserDetails(string userId)
DB06     {
DB07
DB08         return await policy.ExecuteAsync(async () =>
DB09         {
DB10             using (var connection = new SqlConnection(ConnectionString))
DB11             {
DB12                 await connection.OpenAsync();
DB13                 using (var command = new SqlCommand(" . . .", connection))
DB14                     using (var reader = command.ExecuteReader())
DB15                     {
DB16                         . . .
DB17                     }
DB18                 }
DB19             });
DB20         }
DB21     }

```

ReceiptUploader.cs -

```

RU01 public class ReceiptUploader
RU02 {
RU03     public async Task UploadFile(string file, byte[] binary)
RU04     {
RU05         var httpClient = new HttpClient();
RU06         var response = await httpClient.PutAsync("...", new ByteArrayContent(binary));
RU07         while (ShouldRetry(response))
RU08         {
RU09             response = await httpClient.PutAsync("...", new ByteArrayContent(binary));
RU10         }
RU11     }
RU12     private bool ShouldRetry(HttpStatusCode response)
RU13     {
RU14     }
RU15 }
RU16 }
```

ConfigureSSE.ps1 -

```

CS01 $storageAccount = Get-AzureRmStorageAccount -ResourceGroupName "..." -AccountName "..."
CS02 $keyVault = Get-AzureRmKeyVault -VaultName "..."
CS03 $key = Get-AzureKeyVaultKey -VaultName $keyVault.VaultName -Name "..."
CS04 Set-AzureRmKeyVaultAccessPolicy ` 
CS05     -VaultName $keyVault.VaultName ` 
CS06     -ObjectId $storageAccount.Identity.PrincipalId ` 
CS07
CS08
CS09 Set-AzureRmStorageAccount ` 
CS10     -ResourceGroupName $storageAccount.ResourceGroupName ` 
CS11     -AccountName $storageAccount.StorageAccountName ` 
CS12     -EnableEncryptionService File ` 
CS13     -KeyvaultEncryption ` 
CS14     -KeyName $key.Name 
CS15     -KeyVersion $key.Version ` 
CS16     -KeyVaultUri $keyVault.VaultUri
```

### Question

You need to resolve the capacity issue.

What should you do?

- A. Convert the trigger on the Azure Function to an Azure Blob storage trigger
- B. Ensure that the consumption plan is configured correctly to allow scaling
- C. Move the Azure Function to a dedicated App Service Plan
- D. Update the loop starting on line PC09 to process items in parallel

**Correct Answer: D**

If you want to read the files in parallel, you cannot use forEach. Each of the async callback function calls does return a promise. You can await the array of promises that you'll get with Promise.all.

Scenario: Capacity issue: During busy periods, employees report long delays between the time they upload the receipt and when it appears in the web application.

```
PC08     var container = await GetCloudBlobContainer();
PC09     foreach (var fileItem in await ListFiles())
PC10     {
PC11         var file = new CloudFile(fileItem.StorageUri.PrimaryUri);
PC12         var ms = new MemoryStream();
PC13         await file.DownloadToStreamAsync(ms);
PC14         var blob = container.GetBlockBlobReference(fileItem.Uri.ToString());
PC15         await blob.UploadFromStreamAsync(ms);
PC16     }
PC17 }
```

Reference:

<https://stackoverflow.com/questions/37576685/using-async-await-with-a-foreach-loop>

Question #3

## Introductory Info

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 questions in 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 section 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 in 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 SaaS company that provides a solution for managing employee expenses. The solution consists of an ASP.NET Core Web API project that is deployed as an Azure Web App.

Overall architecture -

Employees upload receipts for the system to process. When processing is complete, the employee receives a summary report email that details the processing results. Employees then use a web application to manage their receipts and perform any additional tasks needed for reimbursement.

Receipt processing -

Employees may upload receipts in two ways:

Uploading using an Azure Files mounted folder

Uploading using the web application

Data Storage -

Receipt and employee information is stored in an Azure SQL database.

Documentation -

Employees are provided with a getting started document when they first use the solution. The documentation includes details on supported operating systems for

Azure File upload, and instructions on how to configure the mounted folder.

Solution details -

Users table -

Column	Description
UserId	unique identifier for an employee
ExpenseAccount	employees expense account number in the format 1234-123-1234
AllowedAmount	limit of allowed expenses before approval is needed
SupervisorId	unique identifier for employee's supervisor
SecurityPin	value used to validate user identity

Web Application -

You enable MSI for the Web App and configure the Web App to use the security principal name WebAppIdentity.

Processing -

Processing is performed by an Azure Function that uses version 2 of the Azure Function runtime. Once processing is completed, results are stored in Azure Blob

Storage and an Azure SQL database. Then, an email summary is sent to the user with a link to the processing report. The link to the report must remain valid if the email is forwarded to another user.

#### Logging -

Azure Application Insights is used for telemetry and logging in both the processor and the web application. The processor also has TraceWriter logging enabled.

Application Insights must always contain all log messages.

#### Requirements -

##### Receipt processing -

Concurrent processing of a receipt must be prevented.

##### Disaster recovery -

Regional outage must not impact application availability. All DR operations must not be dependent on application running and must ensure that data in the DR region is up to date.

#### Security -

User's SecurityPin must be stored in such a way that access to the database does not allow the viewing of SecurityPins. The web application is the only system that should have access to SecurityPins.

All certificates and secrets used to secure data must be stored in Azure Key Vault.

You must adhere to the principle of least privilege and provide privileges which are essential to perform the intended function.

All access to Azure Storage and Azure SQL database must use the application's Managed Service Identity (MSI).

Receipt data must always be encrypted at rest.

All data must be protected in transit.

User's expense account number must be visible only to logged in users. All other views of the expense account number should include only the last segment, with the remaining parts obscured.

In the case of a security breach, access to all summary reports must be revoked without impacting other parts of the system.

#### Issues -

##### Upload format issue -

Employees occasionally report an issue with uploading a receipt using the web application. They report that when they upload a receipt using the Azure File

Share, the receipt does not appear in their profile. When this occurs, they delete the file in the file share and use the web application, which returns a 500 Internal

Server error page.

##### Capacity issue -

During busy periods, employees report long delays between the time they upload the receipt and when it appears in the web application.

##### Log capacity issue -

Developers report that the number of log messages in the trace output for the processor is too high, resulting in lost log messages.

#### Application code -

##### Processing.cs -

```

PC01 public static class Processing
PC02 {
PC03     public static class Function
PC04     {
PC05         [FunctionName("IssueWork")]
PC06         public static async Task Run([TimerTrigger("0 */5 * * *")] TimerInfo timer, ILogger
log)
PC07         {
PC08             var container = await GetCloudBlobContainer();
PC09             foreach (var fileItem in await ListFiles())
PC10             {
PC11                 var file = new CloudFile(fileItem.StorageUri.PrimaryUri);
PC12                 var ms = new MemoryStream();
PC13                 await file.DownloadToStreamAsync(ms);
PC14                 var blob = container.GetBlockBlobReference(fileItem.Uri.ToString());
PC15                 await blob.UploadFromStreamAsync(ms);
PC16
PC17             }
PC18         }
PC19         private static CloudBlockBlob GetDRBlock(CloudBlockBlob sourceBlob)
PC20         {
PC21             . . .
PC22         }
PC23         private static async Task<CloudBlobContainer> GetCloudBlobContainer()
PC24         {
PC25             var cloudBlobClient = new CloudBlobClient(new Uri(" . . ."), await GetCredentials());
PC26
PC27             await cloudBlobClient.GetRootContainerReference().CreateIfNotExistsAsync();
PC28             return cloudBlobClient.GetRootContainerReference();
PC29         }
PC30         private static async Task<StorageCredentials> GetCredentials()
PC31         {
PC32             . . .
PC33         }
PC34         private static async Task<List<IListFileItem>> ListFiles()
PC35         {
PC36             . . .
PC37         }
PC38         private KeyVaultClient _keyVaultClient = new KeyVaultClient(" . . .");
PC39     }

```

Database.cs -

```

DB01 public class Database
DB02 {
DB03     private string ConnectionString =
DB04
DB05     public async Task<object> LoadUserDetails(string userId)
DB06     {
DB07
DB08         return await policy.ExecuteAsync(async () =>
DB09         {
DB10             using (var connection = new SqlConnection(ConnectionString))
DB11             {
DB12                 await connection.OpenAsync();
DB13                 using (var command = new SqlCommand(" . . .", connection))
DB14                     using (var reader = command.ExecuteReader())
DB15                     {
DB16                         . . .
DB17                     }
DB18                 }
DB19             });
DB20         }
DB21     }

```

ReceiptUploader.cs -

```

RU01 public class ReceiptUploader
RU02 {
RU03     public async Task UploadFile(string file, byte[] binary)
RU04     {
RU05         var httpClient = new HttpClient();
RU06         var response = await httpClient.PutAsync("...", new ByteArrayContent(binary));
RU07         while (ShouldRetry(response))
RU08         {
RU09             response = await httpClient.PutAsync("...", new ByteArrayContent(binary));
RU10         }
RU11     }
RU12     private bool ShouldRetry(HttpStatusCode response)
RU13     {
RU14     }
RU15 }
RU16 }
```

ConfigureSSE.ps1 -

```

CS01 $storageAccount = Get-AzureRmStorageAccount -ResourceGroupName "..." -AccountName "..."
CS02 $keyVault = Get-AzureRmKeyVault -VaultName "..."
CS03 $key = Get-AzureKeyVaultKey -VaultName $keyVault.VaultName -Name "..."
CS04 Set-AzureRmKeyVaultAccessPolicy ` 
CS05     -VaultName $keyVault.VaultName ` 
CS06     -ObjectId $storageAccount.Identity.PrincipalId ` 
CS07
CS08
CS09 Set-AzureRmStorageAccount ` 
CS10     -ResourceGroupName $storageAccount.ResourceGroupName ` 
CS11     -AccountName $storageAccount.StorageAccountName ` 
CS12     -EnableEncryptionService File ` 
CS13     -KeyvaultEncryption ` 
CS14     -KeyName $key.Name 
CS15     -KeyVersion $key.Version ` 
CS16     -KeyVaultUri $keyVault.VaultUri
```

### Question

You need to resolve the log capacity issue.

What should you do?

- A. Create an Application Insights Telemetry Filter
- B. Change the minimum log level in the host.json file for the function
- C. Implement Application Insights Sampling
- D. Set a LogCategoryFilter during startup

**Correct Answer:** C 

Scenario, the log capacity issue: Developers report that the number of log message in the trace output for the processor is too high, resulting in lost log messages.

Sampling is a feature in Azure Application Insights. It is the recommended way to reduce telemetry traffic and storage, while preserving a statistically correct analysis of application data. The filter selects items that are related, so that you can navigate between items when you are doing diagnostic investigations. When metric counts are presented to you in the portal, they are renormalized to take account of the sampling, to minimize any effect on the statistics.

Sampling reduces traffic and data costs, and helps you avoid throttling.

Reference:

Topic 16 - Testlet 18

Question #1

## Introductory Info

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

LabelMaker app -

Coho Winery produces, bottles, and distributes a variety of wines globally. You are a developer implementing highly scalable and resilient applications to support online order processing by using Azure solutions.

Coho Winery has a LabelMaker application that prints labels for wine bottles. The application sends data to several printers. The application consists of five modules that run independently on virtual machines (VMs). Coho Winery plans to move the application to Azure and continue to support label creation.

External partners send data to the LabelMaker application to include artwork and text for custom label designs.

Requirements. Data -

You identify the following requirements for data management and manipulation:

Order data is stored as nonrelational JSON and must be queried using SQL.

Changes to the Order data must reflect immediately across all partitions. All reads to the Order data must fetch the most recent writes.

Requirements. Security -

You have the following security requirements:

Users of Coho Winery applications must be able to provide access to documents, resources, and applications to external partners.

External partners must use their own credentials and authenticate with their organization's identity management solution.

External partner logins must be audited monthly for application use by a user account administrator to maintain company compliance.

Storage of e-commerce application settings must be maintained in Azure Key Vault.

E-commerce application sign-ins must be secured by using Azure App Service authentication and Azure Active Directory (AAD).

Conditional access policies must be applied at the application level to protect company content.

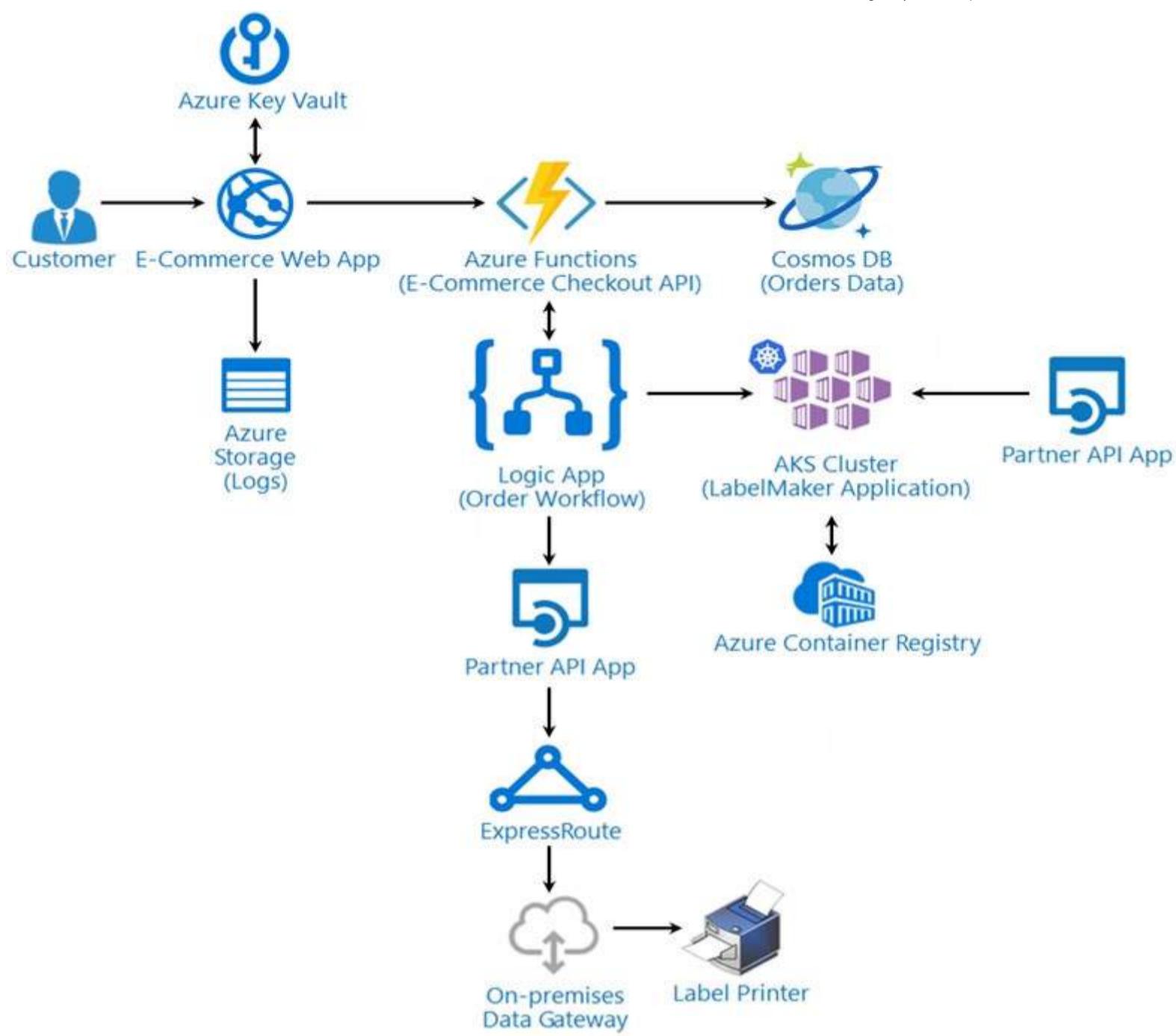
The LabelMaker application must be secured by using an AAD account that has full access to all namespaces of the Azure Kubernetes Service (AKS) cluster.

Requirements. LabelMaker app -

Azure Monitor Container Health must be used to monitor the performance of workloads that are deployed to Kubernetes environments and hosted on Azure Kubernetes Service (AKS).

You must use Azure Container Registry to publish images that support the AKS deployment.

Architecture -

**Issues -**

Calls to the Printer API App fail periodically due to printer communication timeouts.

Printer communication timeouts occur after 10 seconds. The label printer must only receive up to 5 attempts within one minute.

The order workflow fails to run upon initial deployment to Azure.

**Order.json -**

Relevant portions of the app files are shown below. Line numbers are included for reference only.

This JSON file contains a representation of the data for an order that includes a single item.

### Order.json

```
01 {  
02   "id" : 1,  
03   "customers" : [  
04     {  
05       "familyName" : "Doe",  
06       "givenName" : "John",  
07       "customerid" : 5  
08     }  
09   ],  
10   "line_items" : [  
11     {  
12       "fulfillable_quantity" : 1,  
13       "id": 6,  
14       "price" : "199.99" ,  
15       "product_id" : 7513594,  
16       "quantity": 1,  
17       "requires_shipping" : true ,  
18       "sku": "SFC-342-N" ,  
19       "title" : "Surface Go" ,  
20       "vendor" : "Microsoft" ,  
21       "name" : "Surface Go - 8GB" ,  
22       "taxable" : true ,  
23       "tax_lines" : [  
24         {  
25           "title" : "State Tax" ,  
26           "price" : "3.98" ,  
27           "rate" : 0.06  
28         }  
29       ].
```

#### Question

You need to troubleshoot the order workflow.

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

NOTE: Each correct selection is worth one point.

- A. Review the API connections.
- B. Review the activity log.
- C. Review the run history.
- D. Review the trigger history.

#### Correct Answer: CD

Scenario: The order workflow fails to run upon initial deployment to Azure.

Check runs history: Each time that the trigger fires for an item or event, the Logic Apps engine creates and runs a separate workflow instance for each item or event. If a run fails, follow these steps to review what happened during that run, including the status for each step in the workflow plus the inputs and outputs for each step.

Check the workflow's run status by checking the runs history. To view more information about a failed run, including all the steps in that run in their status, select the failed run.

Example:

**Runs history**  
DataOperationsLogicApp

Refresh

All Start time earlier than  
Pick a date Pick a time

Search to filter items by identifier

Status	Start time	Identifi...	Duration	Static Results
Failed	9/23/2019, 7:08 PM	085863...	640 Milliseconds	
Failed	9/23/2019, 7:08 PM	085863...	1.55 Seconds	

Check the trigger's status by checking the trigger history

To view more information about the trigger attempt, select that trigger event, for example:

**Recurrence**  
DataOperationsLogicApp

All Pick a date Pick a time

Recurrence

Status	Start time	Fired
Skipped	11/4/2019, 9:07 AM	
Failed	11/4/2019, 9:00 AM	
Succeeded	11/3/2019, 6:23 PM	Fired
Succeeded	11/3/2019, 6:23 PM	Fired

Reference:

<https://docs.microsoft.com/en-us/azure/logic-apps/logic-apps-diagnosing-failures>

Question #2

## Introductory Info

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

LabelMaker app -

Coho Winery produces, bottles, and distributes a variety of wines globally. You are a developer implementing highly scalable and resilient applications to support online order processing by using Azure solutions.

Coho Winery has a LabelMaker application that prints labels for wine bottles. The application sends data to several printers. The application consists of five modules that run independently on virtual machines (VMs). Coho Winery plans to move the application to Azure and continue to support label creation.

External partners send data to the LabelMaker application to include artwork and text for custom label designs.

Requirements. Data -

You identify the following requirements for data management and manipulation:

Order data is stored as nonrelational JSON and must be queried using SQL.

Changes to the Order data must reflect immediately across all partitions. All reads to the Order data must fetch the most recent writes.

Requirements. Security -

You have the following security requirements:

Users of Coho Winery applications must be able to provide access to documents, resources, and applications to external partners.

External partners must use their own credentials and authenticate with their organization's identity management solution.

External partner logins must be audited monthly for application use by a user account administrator to maintain company compliance.

Storage of e-commerce application settings must be maintained in Azure Key Vault.

E-commerce application sign-ins must be secured by using Azure App Service authentication and Azure Active Directory (AAD).

Conditional access policies must be applied at the application level to protect company content.

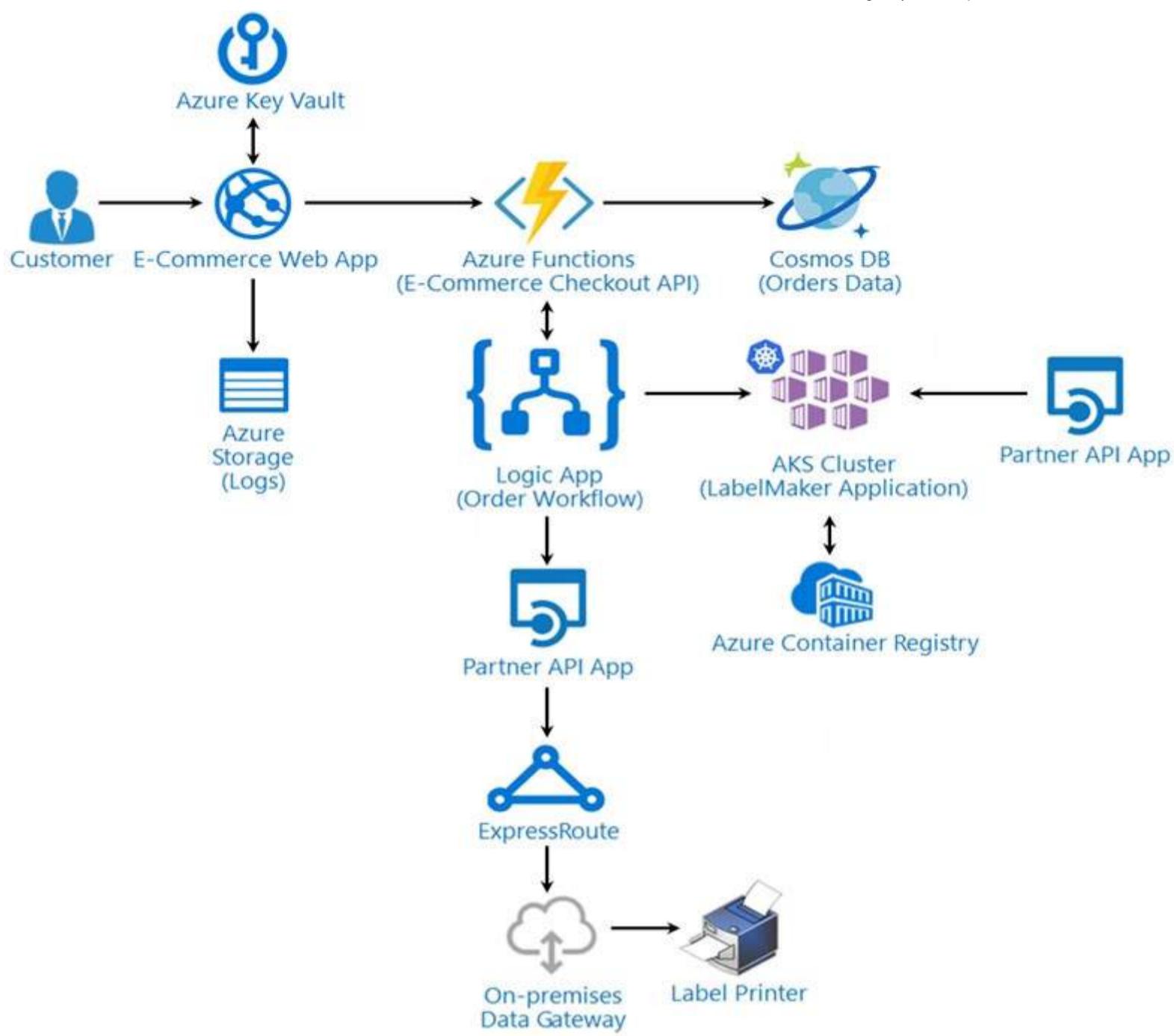
The LabelMaker application must be secured by using an AAD account that has full access to all namespaces of the Azure Kubernetes Service (AKS) cluster.

Requirements. LabelMaker app -

Azure Monitor Container Health must be used to monitor the performance of workloads that are deployed to Kubernetes environments and hosted on Azure Kubernetes Service (AKS).

You must use Azure Container Registry to publish images that support the AKS deployment.

Architecture -

**Issues -**

Calls to the Printer API App fail periodically due to printer communication timeouts.

Printer communication timeouts occur after 10 seconds. The label printer must only receive up to 5 attempts within one minute.

The order workflow fails to run upon initial deployment to Azure.

**Order.json -**

Relevant portions of the app files are shown below. Line numbers are included for reference only.

This JSON file contains a representation of the data for an order that includes a single item.

### Order.json

```
01 {  
02   "id" : 1,  
03   "customers" : [  
04     {  
05       "familyName" : "Doe",  
06       "givenName" : "John",  
07       "customerid" : 5  
08     }  
09   ],  
10   "line_items" : [  
11     {  
12       "fulfillable_quantity" : 1,  
13       "id": 6,  
14       "price" : "199.99" ,  
15       "product_id" : 7513594,  
16       "quantity": 1,  
17       "requires_shipping" : true ,  
18       "sku": "SFC-342-N" ,  
19       "title" : "Surface Go" ,  
20       "vendor" : "Microsoft" ,  
21       "name" : "Surface Go - 8GB" ,  
22       "taxable" : true ,  
23       "tax_lines" : [  
24         {  
25           "title" : "State Tax" ,  
26           "price" : "3.98" ,  
27           "rate" : 0.06  
28         }  
29       ].
```

#### Question

HOTSPOT -

You need to update the order workflow to address the issue when calling the Printer API App.

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

NOTE: Each correct selection is worth one point.

Hot Area:

## Answer Area

```
"print_label": {  
    "type": "Http",  
    "inputs": {  
        "method": "POST",  
        "uri": "https://www.cohowinery.com/printer/printlabel",  
        "retryPolicy": {  
            "type": "",  
            "options": ["default", "none", "fixed", "exponential"]  
        },  
        "interval": "",  
        "options": ["PT10S", "PT30S", "PT60S", "PT1D"]  
    },  
    "count": "",  
    "options": [5, 10, 60]  
},  
}  
}
```

Correct Answer:

## Answer Area

```
"print_label": {  
    "type": "Http",  
    "inputs": {  
        "method": "POST",  
        "uri": "https://www.cohowinery.com/printer/printlabel",  
        "retryPolicy": {  
            "type": "fixed",  
            "interval": "PT60S",  
            "count": 5  
        }  
    }  
}
```

Topic 17 - Testlet 19

```
"print_label": {  
    "type": "Http",  
    "inputs": {  
        "method": "POST",  
        "uri": "https://www.cohowinery.com/printer/printlabel",  
        "retryPolicy": {  
            "type": "fixed",  
            "interval": "PT60S",  
            "count": 5  
        }  
    }  
}
```

Box 1: fixed -

The “Default” policy does 4 exponential retries and from my experience the interval times are often too short in situations.

Box 2: PT60S -

We could set a fixed interval, e.g. 5 retries every 60 seconds (PT60S).

PT60S is 60 seconds.

Scenario: Calls to the Printer API App fail periodically due to printer communication timeouts.

Printer communication timeouts occur after 10 seconds. The label printer must only receive up to 5 attempts within one minute.

Box 3: 5 -

Reference:

<https://michalsacewicz.com/error-handling-in-power-automate/>

Question #1

**Introductory Info**

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Current environment -

Windows Server 2016 virtual machine

This virtual machine (VM) runs BizTalk Server 2016. The VM runs the following workflows:

Ocean Transport `" This workflow gathers and validates container information including container contents and arrival notices at various shipping ports.

Inland Transport `" This workflow gathers and validates trucking information including fuel usage, number of stops, and routes.

The VM supports the following REST API calls:

Container API `" This API provides container information including weight, contents, and other attributes.

Location API `" This API provides location information regarding shipping ports of call and trucking stops.

Shipping REST API `" This API provides shipping information for use and display on the shipping website.

Shipping Data -

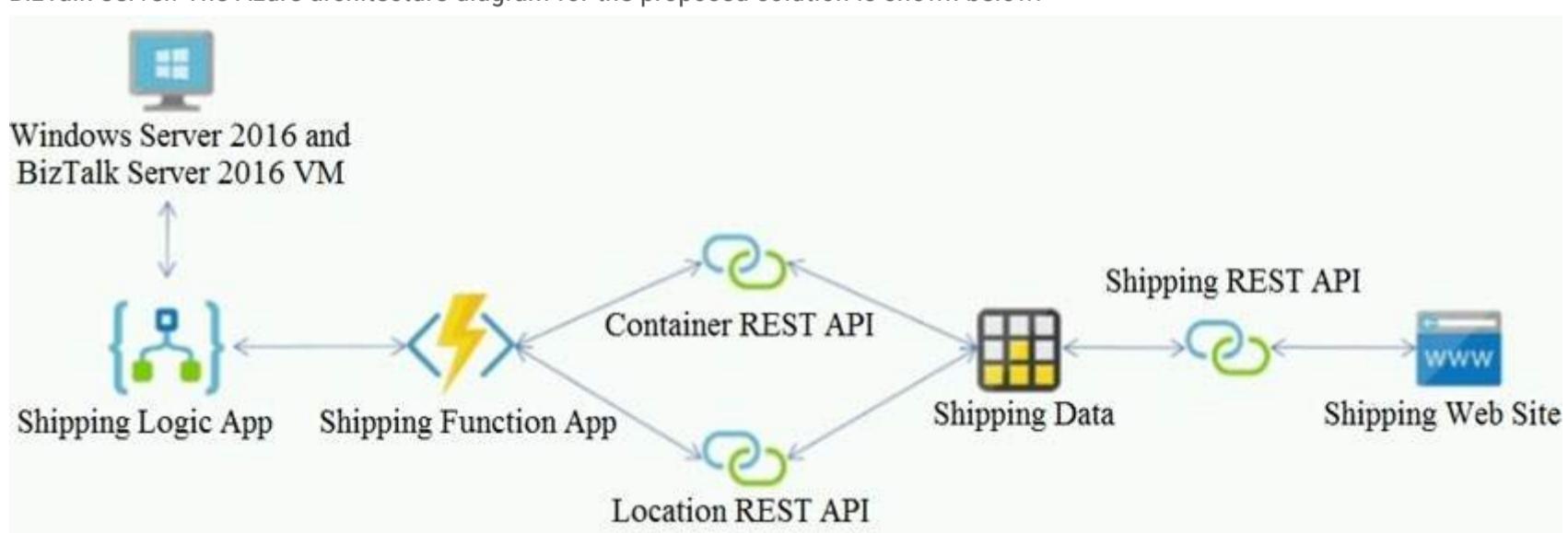
The application uses MongoDB JSON document storage database for all container and transport information.

Shipping Web Site -

The site displays shipping container tracking information and container contents. The site is located at <http://shipping.wideworldimporters.com/>

Proposed solution -

The on-premises shipping application must be moved to Azure. The VM has been migrated to a new Standard\_D16s\_v3 Azure VM by using Azure Site Recovery and must remain running in Azure to complete the BizTalk component migrations. You create a Standard\_D16s\_v3 Azure VM to host BizTalk Server. The Azure architecture diagram for the proposed solution is shown below:



Requirements -

Shipping Logic app -

The Shipping Logic app must meet the following requirements:

Support the ocean transport and inland transport workflows by using a Logic App.

Support industry-standard protocol X12 message format for various messages including vessel content details and arrival notices.

Secure resources to the corporate VNet and use dedicated storage resources with a fixed costing model.

Maintain on-premises connectivity to support legacy applications and final BizTalk migrations.

Shipping Function app -

Implement secure function endpoints by using app-level security and include Azure Active Directory (Azure AD).

REST APIs -

The REST API's that support the solution must meet the following requirements:

Secure resources to the corporate VNet.

- Allow deployment to a testing location within Azure while not incurring additional costs.

- Automatically scale to double capacity during peak shipping times while not causing application downtime.

- Minimize costs when selecting an Azure payment model.

Shipping data -

Data migration from on-premises to Azure must minimize costs and downtime.

Shipping website -

Use Azure Content Delivery Network (CDN) and ensure maximum performance for dynamic content while minimizing latency and costs.

Issues -

Windows Server 2016 VM -

The VM shows high network latency, jitter, and high CPU utilization. The VM is critical and has not been backed up in the past. The VM must enable a quick restore from a 7-day snapshot to include in-place restore of disks in case of failure.

Shipping website and REST APIs -

The following error message displays while you are testing the website:

Failed to load http://test-shippingapi.wideworldimporters.com/: No 'Access-Control-Allow-Origin' header is present on the requested resource. Origin 'http://test.wideworldimporters.com/' is therefore not allowed access.

### Question

DRAG DROP -

You need to support the message processing for the ocean transport workflow.

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

Link the Logic App to the integration account.

Add partners, schemas, certificates, maps, and agreements.

Update the Logic App to use the partners, schemas, certificates, maps, and agreements.

Create a custom connector for the Logic App.

Link the custom connector to the Logic App.

Create an integration account in the Azure portal.



	Actions	Answer Area
Correct Answer:	Link the Logic App to the integration account.	Create an integration account in the Azure portal.
	Add partners, schemas, certificates, maps, and agreements.	Link the Logic App to the integration account.
	Update the Logic App to use the partners, schemas, certificates, maps, and agreements.	Add partners, schemas, certificates, maps, and agreements.
	Create a custom connector for the Logic App.	Create a custom connector for the Logic App.
	Link the custom connector to the Logic App.	
	Create an integration account in the Azure portal.	

Step 1: Create an integration account in the Azure portal  
You can define custom metadata for artifacts in integration accounts and get that metadata during runtime for your logic app to use. For example, you can provide metadata for artifacts, such as partners, agreements, schemas, and maps - all store metadata using key-value pairs.  
Step 2: Link the Logic App to the integration account  
A logic app that's linked to the integration account and artifact metadata you want to use.  
Step 3: Add partners, schemas, certificates, maps, and agreements  
Step 4: Create a custom connector for the Logic App.  
Reference:  
<https://docs.microsoft.com/bs-latn-ba/azure/logic-apps/logic-apps-enterprise-integration-metadata>

Question #2

**Introductory Info**

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Current environment -

Windows Server 2016 virtual machine

This virtual machine (VM) runs BizTalk Server 2016. The VM runs the following workflows:

Ocean Transport `" This workflow gathers and validates container information including container contents and arrival notices at various shipping ports.

Inland Transport `" This workflow gathers and validates trucking information including fuel usage, number of stops, and routes.

The VM supports the following REST API calls:

Container API `" This API provides container information including weight, contents, and other attributes.

Location API `" This API provides location information regarding shipping ports of call and trucking stops.

Shipping REST API `" This API provides shipping information for use and display on the shipping website.

Shipping Data -

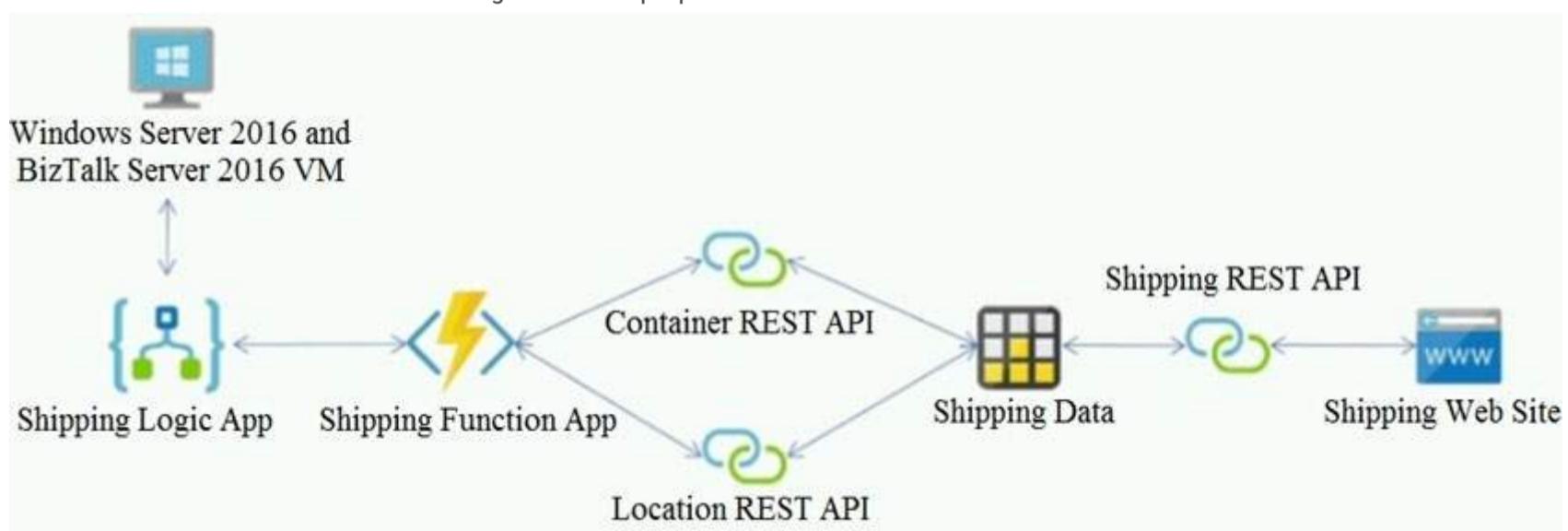
The application uses MongoDB JSON document storage database for all container and transport information.

Shipping Web Site -

The site displays shipping container tracking information and container contents. The site is located at <http://shipping.wideworldimporters.com/>

Proposed solution -

The on-premises shipping application must be moved to Azure. The VM has been migrated to a new Standard\_D16s\_v3 Azure VM by using Azure Site Recovery and must remain running in Azure to complete the BizTalk component migrations. You create a Standard\_D16s\_v3 Azure VM to host BizTalk Server. The Azure architecture diagram for the proposed solution is shown below:



Requirements -

Shipping Logic app -

The Shipping Logic app must meet the following requirements:

Support the ocean transport and inland transport workflows by using a Logic App.

Support industry-standard protocol X12 message format for various messages including vessel content details and arrival notices.

Secure resources to the corporate VNet and use dedicated storage resources with a fixed costing model.

Maintain on-premises connectivity to support legacy applications and final BizTalk migrations.

#### Shipping Function app -

Implement secure function endpoints by using app-level security and include Azure Active Directory (Azure AD).

#### REST APIs -

The REST API's that support the solution must meet the following requirements:

Secure resources to the corporate VNet.

- Allow deployment to a testing location within Azure while not incurring additional costs.

Automatically scale to double capacity during peak shipping times while not causing application downtime.

Minimize costs when selecting an Azure payment model.

#### Shipping data -

Data migration from on-premises to Azure must minimize costs and downtime.

#### Shipping website -

Use Azure Content Delivery Network (CDN) and ensure maximum performance for dynamic content while minimizing latency and costs.

#### Issues -

##### Windows Server 2016 VM -

The VM shows high network latency, jitter, and high CPU utilization. The VM is critical and has not been backed up in the past. The VM must enable a quick restore from a 7-day snapshot to include in-place restore of disks in case of failure.

##### Shipping website and REST APIs -

The following error message displays while you are testing the website:

Failed to load http://test-shippingapi.wideworldimporters.com/: No 'Access-Control-Allow-Origin' header is present on the requested resource. Origin 'http://test.wideworldimporters.com/' is therefore not allowed access.

#### Question

You need to support the requirements for the Shipping Logic App.

What should you use?

- A. Azure Active Directory Application Proxy
- B. Site-to-Site (S2S) VPN connection
- C. On-premises Data Gateway
- D. Point-to-Site (P2S) VPN connection

#### Correct Answer: C

Before you can connect to on-premises data sources from Azure Logic Apps, download and install the on-premises data gateway on a local computer. The gateway works as a bridge that provides quick data transfer and encryption between data sources on premises (not in the cloud) and your logic apps.

The gateway supports BizTalk Server 2016.

Note: Microsoft have now fully incorporated the Azure BizTalk Services capabilities into Logic Apps and Azure App Service Hybrid Connections.

Logic Apps Enterprise Integration pack bring some of the enterprise B2B capabilities like AS2 and X12, EDI standards support

Scenario: The Shipping Logic app must meet the following requirements:

- Support the ocean transport and inland transport workflows by using a Logic App.
- Support industry-standard protocol X12 message format for various messages including vessel content details and arrival notices.
- Secure resources to the corporate VNet and use dedicated storage resources with a fixed costing model.
- Maintain on-premises connectivity to support legacy applications and final BizTalk migrations.

Reference:

<https://docs.microsoft.com/en-us/azure/logic-apps/logic-apps-gateway-install>

**Topic 18 - Testlet 2**

Question #1

**Introductory Info**

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

City Power & Light company provides electrical infrastructure monitoring solutions for homes and businesses. The company is migrating solutions to Azure.

Current environment -

Architecture overview -

The company has a public website located at <http://www.cpndl.com/>. The site is a single-page web application that runs in Azure App Service on Linux. The website uses files stored in Azure Storage and cached in Azure Content Delivery Network (CDN) to serve static content.

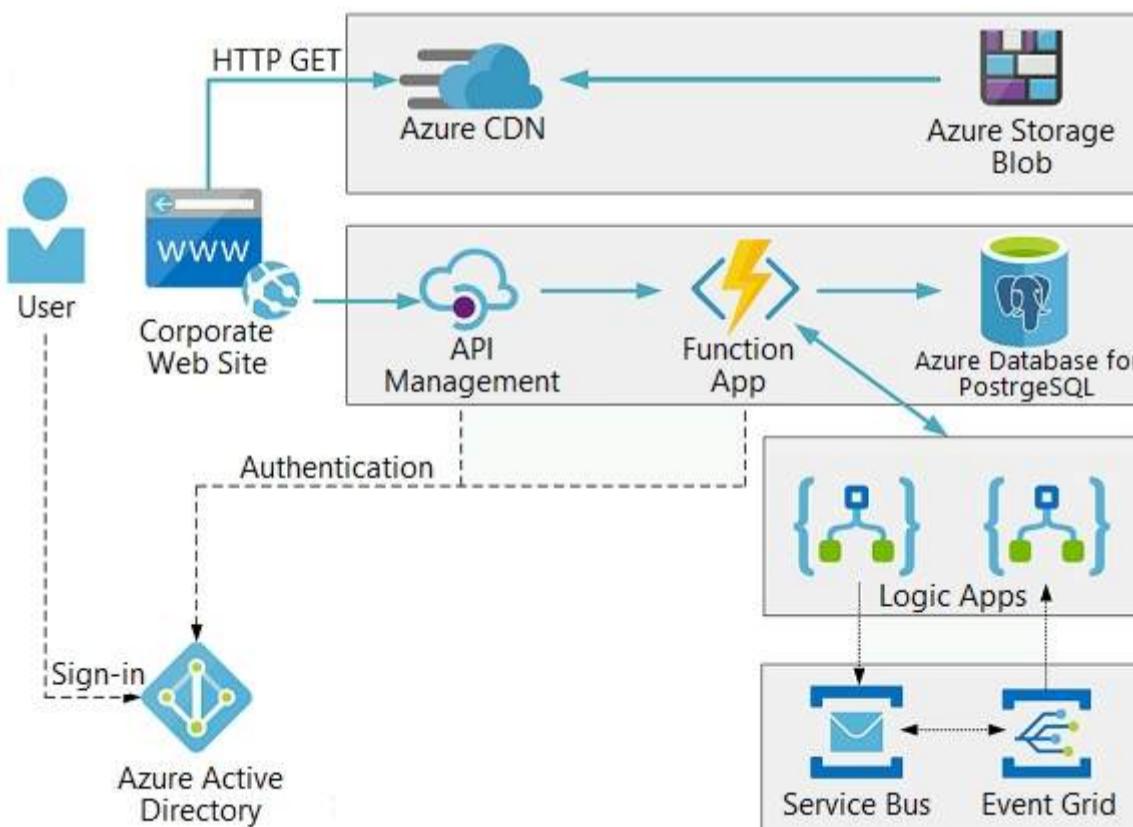
API Management and Azure Function App functions are used to process and store data in Azure Database for PostgreSQL. API Management is used to broker communications to the Azure Function app functions for Logic app integration. Logic apps are used to orchestrate the data processing while Service Bus and

Event Grid handle messaging and events.

The solution uses Application Insights, Azure Monitor, and Azure Key Vault.

Architecture diagram -

The company has several applications and services that support their business. The company plans to implement serverless computing where possible. The overall architecture is shown below.



User authentication -

The following steps detail the user authentication process:

1. The user selects Sign in in the website.
2. The browser redirects the user to the Azure Active Directory (Azure AD) sign in page.

3. The user signs in.
4. Azure AD redirects the user's session back to the web application. The URL includes an access token.
5. The web application calls an API and includes the access token in the authentication header. The application ID is sent as the audience ('`aud`') claim in the access token.
6. The back-end API validates the access token.

Requirements -

Corporate website -

Communications and content must be secured by using SSL.

Communications must use HTTPS.

Data must be replicated to a secondary region and three availability zones.

Data storage costs must be minimized.

Azure Database for PostgreSQL -

The database connection string is stored in Azure Key Vault with the following attributes:

Azure Key Vault name: cpandlkeyvault

Secret name: PostgreSQLConn

Id: 80df3e46ffcd4f1cb187f79905e9a1e8

The connection information is updated frequently. The application must always use the latest information to connect to the database.

Azure Service Bus and Azure Event Grid

Azure Event Grid must use Azure Service Bus for queue-based load leveling.

Events in Azure Event Grid must be routed directly to Service Bus queues for use in buffering.

Events from Azure Service Bus and other Azure services must continue to be routed to Azure Event Grid for processing.

Security -

All SSL certificates and credentials must be stored in Azure Key Vault.

File access must restrict access by IP, protocol, and Azure AD rights.

All user accounts and processes must receive only those privileges which are essential to perform their intended function.

Compliance -

Auditing of the file updates and transfers must be enabled to comply with General Data Protection Regulation (GDPR). The file updates must be read-only, stored in the order in which they occurred, include only create, update, delete, and copy operations, and be retained for compliance reasons.

Issues -

Corporate website -

While testing the site, the following error message displays:

CryptographicException: The system cannot find the file specified.

Function app -

You perform local testing for the RequestUserApproval function. The following error message displays:

'Timeout value of 00:10:00 exceeded by function: RequestUserApproval'

The same error message displays when you test the function in an Azure development environment when you run the following Kusto query:

FunctionAppLogs -

```
| where FunctionName == "RequestUserApproval"
```

Logic app -

You test the Logic app in a development environment. The following error message displays:

'400 Bad Request'

Troubleshooting of the error shows an HttpTrigger action to call the RequestUserApproval function.

Code -

Corporate website -

Security.cs:

```

SC01 public class Security
SC02 {
SC03 var bytes = System.IO.File.ReadAllBytes("~/var/ssl/private");
SC04 var cert = new System.Security.Cryptography.X509Certificate2(bytes);
SC05 var certName = cert.FriendlyName;
SC06 }

```

Function app -

RequestUserApproval.cs:

```

RA01 public static class RequestUserApproval
RA02 {
RA03 [FunctionName("RequestUserApproval")]
RA04 public static async Task<IActionResult> Run(
RA05 [HttpTrigger(AuthorizationLevel.Function, "get", "post", Route = null)] HttpRequest req,
RA06 ILogger log)
RA06 {
RA07     log.LogInformation("RequestUserApproval function processed a request.");
RA08 ...
RA09 return ProcessRequest(req)
RA10     ? (ActionResult)new OkObjectResult($"User approval processed")
RA11     : new BadRequestObjectResult("Failed to process user approval");
RA12 }
RA13 private static bool ProcessRequest(HttpRequest req)
RA14 {
RA15     ...
RA16 }
RA17 }

```

### Question

You need to correct the RequestUserApproval Function app error.

What should you do?

- A. Update line RA13 to use the `async` keyword and return an `HttpRequest` object value.
- B. Configure the Function app to use an App Service hosting plan. Enable the Always On setting of the hosting plan.
- C. Update the function to be stateful by using Durable Functions to process the request payload.
- D. Update the `functionTimeout` property of the `host.json` project file to 15 minutes.

**Correct Answer:** C 

Async operation tracking -

The HTTP response mentioned previously is designed to help implement long-running HTTP `async` APIs with Durable Functions. This pattern is sometimes referred to as the polling consumer pattern.

Both the client and server implementations of this pattern are built into the Durable Functions HTTP APIs.

Function app -

You perform local testing for the RequestUserApproval function. The following error message displays:

'Timeout value of 00:10:00 exceeded by function: RequestUserApproval'

The same error message displays when you test the function in an Azure development environment when you run the following Kusto query:

FunctionAppLogs -

```
| where FunctionName == "RequestUserApproval"
```

Reference:

<https://docs.microsoft.com/en-us/azure/azure-functions/durable/durable-functions-http-features>

Topic 19 - Testlet 20

Question #1

**Introductory Info**

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

City Power & Light company provides electrical infrastructure monitoring solutions for homes and businesses. The company is migrating solutions to Azure.

Current environment -

Architecture overview -

The company has a public website located at <http://www.cpndl.com/>. The site is a single-page web application that runs in Azure App Service on Linux. The website uses files stored in Azure Storage and cached in Azure Content Delivery Network (CDN) to serve static content.

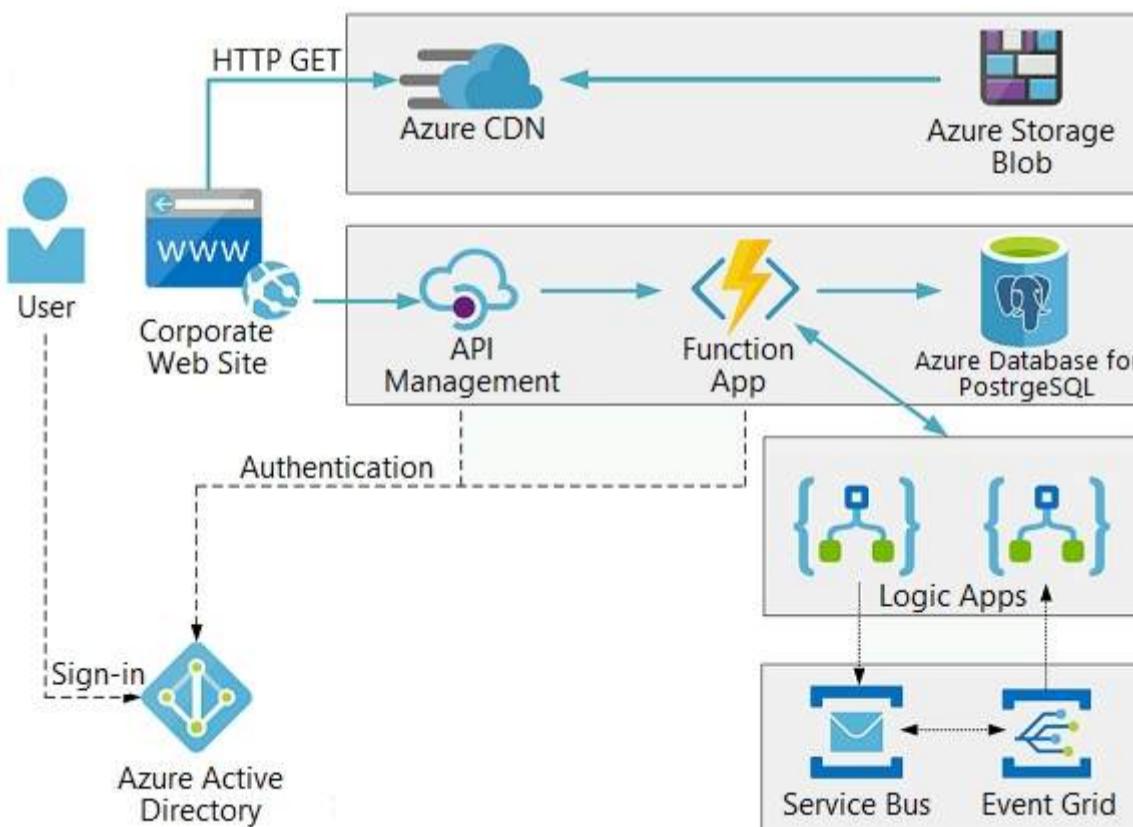
API Management and Azure Function App functions are used to process and store data in Azure Database for PostgreSQL. API Management is used to broker communications to the Azure Function app functions for Logic app integration. Logic apps are used to orchestrate the data processing while Service Bus and

Event Grid handle messaging and events.

The solution uses Application Insights, Azure Monitor, and Azure Key Vault.

Architecture diagram -

The company has several applications and services that support their business. The company plans to implement serverless computing where possible. The overall architecture is shown below.



User authentication -

The following steps detail the user authentication process:

1. The user selects Sign in in the website.
2. The browser redirects the user to the Azure Active Directory (Azure AD) sign in page.

3. The user signs in.
4. Azure AD redirects the user's session back to the web application. The URL includes an access token.
5. The web application calls an API and includes the access token in the authentication header. The application ID is sent as the audience ('`aud`') claim in the access token.
6. The back-end API validates the access token.

Requirements -

Corporate website -

Communications and content must be secured by using SSL.

Communications must use HTTPS.

Data must be replicated to a secondary region and three availability zones.

Data storage costs must be minimized.

Azure Database for PostgreSQL -

The database connection string is stored in Azure Key Vault with the following attributes:

Azure Key Vault name: cpandlkeyvault

Secret name: PostgreSQLConn

Id: 80df3e46ffcd4f1cb187f79905e9a1e8

The connection information is updated frequently. The application must always use the latest information to connect to the database.

Azure Service Bus and Azure Event Grid

Azure Event Grid must use Azure Service Bus for queue-based load leveling.

Events in Azure Event Grid must be routed directly to Service Bus queues for use in buffering.

Events from Azure Service Bus and other Azure services must continue to be routed to Azure Event Grid for processing.

Security -

All SSL certificates and credentials must be stored in Azure Key Vault.

File access must restrict access by IP, protocol, and Azure AD rights.

All user accounts and processes must receive only those privileges which are essential to perform their intended function.

Compliance -

Auditing of the file updates and transfers must be enabled to comply with General Data Protection Regulation (GDPR). The file updates must be read-only, stored in the order in which they occurred, include only create, update, delete, and copy operations, and be retained for compliance reasons.

Issues -

Corporate website -

While testing the site, the following error message displays:

CryptographicException: The system cannot find the file specified.

Function app -

You perform local testing for the RequestUserApproval function. The following error message displays:

'Timeout value of 00:10:00 exceeded by function: RequestUserApproval'

The same error message displays when you test the function in an Azure development environment when you run the following Kusto query:

FunctionAppLogs -

```
| where FunctionName == "RequestUserApproval"
```

Logic app -

You test the Logic app in a development environment. The following error message displays:

'400 Bad Request'

Troubleshooting of the error shows an HttpTrigger action to call the RequestUserApproval function.

Code -

Corporate website -

Security.cs:

```

SC01 public class Security
SC02 {
SC03 var bytes = System.IO.File.ReadAllBytes("~/var/ssl/private");
SC04 var cert = new System.Security.Cryptography.X509Certificate2(bytes);
SC05 var certName = cert.FriendlyName;
SC06 }

```

Function app -

RequestUserApproval.cs:

```

RA01 public static class RequestUserApproval
RA02 {
RA03 [FunctionName("RequestUserApproval")]
RA04 public static async Task<IActionResult> Run(
RA05 [HttpTrigger(AuthorizationLevel.Function, "get", "post", Route = null)] HttpRequest req,
ILogger log)
RA06 {
RA07     log.LogInformation("RequestUserApproval function processed a request.");
RA08 ...
RA09     return ProcessRequest(req)
RA10     ? (ActionResult)new OkObjectResult($"User approval processed")
RA11     : new BadRequestObjectResult("Failed to process user approval");
RA12 }
RA13 private static bool ProcessRequest(HttpContext req)
RA14 {
RA15     ...
RA16 }
RA17 }

```

### Question

HOTSPOT -

You need to configure the integration for Azure Service Bus and Azure Event Grid.

How should you complete the CLI statement? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

### Answer Area

az <div style="border: 1px solid #ccc; padding: 2px; width: 150px; height: 100px; margin-bottom: 10px;"></div> <div style="border: 1px solid #ccc; padding: 2px; width: 150px; height: 100px; margin-bottom: 10px;"></div> <div style="border: 1px solid #ccc; padding: 2px; width: 150px; height: 100px; margin-bottom: 10px;"></div>	<div style="border: 1px solid #ccc; padding: 2px; width: 150px; height: 100px; margin-bottom: 10px;"></div> <div style="border: 1px solid #ccc; padding: 2px; width: 150px; height: 100px; margin-bottom: 10px;"></div> <div style="border: 1px solid #ccc; padding: 2px; width: 150px; height: 100px; margin-bottom: 10px;"></div>	<div style="border: 1px solid #ccc; padding: 2px; width: 150px; height: 100px; margin-bottom: 10px;"></div> <div style="border: 1px solid #ccc; padding: 2px; width: 150px; height: 100px; margin-bottom: 10px;"></div> <div style="border: 1px solid #ccc; padding: 2px; width: 150px; height: 100px; margin-bottom: 10px;"></div>
		create --source-resource-id \$topicid --name \$name --
		--endpoint \$endpoint

### Correct Answer:

#### Answer Area

az <div style="border: 1px solid #ccc; padding: 2px; width: 150px; height: 100px; background-color: #e0f2e0; margin-bottom: 10px;"></div> <div style="border: 1px solid #ccc; padding: 2px; width: 150px; height: 100px; background-color: #e0f2e0; margin-bottom: 10px;"></div> <div style="border: 1px solid #ccc; padding: 2px; width: 150px; height: 100px; background-color: #e0f2e0; margin-bottom: 10px;"></div>	<div style="border: 1px solid #ccc; padding: 2px; width: 150px; height: 100px; background-color: #e0f2e0; margin-bottom: 10px;"></div> <div style="border: 1px solid #ccc; padding: 2px; width: 150px; height: 100px; background-color: #e0f2e0; margin-bottom: 10px;"></div> <div style="border: 1px solid #ccc; padding: 2px; width: 150px; height: 100px; background-color: #e0f2e0; margin-bottom: 10px;"></div>	<div style="border: 1px solid #ccc; padding: 2px; width: 150px; height: 100px; background-color: #e0f2e0; margin-bottom: 10px;"></div> <div style="border: 1px solid #ccc; padding: 2px; width: 150px; height: 100px; background-color: #e0f2e0; margin-bottom: 10px;"></div> <div style="border: 1px solid #ccc; padding: 2px; width: 150px; height: 100px; background-color: #e0f2e0; margin-bottom: 10px;"></div>
		create --source-resource-id \$topicid --name \$name --
		--endpoint \$endpoint

Box 1: eventgrid -

To create event subscription use: az eventgrid event-subscription create

Box 2: event-subscription -

Box 3: servicebusqueue -

Scenario: Azure Service Bus and Azure Event Grid

Azure Event Grid must use Azure Service Bus for queue-based load leveling.

Events in Azure Event Grid must be routed directly to Service Bus queues for use in buffering.

Events from Azure Service Bus and other Azure services must continue to be routed to Azure Event Grid for processing.

Reference:

[https://docs.microsoft.com/en-us/cli/azure/eventgrid/event-subscription?view=azure-cli-latest#az\\_eventgrid\\_event\\_subscription\\_create](https://docs.microsoft.com/en-us/cli/azure/eventgrid/event-subscription?view=azure-cli-latest#az_eventgrid_event_subscription_create)

Question #2

**Introductory Info**

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

City Power & Light company provides electrical infrastructure monitoring solutions for homes and businesses. The company is migrating solutions to Azure.

Current environment -

Architecture overview -

The company has a public website located at <http://www.cpndl.com/>. The site is a single-page web application that runs in Azure App Service on Linux. The website uses files stored in Azure Storage and cached in Azure Content Delivery Network (CDN) to serve static content.

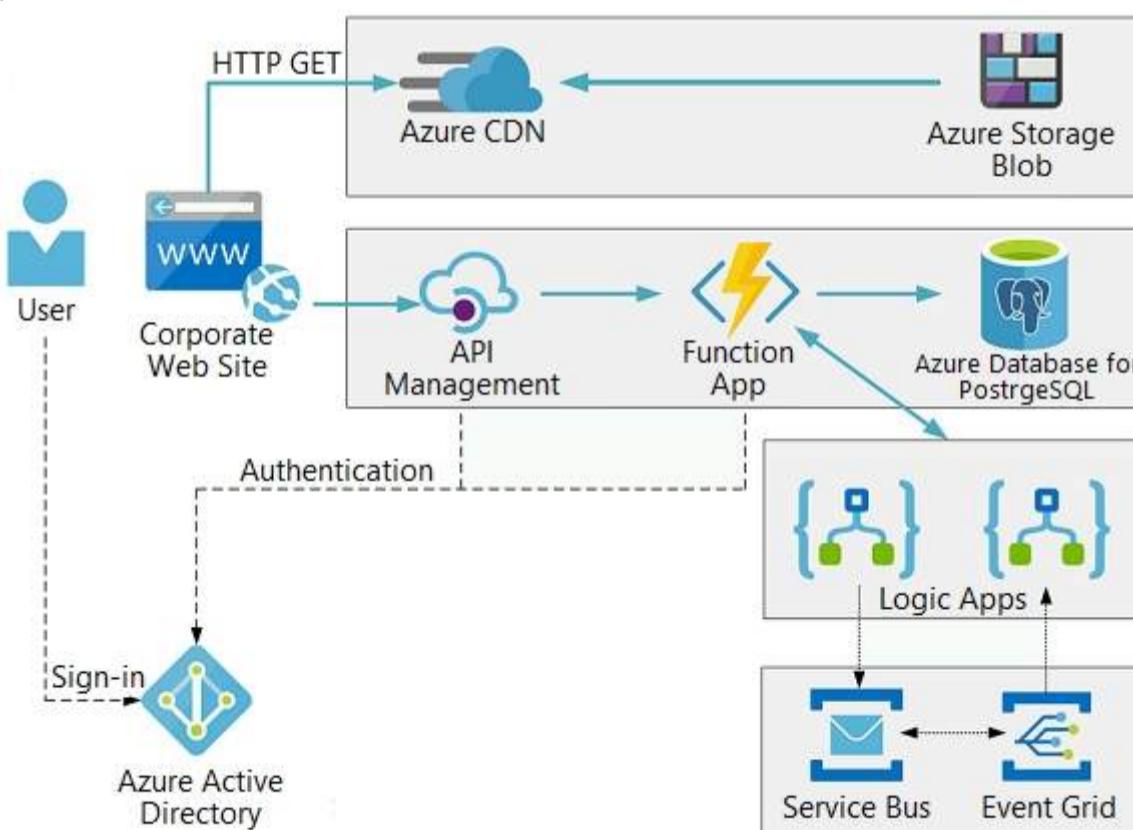
API Management and Azure Function App functions are used to process and store data in Azure Database for PostgreSQL. API Management is used to broker communications to the Azure Function app functions for Logic app integration. Logic apps are used to orchestrate the data processing while Service Bus and

Event Grid handle messaging and events.

The solution uses Application Insights, Azure Monitor, and Azure Key Vault.

Architecture diagram -

The company has several applications and services that support their business. The company plans to implement serverless computing where possible. The overall architecture is shown below.



User authentication -

The following steps detail the user authentication process:

1. The user selects Sign in in the website.
2. The browser redirects the user to the Azure Active Directory (Azure AD) sign in page.

3. The user signs in.
4. Azure AD redirects the user's session back to the web application. The URL includes an access token.
5. The web application calls an API and includes the access token in the authentication header. The application ID is sent as the audience ('`aud`') claim in the access token.
6. The back-end API validates the access token.

Requirements -

Corporate website -

Communications and content must be secured by using SSL.

Communications must use HTTPS.

Data must be replicated to a secondary region and three availability zones.

Data storage costs must be minimized.

Azure Database for PostgreSQL -

The database connection string is stored in Azure Key Vault with the following attributes:

Azure Key Vault name: cpandlkeyvault

Secret name: PostgreSQLConn

Id: 80df3e46ffcd4f1cb187f79905e9a1e8

The connection information is updated frequently. The application must always use the latest information to connect to the database.

Azure Service Bus and Azure Event Grid

Azure Event Grid must use Azure Service Bus for queue-based load leveling.

Events in Azure Event Grid must be routed directly to Service Bus queues for use in buffering.

Events from Azure Service Bus and other Azure services must continue to be routed to Azure Event Grid for processing.

Security -

All SSL certificates and credentials must be stored in Azure Key Vault.

File access must restrict access by IP, protocol, and Azure AD rights.

All user accounts and processes must receive only those privileges which are essential to perform their intended function.

Compliance -

Auditing of the file updates and transfers must be enabled to comply with General Data Protection Regulation (GDPR). The file updates must be read-only, stored in the order in which they occurred, include only create, update, delete, and copy operations, and be retained for compliance reasons.

Issues -

Corporate website -

While testing the site, the following error message displays:

CryptographicException: The system cannot find the file specified.

Function app -

You perform local testing for the RequestUserApproval function. The following error message displays:

'Timeout value of 00:10:00 exceeded by function: RequestUserApproval'

The same error message displays when you test the function in an Azure development environment when you run the following Kusto query:

FunctionAppLogs -

```
| where FunctionName == "RequestUserApproval"
```

Logic app -

You test the Logic app in a development environment. The following error message displays:

'400 Bad Request'

Troubleshooting of the error shows an HttpTrigger action to call the RequestUserApproval function.

Code -

Corporate website -

Security.cs:

```
SC01 public class Security
SC02 {
SC03     var bytes = System.IO.File.ReadAllBytes("~/var/ssl/private");
SC04     var cert = new System.Security.Cryptography.X509Certificate2(bytes);
SC05     var certName = cert.FriendlyName;
SC06 }
```

Function app -

RequestUserApproval.cs:

```
RA01 public static class RequestUserApproval
RA02 {
RA03     [FunctionName("RequestUserApproval")]
RA04     public static async Task<IActionResult> Run(
RA05         [HttpTrigger(AuthorizationLevel.Function, "get", "post", Route = null)] HttpRequest req,
RA06         ILogger log)
RA07     {
RA08         log.LogInformation("RequestUserApproval function processed a request.");
RA09         ...
RA10         return ProcessRequest(req)
RA11             ? (ActionResult)new OkObjectResult($"User approval processed")
RA12             : new BadRequestObjectResult("Failed to process user approval");
RA13     }
RA14     private static bool ProcessRequest(HttpRequest req)
RA15     {
RA16         ...
RA17     }
```

### Question

You need to ensure that all messages from Azure Event Grid are processed.

What should you use?

- A. Azure Event Grid topic
- B. Azure Service Bus topic
- C. Azure Service Bus queue
- D. Azure Storage queue
- E. Azure Logic App custom connector

#### Correct Answer: C

As a solution architect/developer, you should consider using Service Bus queues when:

☞ Your solution needs to receive messages without having to poll the queue. With Service Bus, you can achieve it by using a long-polling receive operation using the TCP-based protocols that Service Bus supports.

Reference:

<https://docs.microsoft.com/en-us/azure/service-bus-messaging/service-bus-azure-and-service-bus-queues-compared-contrasted>

**Topic 20 - Testlet 21**

Question #1

## Introductory Info

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

You are a developer for Proseware, Inc. You are developing an application that applies a set of governance policies for Proseware's internal services, external services, and applications. The application will also provide a shared library for common functionality.

Requirements -

Policy service -

You develop and deploy a stateful ASP.NET Core 2.1 web application named Policy service to an Azure App Service Web App. The application reacts to events from Azure Event Grid and performs policy actions based on those events.

The application must include the Event Grid Event ID field in all Application Insights telemetry.

Policy service must use Application Insights to automatically scale with the number of policy actions that it is performing.

Policies -

Log policy -

All Azure App Service Web Apps must write logs to Azure Blob storage. All log files should be saved to a container named logdrop. Logs must remain in the container for 15 days.

Authentication events -

Authentication events are used to monitor users signing in and signing out. All authentication events must be processed by Policy service. Sign outs must be processed as quickly as possible.

PolicyLib -

You have a shared library named PolicyLib that contains functionality common to all ASP.NET Core web services and applications. The PolicyLib library must:

Exclude non-user actions from Application Insights telemetry.

Provide methods that allow a web service to scale itself.

Ensure that scaling actions do not disrupt application usage.

Other -

Anomaly detection service -

You have an anomaly detection service that analyzes log information for anomalies. It is implemented as an Azure Machine Learning model. The model is deployed as a web service. If an anomaly is detected, an Azure Function that emails administrators is called by using an HTTP WebHook.

Health monitoring -

All web applications and services have health monitoring at the /health service endpoint.

Issues -

**Policy loss -**

When you deploy Policy service, policies may not be applied if they were in the process of being applied during the deployment.

**Performance issue -**

When under heavy load, the anomaly detection service undergoes slowdowns and rejects connections.

**Notification latency -**

Users report that anomaly detection emails can sometimes arrive several minutes after an anomaly is detected.

**App code -****EventGridController.cs -**

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.

```
EventGridController.cs
EG01 public class EventGridController : Controller
EG02 {
EG03     public static AsyncLocal<string> EventId = new AsyncLocal<string>();
EG04     public IActionResult Process([FromBody] string eventsJson)
EG05     {
EG06         var events = JArray.Parse(eventsJson);
EG07
EG08         foreach (var @event in events)
EG09         {
EG10             EventId.Value = @event["id"].ToString();
EG11             if (@event["topic"].ToString().Contains("providers/Microsoft.Storage"))
EG12             {
EG13                 SendToAnomalyDetectionService(@event["data"]["url"].ToString());
EG14             }
EG15
EG16             {
EG17                 EnsureLogging(@event["subject"].ToString());
EG18             }
EG19         }
EG20         return null;
EG21     }
EG22     private void EnsureLogging(string resource)
EG23     {
EG24         . . .
EG25     }
EG26     private async Task SendToAnomalyDetectionService(string uri)
EG27     {
EG28         var content = GetLogData(uri);
EG29         var scoreRequest = new
EG30         {
EG31             Inputs = new Dictionary<string, List<Dictionary<string, string>>>()
EG32             {
EG33                 {
EG34                     "input1",
EG35                     new List<Dictionary<string, string>>()
EG36                     {
EG37                         new Dictionary<string, string>()
EG38                         {
EG39                             {
EG40                                 "logcontent", content
EG41                             }
EG42                         }
EG43                     }
EG44                 },
EG45             },
EG46             GlobalParameters = new Dictionary<string, string>() { }
EG47         };
EG48         var result = await (new HttpClient()).PostAsJsonAsync("...", scoreRequest);
EG49         var rawModelResult = await result.Content.ReadAsStringAsync();
EG50         var modelResult = JObject.Parse(rawModelResult);
EG51         if (modelResult["notify"].HasValues)
EG52         {
EG53             . . .
EG54         }
EG55     }
EG56     private (string name, string resourceGroup) ParseResourceId(string resourceId)
EG57     {
EG58         . . .
EG59     }
EG60     private string GetLogData(string uri)
EG61     {
EG62         . . .
EG63     }
EG64     static string BlobStoreAccountSAS(string containerName)
EG65     {
EG66         . . .
EG67     }
EG68 }
```

LoginEvent.cs -

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.

#### LoginEvent.cs

```
LE01 public class LoginEvent
LE02 {
LE03
LE04     public string subject { get; set; }
LE05     public DateTime eventTime { get; set; }
LE06     public Dictionary<string, string> data { get; set; }
LE07     public string Serialize()
LE08     {
LE09         return JsonConvert.SerializeObject(this);
LE10     }
LE11 }
```

#### Question

DRAG DROP -

You need to add code at line EG15 in EventGridController.cs to ensure that the Log policy applies to all services.

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 one point.

Select and Place:

#### Code segments

topic  
status  
eventType  
Succeeded  
operationName  
resourceProvider

#### Answer Area

```
if {
    @event[ "data" ][ " code segment " ].ToString() == " code segment "
    &&
    @event[ "data" ][ " code segment " ].ToString() == "Microsoft.Web/sites/write"
}
```

#### Correct Answer:

#### Code segments

topic  
status  
eventType  
Succeeded  
operationName  
resourceProvider

#### Answer Area

```
if {
    @event[ "data" ][ " status " ].ToString() == " Succeeded "
    &&
    @event[ "data" ][ " operationName " ].ToString() == "Microsoft.Web/sites/write"
}
```

Scenario, Log policy: All Azure App Service Web Apps must write logs to Azure Blob storage.

Box 1: Status -

Box 2: Succeeded -

Box 3: operationName -

Microsoft.Web/sites/write is resource provider operation. It creates a new Web App or updates an existing one.

Reference:

<https://docs.microsoft.com/en-us/azure/role-based-access-control/resource-provider-operations>

Question #2

## Introductory Info

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

You are a developer for Proseware, Inc. You are developing an application that applies a set of governance policies for Proseware's internal services, external services, and applications. The application will also provide a shared library for common functionality.

Requirements -

Policy service -

You develop and deploy a stateful ASP.NET Core 2.1 web application named Policy service to an Azure App Service Web App. The application reacts to events from Azure Event Grid and performs policy actions based on those events.

The application must include the Event Grid Event ID field in all Application Insights telemetry.

Policy service must use Application Insights to automatically scale with the number of policy actions that it is performing.

Policies -

Log policy -

All Azure App Service Web Apps must write logs to Azure Blob storage. All log files should be saved to a container named logdrop. Logs must remain in the container for 15 days.

Authentication events -

Authentication events are used to monitor users signing in and signing out. All authentication events must be processed by Policy service. Sign outs must be processed as quickly as possible.

PolicyLib -

You have a shared library named PolicyLib that contains functionality common to all ASP.NET Core web services and applications. The PolicyLib library must:

Exclude non-user actions from Application Insights telemetry.

Provide methods that allow a web service to scale itself.

Ensure that scaling actions do not disrupt application usage.

Other -

Anomaly detection service -

You have an anomaly detection service that analyzes log information for anomalies. It is implemented as an Azure Machine Learning model. The model is deployed as a web service. If an anomaly is detected, an Azure Function that emails administrators is called by using an HTTP WebHook.

Health monitoring -

All web applications and services have health monitoring at the /health service endpoint.

Issues -

**Policy loss -**

When you deploy Policy service, policies may not be applied if they were in the process of being applied during the deployment.

**Performance issue -**

When under heavy load, the anomaly detection service undergoes slowdowns and rejects connections.

**Notification latency -**

Users report that anomaly detection emails can sometimes arrive several minutes after an anomaly is detected.

**App code -****EventGridController.cs -**

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.

```
EventGridController.cs
EG01 public class EventGridController : Controller
EG02 {
EG03     public static AsyncLocal<string> EventId = new AsyncLocal<string>();
EG04     public IActionResult Process([FromBody] string eventsJson)
EG05     {
EG06         var events = JArray.Parse(eventsJson);
EG07
EG08         foreach (var @event in events)
EG09         {
EG10             EventId.Value = @event["id"].ToString();
EG11             if (@event["topic"].ToString().Contains("providers/Microsoft.Storage"))
EG12             {
EG13                 SendToAnomalyDetectionService(@event["data"]["url"].ToString());
EG14             }
EG15
EG16             {
EG17                 EnsureLogging(@event["subject"].ToString());
EG18             }
EG19         }
EG20         return null;
EG21     }
EG22     private void EnsureLogging(string resource)
EG23     {
EG24         . . .
EG25     }
EG26     private async Task SendToAnomalyDetectionService(string uri)
EG27     {
EG28         var content = GetLogData(uri);
EG29         var scoreRequest = new
EG30         {
EG31             Inputs = new Dictionary<string, List<Dictionary<string, string>>>()
EG32             {
EG33                 {
EG34                     "input1",
EG35                     new List<Dictionary<string, string>>()
EG36                     {
EG37                         new Dictionary<string, string>()
EG38                         {
EG39                             {
EG40                                 "logcontent", content
EG41                             }
EG42                         }
EG43                     }
EG44                 },
EG45             },
EG46             GlobalParameters = new Dictionary<string, string>() { }
EG47         };
EG48         var result = await (new HttpClient()).PostAsJsonAsync("...", scoreRequest);
EG49         var rawModelResult = await result.Content.ReadAsStringAsync();
EG50         var modelResult = JObject.Parse(rawModelResult);
EG51         if (modelResult["notify"].HasValues)
EG52         {
EG53             . . .
EG54         }
EG55     }
EG56     private (string name, string resourceGroup) ParseResourceId(string resourceId)
EG57     {
EG58         . . .
EG59     }
EG60     private string GetLogData(string uri)
EG61     {
EG62         . . .
EG63     }
EG64     static string BlobStoreAccountSAS(string containerName)
EG65     {
EG66         . . .
EG67     }
EG68 }
```

LoginEvent.cs -

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.

**LoginEvent.cs**

```
LE01 public class LoginEvent
LE02 {
LE03
LE04     public string subject { get; set; }
LE05     public DateTime eventTime { get; set; }
LE06     public Dictionary<string, string> data { get; set; }
LE07     public string Serialize()
LE08     {
LE09         return JsonConvert.SerializeObject(this);
LE10     }
LE11 }
```

**Question**

HOTSPOT -

You need to insert code at line LE03 of LoginEvent.cs to ensure that all authentication events are processed correctly.

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

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

public string ( get; set; )

id  
eventType  
dataVersion  
metadataVersion

public string ( get; set; )

id  
eventType  
dataVersion  
metadataVersion

public string ( get; set; )

id  
eventType  
dataVersion  
metadataVersion

**Answer Area**

public string ( get; set; )

id  
eventType  
dataVersion  
metadataVersion

public string ( get; set; )

id  
eventType  
dataVersion  
metadataVersion

public string ( get; set; )

id  
eventType  
dataVersion  
metadataVersion

Box 1: id -

id is a unique identifier for the event.

Question #3

## Introductory Info

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

You are a developer for Proseware, Inc. You are developing an application that applies a set of governance policies for Proseware's internal services, external services, and applications. The application will also provide a shared library for common functionality.

Requirements -

Policy service -

You develop and deploy a stateful ASP.NET Core 2.1 web application named Policy service to an Azure App Service Web App. The application reacts to events from Azure Event Grid and performs policy actions based on those events.

The application must include the Event Grid Event ID field in all Application Insights telemetry.

Policy service must use Application Insights to automatically scale with the number of policy actions that it is performing.

Policies -

Log policy -

All Azure App Service Web Apps must write logs to Azure Blob storage. All log files should be saved to a container named logdrop. Logs must remain in the container for 15 days.

Authentication events -

Authentication events are used to monitor users signing in and signing out. All authentication events must be processed by Policy service. Sign outs must be processed as quickly as possible.

PolicyLib -

You have a shared library named PolicyLib that contains functionality common to all ASP.NET Core web services and applications. The PolicyLib library must:

Exclude non-user actions from Application Insights telemetry.

Provide methods that allow a web service to scale itself.

Ensure that scaling actions do not disrupt application usage.

Other -

Anomaly detection service -

You have an anomaly detection service that analyzes log information for anomalies. It is implemented as an Azure Machine Learning model. The model is deployed as a web service. If an anomaly is detected, an Azure Function that emails administrators is called by using an HTTP WebHook.

Health monitoring -

All web applications and services have health monitoring at the /health service endpoint.

Issues -

**Policy loss -**

When you deploy Policy service, policies may not be applied if they were in the process of being applied during the deployment.

**Performance issue -**

When under heavy load, the anomaly detection service undergoes slowdowns and rejects connections.

**Notification latency -**

Users report that anomaly detection emails can sometimes arrive several minutes after an anomaly is detected.

**App code -****EventGridController.cs -**

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.

```
EventGridController.cs
EG01 public class EventGridController : Controller
EG02 {
EG03     public static AsyncLocal<string> EventId = new AsyncLocal<string>();
EG04     public IActionResult Process([FromBody] string eventsJson)
EG05     {
EG06         var events = JArray.Parse(eventsJson);
EG07
EG08         foreach (var @event in events)
EG09         {
EG10             EventId.Value = @event["id"].ToString();
EG11             if (@event["topic"].ToString().Contains("providers/Microsoft.Storage"))
EG12             {
EG13                 SendToAnomalyDetectionService(@event["data"]["url"].ToString());
EG14             }
EG15
EG16             {
EG17                 EnsureLogging(@event["subject"].ToString());
EG18             }
EG19         }
EG20         return null;
EG21     }
EG22     private void EnsureLogging(string resource)
EG23     {
EG24         . . .
EG25     }
EG26     private async Task SendToAnomalyDetectionService(string uri)
EG27     {
EG28         var content = GetLogData(uri);
EG29         var scoreRequest = new
EG30         {
EG31             Inputs = new Dictionary<string, List<Dictionary<string, string>>>()
EG32             {
EG33                 {
EG34                     "input1",
EG35                     new List<Dictionary<string, string>>()
EG36                     {
EG37                         new Dictionary<string, string>()
EG38                         {
EG39                             {
EG40                                 "logcontent", content
EG41                             }
EG42                         }
EG43                     }
EG44                 },
EG45             },
EG46             GlobalParameters = new Dictionary<string, string>() { }
EG47         };
EG48         var result = await (new HttpClient()).PostAsJsonAsync("...", scoreRequest);
EG49         var rawModelResult = await result.Content.ReadAsStringAsync();
EG50         var modelResult = JObject.Parse(rawModelResult);
EG51         if (modelResult["notify"].HasValues)
EG52         {
EG53             . . .
EG54         }
EG55     }
EG56     private (string name, string resourceGroup) ParseResourceId(string resourceId)
EG57     {
EG58         . . .
EG59     }
EG60     private string GetLogData(string uri)
EG61     {
EG62         . . .
EG63     }
EG64     static string BlobStoreAccountSAS(string containerName)
EG65     {
EG66         . . .
EG67     }
EG68 }
```

LoginEvent.cs -

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.

**LoginEvent.cs**

```
LE01 public class LoginEvent
LE02 {
LE03
LE04     public string subject { get; set; }
LE05     public DateTime eventTime { get; set; }
LE06     public Dictionary<string, string> data { get; set; }
LE07     public string Serialize()
LE08     {
LE09         return JsonConvert.SerializeObject(this);
LE10     }
LE11 }
```

**Question**

HOTSPOT -

You need to implement the Log policy.

How should you complete the EnsureLogging method in EventGridController.cs? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

```
var client = new WebSiteManagementClient(. . .);
var id = ParseResourceId(resource);
var appSettings = new StringDictionary(name: "properties",
    properties: new Dictionary<string, string> {
        {"DIAGNOSTICS_AZUREBLOBCONTAINERSASURL", BlobStoreAccountSAS(""),
            logs
            logdrop
        },
        {"DIAGNOSTICS_AZUREBLOBRETENTIONINDAYS", "15"} // 15 or 30
    });
client.WebApps. // (UploadLoggingSettings or UpdateApplicationSetting)
id.resourceGroup,
```

**Topic 21 - Testlet 3**

**Correct Answer:**

**Answer Area**

```
var client = new WebSiteManagementClient(. . .);
var id = ParseResourceId(resource);
var appSettings = new StringDictionary(name: "properties",
    properties: new Dictionary<string, string> {
        {"DIAGNOSTICS_AZUREBLOBCONTAINERSASURL", BlobStoreAccountSAS(""),
            logs
            logdrop
        },
        {"DIAGNOSTICS_AZUREBLOBRETENTIONINDAYS", "15"} // 15
    });
client.WebApps. // (UploadLoggingSettings or UpdateApplicationSetting)
id.resourceGroup,
id.name, appSettings);
```

Box 1: logdrop -

All log files should be saved to a container named logdrop.

Question #1

## Introductory Info

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

You are a developer for Proseware, Inc. You are developing an application that applies a set of governance policies for Proseware's internal services, external services, and applications. The application will also provide a shared library for common functionality.

Requirements -

Policy service -

You develop and deploy a stateful ASP.NET Core 2.1 web application named Policy service to an Azure App Service Web App. The application reacts to events from Azure Event Grid and performs policy actions based on those events.

The application must include the Event Grid Event ID field in all Application Insights telemetry.

Policy service must use Application Insights to automatically scale with the number of policy actions that it is performing.

Policies -

Log policy -

All Azure App Service Web Apps must write logs to Azure Blob storage. All log files should be saved to a container named logdrop. Logs must remain in the container for 15 days.

Authentication events -

Authentication events are used to monitor users signing in and signing out. All authentication events must be processed by Policy service. Sign outs must be processed as quickly as possible.

PolicyLib -

You have a shared library named PolicyLib that contains functionality common to all ASP.NET Core web services and applications. The PolicyLib library must:

Exclude non-user actions from Application Insights telemetry.

Provide methods that allow a web service to scale itself.

Ensure that scaling actions do not disrupt application usage.

Other -

Anomaly detection service -

You have an anomaly detection service that analyzes log information for anomalies. It is implemented as an Azure Machine Learning model. The model is deployed as a web service. If an anomaly is detected, an Azure Function that emails administrators is called by using an HTTP WebHook.

Health monitoring -

All web applications and services have health monitoring at the /health service endpoint.

Issues -

**Policy loss -**

When you deploy Policy service, policies may not be applied if they were in the process of being applied during the deployment.

**Performance issue -**

When under heavy load, the anomaly detection service undergoes slowdowns and rejects connections.

**Notification latency -**

Users report that anomaly detection emails can sometimes arrive several minutes after an anomaly is detected.

**App code -****EventGridController.cs -**

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.

```
EventGridController.cs
EG01 public class EventGridController : Controller
EG02 {
EG03     public static AsyncLocal<string> EventId = new AsyncLocal<string>();
EG04     public IActionResult Process([FromBody] string eventsJson)
EG05     {
EG06         var events = JArray.Parse(eventsJson);
EG07
EG08         foreach (var @event in events)
EG09         {
EG10             EventId.Value = @event["id"].ToString();
EG11             if (@event["topic"].ToString().Contains("providers/Microsoft.Storage"))
EG12             {
EG13                 SendToAnomalyDetectionService(@event["data"]["url"].ToString());
EG14             }
EG15
EG16             {
EG17                 EnsureLogging(@event["subject"].ToString());
EG18             }
EG19         }
EG20         return null;
EG21     }
EG22     private void EnsureLogging(string resource)
EG23     {
EG24     . .
EG25     }
EG26     private async Task SendToAnomalyDetectionService(string uri)
EG27     {
EG28         var content = GetLogData(uri);
EG29         var scoreRequest = new
EG30         {
EG31             Inputs = new Dictionary<string, List<Dictionary<string, string>>>()
EG32             {
EG33                 {
EG34                     "input1",
EG35                     new List<Dictionary<string, string>>()
EG36                     {
EG37                         new Dictionary<string, string>()
EG38                         {
EG39                             {
EG40                                 "logcontent", content
EG41                             }
EG42                         }
EG43                     }
EG44                 },
EG45             },
EG46             GlobalParameters = new Dictionary<string, string>() { }
EG47         };
EG48         var result = await (new HttpClient()).PostAsJsonAsync("...", scoreRequest);
EG49         var rawModelResult = await result.Content.ReadAsStringAsync();
EG50         var modelResult = JObject.Parse(rawModelResult);
EG51         if (modelResult["notify"].HasValues)
EG52         {
EG53             . .
EG54         }
EG55     }
EG56     private (string name, string resourceGroup) ParseResourceId(string resourceId)
EG57     {
EG58     . .
EG59     }
EG60     private string GetLogData(string uri)
EG61     {
EG62     . .
EG63     }
EG64     static string BlobStoreAccountSAS(string containerName)
EG65     {
EG66     . .
EG67     }
EG68 }
```

LoginEvent.cs -

Relevant portions of the .ann files are shown below. Line numbers are included for reference only and include a two-character prefix that denotes

## Topic 22 - Testlet 4

```
LE01 public class LoginEvent
LE02 {
LE03
LE04     public string subject { get; set; }
LE05     public DateTime eventTime { get; set; }
LE06     public Dictionary<string, string> data { get; set; }
LE07     public string Serialize()
LE08     {
LE09         return JsonConvert.SerializeObject(this);
LE10     }
LE11 }
```

### Question

You need to resolve a notification latency issue.

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

NOTE: Each correct selection is worth one point.

- A. Set Always On to true.
- B. Ensure that the Azure Function is using an App Service plan.
- C. Set Always On to false.
- D. Ensure that the Azure Function is set to use a consumption plan.

#### Correct Answer: AB

Azure Functions can run on either a Consumption Plan or a dedicated App Service Plan. If you run in a dedicated mode, you need to turn on the Always On setting for your Function App to run properly. The Function runtime will go idle after a few minutes of inactivity, so only HTTP triggers will actually "wake up" your functions. This is similar to how WebJobs must have Always On enabled.

Scenario: Notification latency: Users report that anomaly detection emails can sometimes arrive several minutes after an anomaly is detected.

Anomaly detection service: You have an anomaly detection service that analyzes log information for anomalies. It is implemented as an Azure Machine Learning model. The model is deployed as a web service.

If an anomaly is detected, an Azure Function that emails administrators is called by using an HTTP WebHook.

Reference:

<https://github.com/Azure/Azure-Functions/wiki/Enable-Always-On-when-running-on-dedicated-App-Service-Plan>

Question #1

## Introductory Info

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

Overview -

You are a developer for Contoso, Ltd. The company has a social networking website that is developed as a Single Page Application (SPA). The main web application for the social networking website loads user uploaded content from blob storage.

You are developing a solution to monitor uploaded data for inappropriate content. The following process occurs when users upload content by using the SPA:

Messages are sent to ContentUploadService.

Content is processed by ContentAnalysisService.

After processing is complete, the content is posted to the social network or a rejection message is posted in its place.

The ContentAnalysisService is deployed with Azure Container Instances from a private Azure Container Registry named contosoimages.

The solution will use eight CPU cores.

Azure Active Directory -

Contoso, Ltd. uses Azure Active Directory (Azure AD) for both internal and guest accounts.

Requirements -

ContentAnalysisService -

The company's data science group built ContentAnalysisService which accepts user generated content as a string and returns a probable value for inappropriate content. Any values over a specific threshold must be reviewed by an employee of Contoso, Ltd.

You must create an Azure Function named CheckUserContent to perform the content checks.

Costs -

You must minimize costs for all Azure services.

Manual review -

To review content, the user must authenticate to the website portion of the ContentAnalysisService using their Azure AD credentials. The website is built using

React and all pages and API endpoints require authentication. In order to review content a user must be part of a ContentReviewer role. All completed reviews must include the reviewer's email address for auditing purposes.

High availability -

All services must run in multiple regions. The failure of any service in a region must not impact overall application availability.

Monitoring -

An alert must be raised if the ContentUploadService uses more than 80 percent of available CPU cores.

Security -

You have the following security requirements:

- Any web service accessible over the Internet must be protected from cross site scripting attacks.
- All websites and services must use SSL from a valid root certificate authority.
- Azure Storage access keys must only be stored in memory and must be available only to the service.
- All Internal services must only be accessible from internal Virtual Networks (VNets).
- All parts of the system must support inbound and outbound traffic restrictions.
- All service calls must be authenticated by using Azure AD.

User agreements -

When a user submits content, they must agree to a user agreement. The agreement allows employees of Contoso, Ltd. to review content, store cookies on user devices, and track user's IP addresses.

Information regarding agreements is used by multiple divisions within Contoso, Ltd.

User responses must not be lost and must be available to all parties regardless of individual service uptime. The volume of agreements is expected to be in the millions per hour.

Validation testing -

When a new version of the ContentAnalysisService is available the previous seven days of content must be processed with the new version to verify that the new version does not significantly deviate from the old version.

Issues -

Users of the ContentUploadService report that they occasionally see HTTP 502 responses on specific pages.

Code -

ContentUploadService -

```
CS01 apiVersion: '2018-10-01'
CS02 type: Microsoft.ContainerInstance/containerGroups
CS03 location: westus
CS04 name: contentUploadService
CS05 properties:
CS06   containers:
CS07     - name: service
CS08       properties:
CS09         image: contoso/contentUploadService:latest
CS10       ports:
CS11         - port: 80
CS12           protocol: TCP
CS13       resources:
CS14         requests:
CS15           cpu: 1.0
CS16           memoryInGB: 1.5
CS17
CS18 ipAddress:
CS19   ip: 10.23.121.112
CS20   ports:
CS21     - port: 80
CS22       protocol: TCP
CS23
CS24
CS25 networkProfile:
CS26
id: /subscriptions/98...19/resourceGroups/container/providers/Microsoft.Network/networkProfiles/subnet
```

**ApplicationManifest -**

```
AM01 {  
AM02     "id" : "2b079f03-9b06-2d44-98bb-e9182901fcb6",  
AM03     "appId" : "7118a7f0-b5c2-4c9d-833c-3d711396fe65",  
AM04  
AM05     "createdDateTime" : "2019-12-24T06:01:44Z",  
AM06     "logoUrl" : null,  
AM07     "logoutUrl" : null,  
AM08     "name" : "ContentAnalysisService",  
AM09  
AM10  
AM11     "orgRestrictions" : [],  
AM12     "parentalControlSettings" : {  
AM13         "countriesBlockedForMinors" : [],  
AM14         "legalAgeGroupRule" : "Allow"  
AM15     },  
AM16     "passwordCredentials" : []  
AM17 }
```

**Question**

HOTSPOT -

You need to ensure that validation testing is triggered per the requirements.

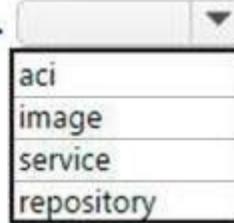
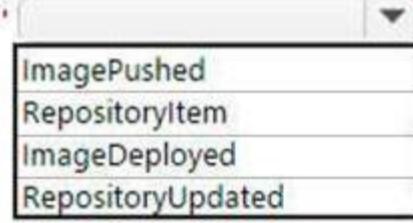
How should you complete the code segment? To answer, select the appropriate values in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

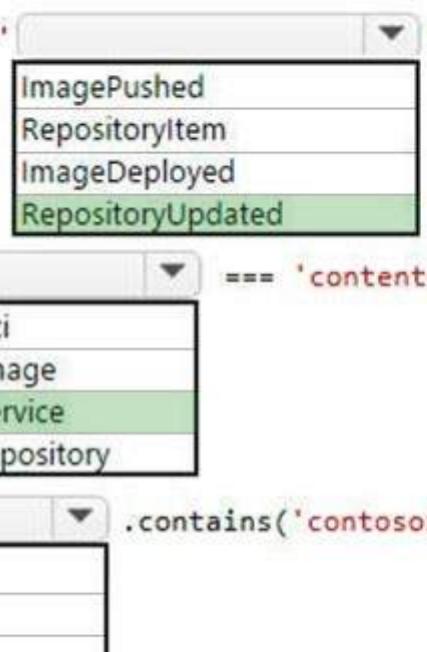
**Answer Area**

```
var event = getEvent();  
if (event.eventType === 'ImagePushed') {  
    && event.data.target === 'contentanalysisservice'  
    && event.data.topic.contains('contosoimages'))  
{  
    startValidationTesting();  
}
```



**Answer Area**

```
var event = getEvent();
if (event.eventType === 'RepositoryUpdated' && event.data.target.service === 'contentanalysisservice' && event.data.imageCollection.contains('contosoimages')) {
    startValidationTesting();
}
```



**Correct Answer:**

Box 1: RepositoryUpdated -

When a new version of the ContentAnalysisService is available the previous seven days of content must be processed with the new version to verify that the new version does not significantly deviate from the old version.

Box 2: service -

Box 3: imageCollection -

Reference:

<https://docs.microsoft.com/en-us/azure/devops/notifications/oob-supported-event-types>

Question #2

## Introductory Info

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

Overview -

You are a developer for Contoso, Ltd. The company has a social networking website that is developed as a Single Page Application (SPA). The main web application for the social networking website loads user uploaded content from blob storage.

You are developing a solution to monitor uploaded data for inappropriate content. The following process occurs when users upload content by using the SPA:

Messages are sent to ContentUploadService.

Content is processed by ContentAnalysisService.

After processing is complete, the content is posted to the social network or a rejection message is posted in its place.

The ContentAnalysisService is deployed with Azure Container Instances from a private Azure Container Registry named contosoimages.

The solution will use eight CPU cores.

Azure Active Directory -

Contoso, Ltd. uses Azure Active Directory (Azure AD) for both internal and guest accounts.

Requirements -

ContentAnalysisService -

The company's data science group built ContentAnalysisService which accepts user generated content as a string and returns a probable value for inappropriate content. Any values over a specific threshold must be reviewed by an employee of Contoso, Ltd.

You must create an Azure Function named CheckUserContent to perform the content checks.

Costs -

You must minimize costs for all Azure services.

Manual review -

To review content, the user must authenticate to the website portion of the ContentAnalysisService using their Azure AD credentials. The website is built using

React and all pages and API endpoints require authentication. In order to review content a user must be part of a ContentReviewer role. All completed reviews must include the reviewer's email address for auditing purposes.

High availability -

All services must run in multiple regions. The failure of any service in a region must not impact overall application availability.

Monitoring -

An alert must be raised if the ContentUploadService uses more than 80 percent of available CPU cores.

Security -

You have the following security requirements:

Any web service accessible over the Internet must be protected from cross site scripting attacks.

All websites and services must use SSL from a valid root certificate authority.

Azure Storage access keys must only be stored in memory and must be available only to the service.

All Internal services must only be accessible from internal Virtual Networks (VNets).

All parts of the system must support inbound and outbound traffic restrictions.

All service calls must be authenticated by using Azure AD.

User agreements -

When a user submits content, they must agree to a user agreement. The agreement allows employees of Contoso, Ltd. to review content, store cookies on user devices, and track user's IP addresses.

Information regarding agreements is used by multiple divisions within Contoso, Ltd.

User responses must not be lost and must be available to all parties regardless of individual service uptime. The volume of agreements is expected to be in the millions per hour.

Validation testing -

When a new version of the ContentAnalysisService is available the previous seven days of content must be processed with the new version to verify that the new version does not significantly deviate from the old version.

Issues -

Users of the ContentUploadService report that they occasionally see HTTP 502 responses on specific pages.

Code -

ContentUploadService -

```
CS01 apiVersion: '2018-10-01'
CS02 type: Microsoft.ContainerInstance/containerGroups
CS03 location: westus
CS04 name: contentUploadService
CS05 properties:
CS06   containers:
CS07     - name: service
CS08       properties:
CS09         image: contoso/contentUploadService:latest
CS10       ports:
CS11         - port: 80
CS12           protocol: TCP
CS13       resources:
CS14         requests:
CS15           cpu: 1.0
CS16           memoryInGB: 1.5
CS17
CS18 ipAddress:
CS19   ip: 10.23.121.112
CS20   ports:
CS21     - port: 80
CS22       protocol: TCP
CS23
CS24
CS25 networkProfile:
CS26
id: /subscriptions/98...19/resourceGroups/container/providers/Microsoft.Network/networkProfiles/subnet
```

**ApplicationManifest -**

```
AM01 {  
AM02     "id" : "2b079f03-9b06-2d44-98bb-e9182901fcb6",  
AM03     "appId" : "7118a7f0-b5c2-4c9d-833c-3d711396fe65",  
AM04  
AM05     "createdDateTime" : "2019-12-24T06:01:44Z",  
AM06     "logoUrl" : null,  
AM07     "logoutUrl" : null,  
AM08     "name" : "ContentAnalysisService",  
AM09  
AM10  
AM11     "orgRestrictions" : [],  
AM12     "parentalControlSettings" : {  
AM13         "countriesBlockedForMinors" : [],  
AM14         "legalAgeGroupRule" : "Allow"  
AM15     },  
AM16     "passwordCredentials" : []  
AM17 }
```

**Question**

You need to deploy the CheckUserContent Azure Function. The solution must meet the security and cost requirements.

Which hosting model should you use?

- A. Premium plan
- B. App Service plan
- C. Consumption plan

**Correct Answer: B** 

Scenario:

You must minimize costs for all Azure services.

All Internal services must only be accessible from internal Virtual Networks (VNets).

Best for long-running scenarios where Durable Functions can't be used. Consider an App Service plan in the following situations:

- ⇒ You have existing, underutilized VMs that are already running other App Service instances.
- ⇒ You want to provide a custom image on which to run your functions.
- ⇒ Predictive scaling and costs are required.

Note: When you create a function app in Azure, you must choose a hosting plan for your app. There are three basic hosting plans available for Azure Functions:

**Topic 23 - Testlet 5**

A: A Premium plan would be more costly.

C: Need the VNET functionality.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-functions/functions-scale>

Question #1

## Introductory Info

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

LabelMaker app -

Coho Winery produces, bottles, and distributes a variety of wines globally. You are a developer implementing highly scalable and resilient applications to support online order processing by using Azure solutions.

Coho Winery has a LabelMaker application that prints labels for wine bottles. The application sends data to several printers. The application consists of five modules that run independently on virtual machines (VMs). Coho Winery plans to move the application to Azure and continue to support label creation.

External partners send data to the LabelMaker application to include artwork and text for custom label designs.

Requirements. Data -

You identify the following requirements for data management and manipulation:

Order data is stored as nonrelational JSON and must be queried using SQL.

Changes to the Order data must reflect immediately across all partitions. All reads to the Order data must fetch the most recent writes.

Requirements. Security -

You have the following security requirements:

Users of Coho Winery applications must be able to provide access to documents, resources, and applications to external partners.

External partners must use their own credentials and authenticate with their organization's identity management solution.

External partner logins must be audited monthly for application use by a user account administrator to maintain company compliance.

Storage of e-commerce application settings must be maintained in Azure Key Vault.

E-commerce application sign-ins must be secured by using Azure App Service authentication and Azure Active Directory (AAD).

Conditional access policies must be applied at the application level to protect company content.

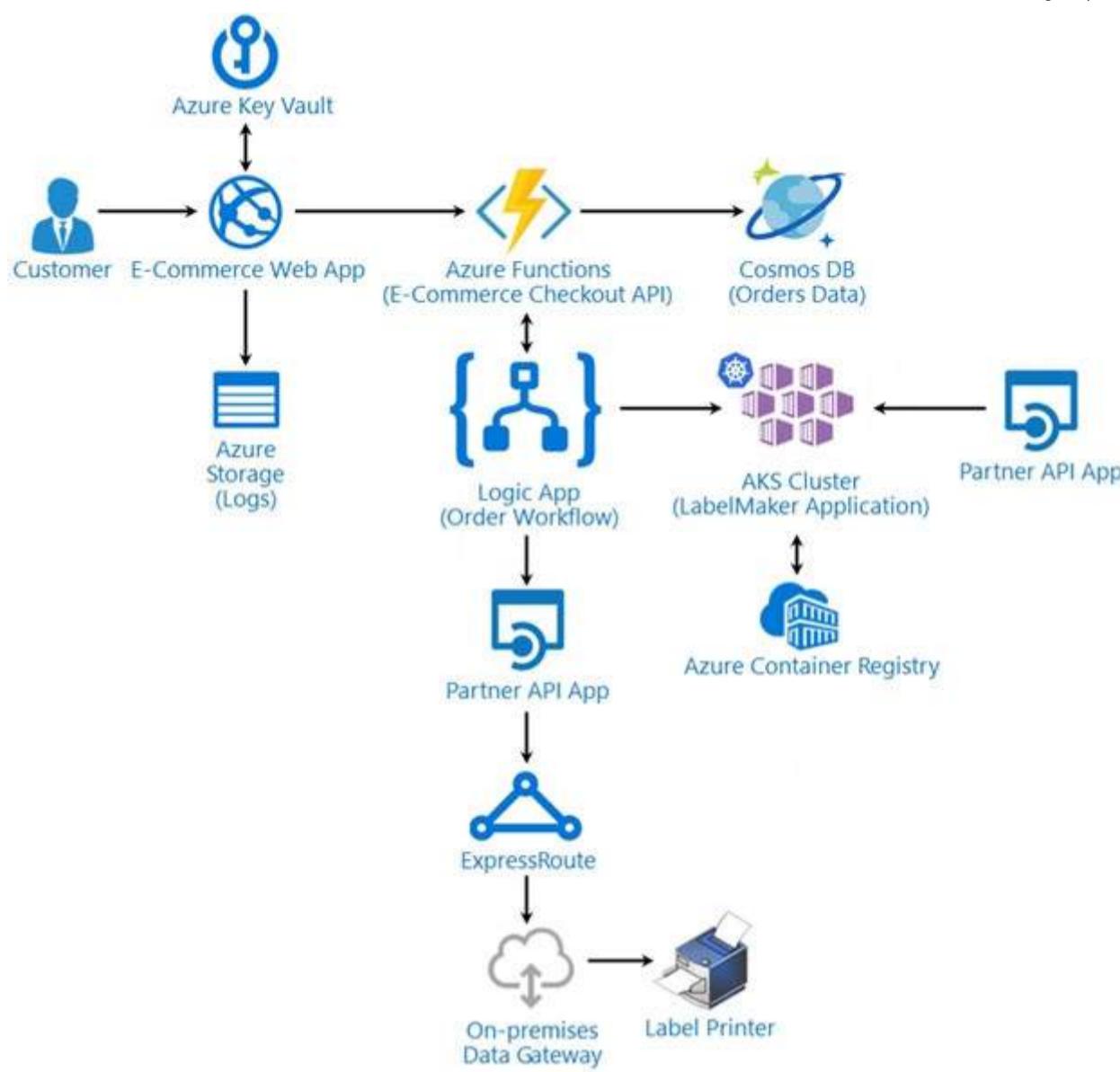
The LabelMaker application must be secured by using an AAD account that has full access to all namespaces of the Azure Kubernetes Service (AKS) cluster.

Requirements. LabelMaker app -

Azure Monitor Container Health must be used to monitor the performance of workloads that are deployed to Kubernetes environments and hosted on Azure Kubernetes Service (AKS).

You must use Azure Container Registry to publish images that support the AKS deployment.

Architecture -



#### Issues -

Calls to the Printer API App fail periodically due to printer communication timeouts.

Printer communication timeouts occur after 10 seconds. The label printer must only receive up to 5 attempts within one minute.

The order workflow fails to run upon initial deployment to Azure.

#### Order.json -

Relevant portions of the app files are shown below. Line numbers are included for reference only.

This JSON file contains a representation of the data for an order that includes a single item.

Order.json -

### Order.json

```
01 {
02   "id" : 1,
03   "customers" : [
04     {
05       "familyName" : "Doe",
06       "givenName" : "John",
07       "customerid" : 5
08     }
09   ],
10   "line_items" : [
11     {
12       "fulfillable_quantity" : 1,
13       "id": 6,
14       "price" : "199.99" ,
15       "product_id" : 7513594,
16       "quantity": 1,
17       "requires_shipping" : true ,
18       "sku": "SFC-342-N" ,
19       "title" : "Surface Go" ,
20       "vendor" : "Microsoft" ,
21       "name" : "Surface Go - 8GB" ,
22       "taxable" : true ,
23       "tax_lines" : [
24         {
25           "title" : "State Tax" ,
26           "price" : "3.98" ,
27           "rate" : 0.06
28         }
29       ],
30       "total_discount" : "5.00" ,
31       "discount_allocations" : [
32         {
33           "amount" : "5.00" ,
34           "discount_application_index" : 2
35         }
36       ]
37 }
```

### Question

DRAG DROP -

You need to deploy a new version of the LabelMaker application to ACR.

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

Log in to the registry and push image.

Create an alias of the image with a new build number.

Create an alias of the image with the fully qualified path to the registry.

Download the image to your local computer.

Build a new application image by using dockerfile.

### Answer area

Correct Answer:

## Actions

Create an alias of the image with a new build number.

Download the image to your local computer.

## Answer area

Build a new application image by using dockerfile.

Create an alias of the image with the fully qualified path to the registry.

Log in to the registry and push image.

Step 1: Build a new application image by using dockerfile

Step 2: Create an alias if the image with the fully qualified path to the registry

Before you can push the image to a private registry, you'll need to ensure a proper image name. This can be achieved using the docker tag command. For demonstration purpose, we'll use Docker's hello world image, rename it and push it to ACR.

# pulls hello-world from the public docker hub

\$ docker pull hello-world

# tag the image in order to be able to push it to a private registry

\$ docker tag hello-world <REGISTRY\_NAME>/hello-world

# push the image

\$ docker push <REGISTRY\_NAME>/hello-world

Step 3: Log in to the registry and push image

In order to push images to the newly created ACR instance, you need to login to ACR from the Docker CLI. Once logged in, you can push any existing docker image to your ACR instance.

Scenario:

Coho Winery plans to move the application to Azure and continue to support label creation.

LabelMaker app -

Azure Monitor Container Health must be used to monitor the performance of workloads that are deployed to Kubernetes environments and hosted on Azure

Kubernetes Service (AKS).

You must use Azure Container Registry to publish images that support the AKS deployment.

Reference:

<https://thorsten-hans.com/how-to-use-a-private-azure-container-registry-with-kubernetes-9b86e67b93b6> <https://docs.microsoft.com/en-us/azure/container-registry/container-registry-tutorial-quick-task>

Question #2

## Introductory Info

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

LabelMaker app -

Coho Winery produces, bottles, and distributes a variety of wines globally. You are a developer implementing highly scalable and resilient applications to support online order processing by using Azure solutions.

Coho Winery has a LabelMaker application that prints labels for wine bottles. The application sends data to several printers. The application consists of five modules that run independently on virtual machines (VMs). Coho Winery plans to move the application to Azure and continue to support label creation.

External partners send data to the LabelMaker application to include artwork and text for custom label designs.

Requirements. Data -

You identify the following requirements for data management and manipulation:

Order data is stored as nonrelational JSON and must be queried using SQL.

Changes to the Order data must reflect immediately across all partitions. All reads to the Order data must fetch the most recent writes.

Requirements. Security -

You have the following security requirements:

Users of Coho Winery applications must be able to provide access to documents, resources, and applications to external partners.

External partners must use their own credentials and authenticate with their organization's identity management solution.

External partner logins must be audited monthly for application use by a user account administrator to maintain company compliance.

Storage of e-commerce application settings must be maintained in Azure Key Vault.

E-commerce application sign-ins must be secured by using Azure App Service authentication and Azure Active Directory (AAD).

Conditional access policies must be applied at the application level to protect company content.

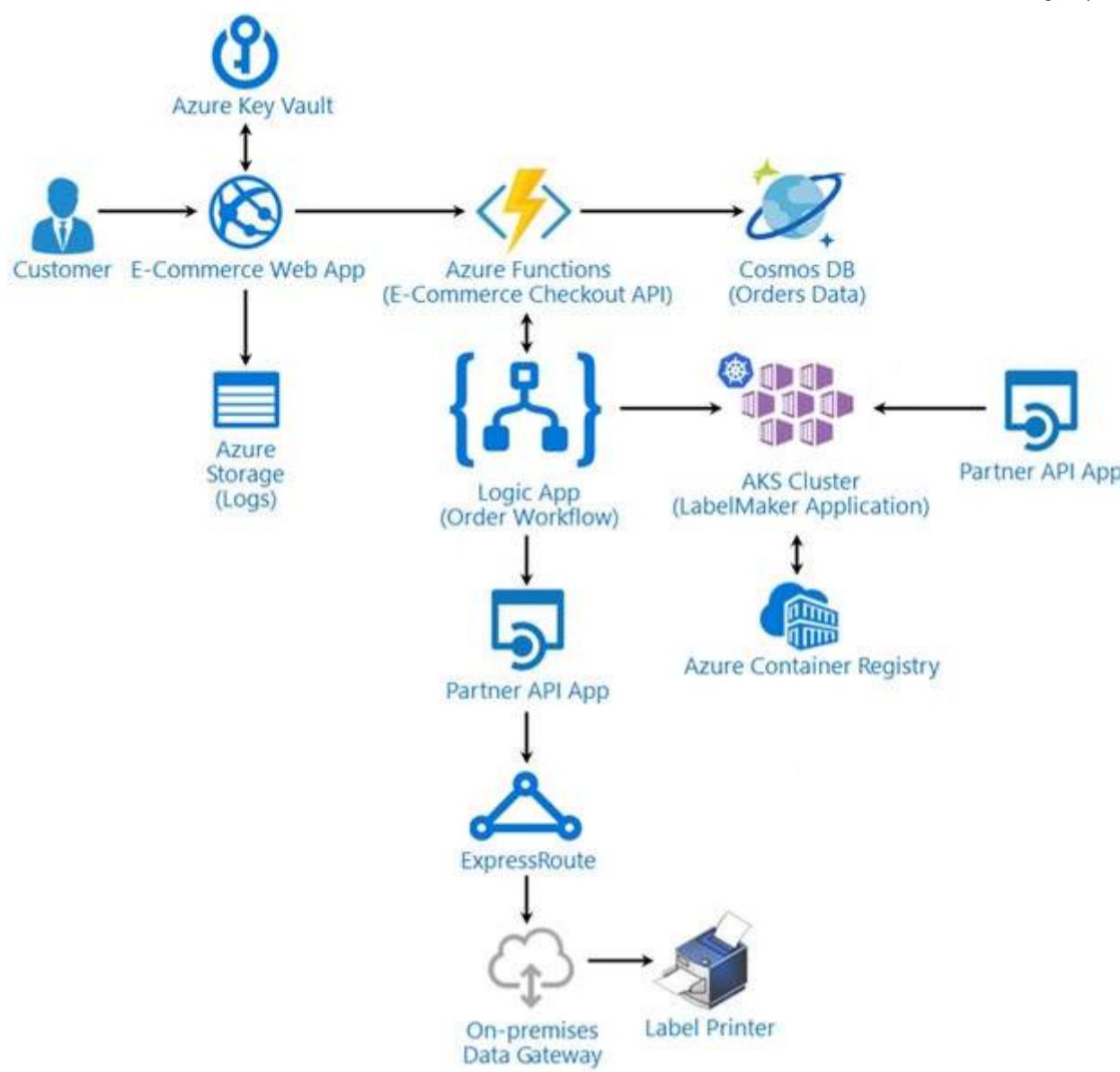
The LabelMaker application must be secured by using an AAD account that has full access to all namespaces of the Azure Kubernetes Service (AKS) cluster.

Requirements. LabelMaker app -

Azure Monitor Container Health must be used to monitor the performance of workloads that are deployed to Kubernetes environments and hosted on Azure Kubernetes Service (AKS).

You must use Azure Container Registry to publish images that support the AKS deployment.

Architecture -



#### Issues -

Calls to the Printer API App fail periodically due to printer communication timeouts.

Printer communication timeouts occur after 10 seconds. The label printer must only receive up to 5 attempts within one minute.

The order workflow fails to run upon initial deployment to Azure.

#### Order.json -

Relevant portions of the app files are shown below. Line numbers are included for reference only.

This JSON file contains a representation of the data for an order that includes a single item.

Order.json -

### Order.json

```
01 {
02   "id" : 1,
03   "customers" : [
04     {
05       "familyName" : "Doe",
06       "givenName" : "John",
07       "customerid" : 5
08     }
09   ],
10   "line_items" : [
11     {
12       "fulfillable_quantity" : 1,
13       "id": 6,
14       "price" : "199.99" ,
15       "product_id" : 7513594,
16       "quantity": 1,
17       "requires_shipping" : true ,
18       "sku": "SFC-342-N" ,
19       "title" : "Surface Go" ,
20       "vendor" : "Microsoft" ,
21       "name" : "Surface Go - 8GB" ,
22       "taxable" : true ,
23       "tax_lines" : [
24         {
25           "title" : "State Tax" ,
26           "price" : "3.98" ,
27           "rate" : 0.06
28         }
29       ],
30       "total_discount" : "5.00" ,
31       "discount_allocations" : [
32         {
33           "amount" : "5.00" ,
34           "discount_application_index" : 2
35         }
36       ]
37 }
```

### Question

You need to access data from the user claim object in the e-commerce web app.

What should you do first?

- A. Write custom code to make a Microsoft Graph API call from the e-commerce web app.
- B. Assign the Contributor RBAC role to the e-commerce web app by using the Resource Manager create role assignment API.
- C. Update the e-commerce web app to read the HTTP request header values.
- D. Using the Azure CLI, enable Cross-origin resource sharing (CORS) from the e-commerce checkout API to the e-commerce web app.

### Correct Answer: C

Methods to Get User Identity and Claims in a .NET Azure Functions App include:

ClaimsPrincipal from the Request Context

The ClaimsPrincipal object is also available as part of the request context and can be extracted from the HttpRequest.HttpContext.

User Claims from the Request Headers.

App Services passes user claims to the app by using special request headers.

### Topic 24 - Testlet 6

<https://levelup.gitconnected.com/four-alternative-methods-to-get-user-identity-and-claims-in-a-net-azure-functions-app-df98c40424bb>

Question #1

## Introductory Info

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

Overview -

You are a developer for Contoso, Ltd. The company has a social networking website that is developed as a Single Page Application (SPA). The main web application for the social networking website loads user uploaded content from blob storage.

You are developing a solution to monitor uploaded data for inappropriate content. The following process occurs when users upload content by using the SPA:

- \* Messages are sent to ContentUploadService.
- \* Content is processed by ContentAnalysisService.
- \* After processing is complete, the content is posted to the social network or a rejection message is posted in its place.

The ContentAnalysisService is deployed with Azure Container Instances from a private Azure Container Registry named contosoimages.

The solution will use eight CPU cores.

Azure Active Directory -

Contoso, Ltd. uses Azure Active Directory (Azure AD) for both internal and guest accounts.

Requirements -

ContentAnalysisService -

The company's data science group built ContentAnalysisService which accepts user generated content as a string and returns a probable value for inappropriate content. Any values over a specific threshold must be reviewed by an employee of Contoso, Ltd.

You must create an Azure Function named CheckUserContent to perform the content checks.

Costs -

You must minimize costs for all Azure services.

Manual review -

To review content, the user must authenticate to the website portion of the ContentAnalysisService using their Azure AD credentials. The website is built using

React and all pages and API endpoints require authentication. In order to review content a user must be part of a ContentReviewer role. All completed reviews must include the reviewer's email address for auditing purposes.

High availability -

All services must run in multiple regions. The failure of any service in a region must not impact overall application availability.

Monitoring -

An alert must be raised if the ContentUploadService uses more than 80 percent of available CPU cores.

Security -

You have the following security requirements:

- Any web service accessible over the Internet must be protected from cross site scripting attacks.
- All websites and services must use SSL from a valid root certificate authority.
- Azure Storage access keys must only be stored in memory and must be available only to the service.
- All Internal services must only be accessible from internal Virtual Networks (VNets).
- All parts of the system must support inbound and outbound traffic restrictions.
- All service calls must be authenticated by using Azure AD.

User agreements -

When a user submits content, they must agree to a user agreement. The agreement allows employees of Contoso, Ltd. to review content, store cookies on user devices, and track user's IP addresses.

Information regarding agreements is used by multiple divisions within Contoso, Ltd.

User responses must not be lost and must be available to all parties regardless of individual service uptime. The volume of agreements is expected to be in the millions per hour.

Validation testing -

When a new version of the ContentAnalysisService is available the previous seven days of content must be processed with the new version to verify that the new version does not significantly deviate from the old version.

Issues -

Users of the ContentUploadService report that they occasionally see HTTP 502 responses on specific pages.

Code -

ContentUploadService -

```
CS01 apiVersion: '2018-10-01'
CS02 type: Microsoft.ContainerInstance/containerGroups
CS03 location: westus
CS04 name: contentUploadService
CS05 properties:
CS06   containers:
CS07     - name: service
CS08       properties:
CS09         image: contoso/contentUploadService:latest
CS10       ports:
CS11         - port: 80
CS12           protocol: TCP
CS13       resources:
CS14         requests:
CS15           cpu: 1.0
CS16           memoryInGB: 1.5
CS17
CS18 ipAddress:
CS19   ip: 10.23.121.112
CS20   ports:
CS21     - port: 80
CS22       protocol: TCP
CS23
CS24
CS25 networkProfile:
CS26
id: /subscriptions/98...19/resourceGroups/container/providers/Microsoft.Network/networkProfiles/subnet
```

**ApplicationManifest -**

```
AM01 {  
AM02     "id" : "2b079f03-9b06-2d44-98bb-e9182901fcb6",  
AM03     "appId" : "7118a7f0-b5c2-4c9d-833c-3d711396fe65",  
AM04  
AM05     "createdDateTime" : "2019-12-24T06:01:44Z",  
AM06     "logoUrl" : null,  
AM07     "logoutUrl" : null,  
AM08     "name" : "ContentAnalysisService",  
AM09  
AM10  
AM11     "orgRestrictions" : [],  
AM12     "parentalControlSettings" : {  
AM13         "countriesBlockedForMinors" : [],  
AM14         "legalAgeGroupRule" : "Allow"  
AM15     },  
AM16     "passwordCredentials" : []  
AM17 }
```

**Question**

You need to configure the ContentUploadService deployment.

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

NOTE: Each correct selection is worth one point.

- A. Add the following markup to line CS23: type: Private
- B. Add the following markup to line CS24: osType: Windows
- C. Add the following markup to line CS24: osType: Linux
- D. Add the following markup to line CS23: type: Public

**Correct Answer:** A 

Scenario: All Internal services must only be accessible from Internal Virtual Networks (VNets)

There are three Network Location types – Private, Public and Domain

Reference:

<https://devblogs.microsoft.com/powershell/setting-network-location-to-private/>

Question #2

## Introductory Info

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

Overview -

You are a developer for Contoso, Ltd. The company has a social networking website that is developed as a Single Page Application (SPA). The main web application for the social networking website loads user uploaded content from blob storage.

You are developing a solution to monitor uploaded data for inappropriate content. The following process occurs when users upload content by using the SPA:

- \* Messages are sent to ContentUploadService.
- \* Content is processed by ContentAnalysisService.
- \* After processing is complete, the content is posted to the social network or a rejection message is posted in its place.

The ContentAnalysisService is deployed with Azure Container Instances from a private Azure Container Registry named contosoimages.

The solution will use eight CPU cores.

Azure Active Directory -

Contoso, Ltd. uses Azure Active Directory (Azure AD) for both internal and guest accounts.

Requirements -

ContentAnalysisService -

The company's data science group built ContentAnalysisService which accepts user generated content as a string and returns a probable value for inappropriate content. Any values over a specific threshold must be reviewed by an employee of Contoso, Ltd.

You must create an Azure Function named CheckUserContent to perform the content checks.

Costs -

You must minimize costs for all Azure services.

Manual review -

To review content, the user must authenticate to the website portion of the ContentAnalysisService using their Azure AD credentials. The website is built using

React and all pages and API endpoints require authentication. In order to review content a user must be part of a ContentReviewer role. All completed reviews must include the reviewer's email address for auditing purposes.

High availability -

All services must run in multiple regions. The failure of any service in a region must not impact overall application availability.

Monitoring -

An alert must be raised if the ContentUploadService uses more than 80 percent of available CPU cores.

Security -

You have the following security requirements:

- Any web service accessible over the Internet must be protected from cross site scripting attacks.
- All websites and services must use SSL from a valid root certificate authority.
- Azure Storage access keys must only be stored in memory and must be available only to the service.
- All Internal services must only be accessible from internal Virtual Networks (VNets).
- All parts of the system must support inbound and outbound traffic restrictions.
- All service calls must be authenticated by using Azure AD.

User agreements -

When a user submits content, they must agree to a user agreement. The agreement allows employees of Contoso, Ltd. to review content, store cookies on user devices, and track user's IP addresses.

Information regarding agreements is used by multiple divisions within Contoso, Ltd.

User responses must not be lost and must be available to all parties regardless of individual service uptime. The volume of agreements is expected to be in the millions per hour.

Validation testing -

When a new version of the ContentAnalysisService is available the previous seven days of content must be processed with the new version to verify that the new version does not significantly deviate from the old version.

Issues -

Users of the ContentUploadService report that they occasionally see HTTP 502 responses on specific pages.

Code -

ContentUploadService -

```
CS01 apiVersion: '2018-10-01'
CS02 type: Microsoft.ContainerInstance/containerGroups
CS03 location: westus
CS04 name: contentUploadService
CS05 properties:
CS06   containers:
CS07     - name: service
CS08       properties:
CS09         image: contoso/contentUploadService:latest
CS10       ports:
CS11         - port: 80
CS12           protocol: TCP
CS13       resources:
CS14         requests:
CS15           cpu: 1.0
CS16           memoryInGB: 1.5
CS17
CS18 ipAddress:
CS19   ip: 10.23.121.112
CS20   ports:
CS21     - port: 80
CS22       protocol: TCP
CS23
CS24
CS25 networkProfile:
CS26
id: /subscriptions/98...19/resourceGroups/container/providers/Microsoft.Network/networkProfiles/subnet
```

**ApplicationManifest -**

```
AM01 {  
AM02     "id" : "2b079f03-9b06-2d44-98bb-e9182901fcb6",  
AM03     "appId" : "7118a7f0-b5c2-4c9d-833c-3d711396fe65",  
AM04  
AM05     "createdDateTime" : "2019-12-24T06:01:44Z",  
AM06     "logoUrl" : null,  
AM07     "logoutUrl" : null,  
AM08     "name" : "ContentAnalysisService",  
AM09  
AM10  
AM11     "orgRestrictions" : [],  
AM12     "parentalControlSettings" : {  
AM13         "countriesBlockedForMinors" : [],  
AM14         "legalAgeGroupRule" : "Allow"  
AM15     },  
AM16     "passwordCredentials" : []  
AM17 }
```

**Question**

You need to store the user agreements.

Where should you store the agreement after it is completed?

- A. Azure Storage queue
- B. Azure Event Hub
- C. Azure Service Bus topic
- D. Azure Event Grid topic

**Correct Answer:** *B* 

Azure Event Hub is used for telemetry and distributed data streaming.

This service provides a single solution that enables rapid data retrieval for real-time processing as well as repeated replay of stored raw data. It can capture the streaming data into a file for processing and analysis.

It has the following characteristics:

- low latency
- capable of receiving and processing millions of events per second
- 
- at least once delivery

Reference:

<https://docs.microsoft.com/en-us/azure/event-grid/compare-messaging-services>

Question #3

## Introductory Info

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

Overview -

You are a developer for Contoso, Ltd. The company has a social networking website that is developed as a Single Page Application (SPA). The main web application for the social networking website loads user uploaded content from blob storage.

You are developing a solution to monitor uploaded data for inappropriate content. The following process occurs when users upload content by using the SPA:

- \* Messages are sent to ContentUploadService.
- \* Content is processed by ContentAnalysisService.
- \* After processing is complete, the content is posted to the social network or a rejection message is posted in its place.

The ContentAnalysisService is deployed with Azure Container Instances from a private Azure Container Registry named contosoimages.

The solution will use eight CPU cores.

Azure Active Directory -

Contoso, Ltd. uses Azure Active Directory (Azure AD) for both internal and guest accounts.

Requirements -

ContentAnalysisService -

The company's data science group built ContentAnalysisService which accepts user generated content as a string and returns a probable value for inappropriate content. Any values over a specific threshold must be reviewed by an employee of Contoso, Ltd.

You must create an Azure Function named CheckUserContent to perform the content checks.

Costs -

You must minimize costs for all Azure services.

Manual review -

To review content, the user must authenticate to the website portion of the ContentAnalysisService using their Azure AD credentials. The website is built using

React and all pages and API endpoints require authentication. In order to review content a user must be part of a ContentReviewer role. All completed reviews must include the reviewer's email address for auditing purposes.

High availability -

All services must run in multiple regions. The failure of any service in a region must not impact overall application availability.

Monitoring -

An alert must be raised if the ContentUploadService uses more than 80 percent of available CPU cores.

Security -

You have the following security requirements:

Any web service accessible over the Internet must be protected from cross site scripting attacks.

All websites and services must use SSL from a valid root certificate authority.

Azure Storage access keys must only be stored in memory and must be available only to the service.

All Internal services must only be accessible from internal Virtual Networks (VNets).

All parts of the system must support inbound and outbound traffic restrictions.

All service calls must be authenticated by using Azure AD.

User agreements -

When a user submits content, they must agree to a user agreement. The agreement allows employees of Contoso, Ltd. to review content, store cookies on user devices, and track user's IP addresses.

Information regarding agreements is used by multiple divisions within Contoso, Ltd.

User responses must not be lost and must be available to all parties regardless of individual service uptime. The volume of agreements is expected to be in the millions per hour.

Validation testing -

When a new version of the ContentAnalysisService is available the previous seven days of content must be processed with the new version to verify that the new version does not significantly deviate from the old version.

Issues -

Users of the ContentUploadService report that they occasionally see HTTP 502 responses on specific pages.

Code -

ContentUploadService -

```
CS01 apiVersion: '2018-10-01'
CS02 type: Microsoft.ContainerInstance/containerGroups
CS03 location: westus
CS04 name: contentUploadService
CS05 properties:
CS06   containers:
CS07     - name: service
CS08       properties:
CS09         image: contoso/contentUploadService:latest
CS10       ports:
CS11         - port: 80
CS12           protocol: TCP
CS13       resources:
CS14         requests:
CS15           cpu: 1.0
CS16           memoryInGB: 1.5
CS17
CS18 ipAddress:
CS19   ip: 10.23.121.112
CS20   ports:
CS21     - port: 80
CS22       protocol: TCP
CS23
CS24
CS25 networkProfile:
CS26
id: /subscriptions/98...19/resourceGroups/container/providers/Microsoft.Network/networkProfiles/subnet
```

**ApplicationManifest -**

```
AM01 {  
AM02     "id" : "2b079f03-9b06-2d44-98bb-e9182901fcb6",  
AM03     "appId" : "7118a7f0-b5c2-4c9d-833c-3d711396fe65",  
AM04  
AM05     "createdDateTime" : "2019-12-24T06:01:44Z",  
AM06     "logoUrl" : null,  
AM07     "logoutUrl" : null,  
AM08     "name" : "ContentAnalysisService",  
AM09  
AM10  
AM11     "orgRestrictions" : [],  
AM12     "parentalControlSettings" : {  
AM13         "countriesBlockedForMinors" : [],  
AM14         "legalAgeGroupRule" : "Allow"  
AM15     },  
AM16     "passwordCredentials" : []  
AM17 }
```

**Question****HOTSPOT -**

You need to implement the bindings for the CheckUserContent function.

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

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

```
public static class CheckUserContent  
{  
    [FunctionName("CheckUserContent")]  
    public static void Run(  
        [QueueTrigger("userContent")] string content,  
        [BlobTrigger("userContent/{name}")] Stream output)  
    {  
        ...  
    }  
}
```

[QueueTrigger("userContent")] [BlobTrigger("userContent/{name}")] [CosmosDBTrigger("content", "userContent")] [Table("content", "userContent", "{name}")]	▼	string content,
[Queue("userContent")] [CosmosDB("content", "userContent")] [Table("content", "userContent", "{name}")] [Blob("userContent/{name}", FileAccess.Write)]	▼	Stream output)

## Answer Area

```
public static class CheckUserContent
{
    [FunctionName("CheckUserContent")]
    public static void Run(
        [QueueTrigger("userContent")]
        [BlobTrigger("userContent/{name}")]
        [CosmosDBTrigger("content", "userContent")]
        [Table("content", "userContent", "{name}")]
        string content,
        Stream output)
    {
        ...
    }
}
```

Correct Answer:

Box 1: [BlobTrigger(..)]

Box 2: [Blob(..)]

Azure Blob storage output binding for Azure Functions. The output binding allows you to modify and delete blob storage data in an Azure Function.

### Topic 25 - Testlet 7

```
public static void Run(
    [BlobTrigger("sample-images/{name}")] Stream image,
    [Blob("sample-images-md/{name}", FileAccess.Write)] Stream imageSmall)
{
    ...
}
```

Scenario: You must create an Azure Function named CheckUserContent to perform the content checks.

The company's data science group built ContentAnalysisService which accepts user generated content as a string and returns a probable value for inappropriate content. Any values over a specific threshold must be reviewed by an employee of Contoso, Ltd.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-functions/functions-bindings-storage-blob-output>

Question #1

**Introductory Info**

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

Background -

City Power & Light company provides electrical infrastructure monitoring solutions for homes and businesses. The company is migrating solutions to Azure.

Current environment -

Architecture overview -

The company has a public website located at <http://www.cpndl.com/>. The site is a single-page web application that runs in Azure App Service on Linux. The website uses files stored in Azure Storage and cached in Azure Content Delivery Network (CDN) to serve static content.

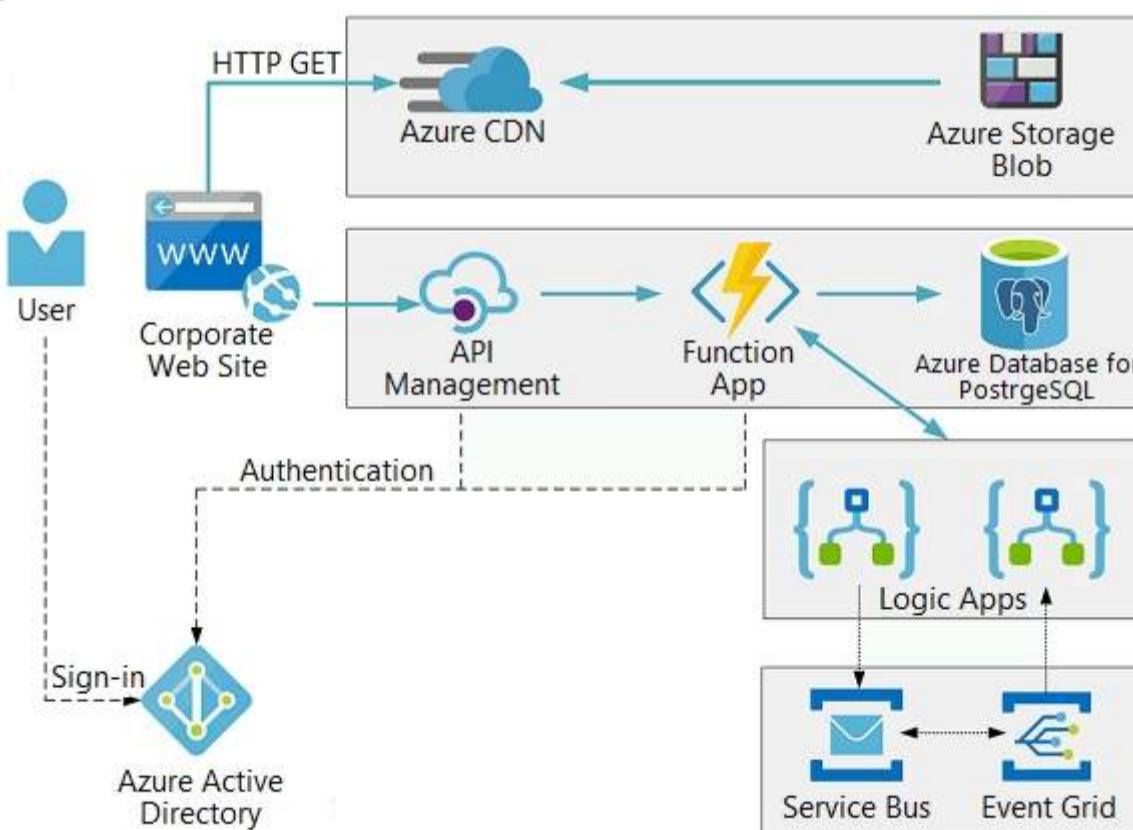
API Management and Azure Function App functions are used to process and store data in Azure Database for PostgreSQL. API Management is used to broker communications to the Azure Function app functions for Logic app integration. Logic apps are used to orchestrate the data processing while Service Bus and

Event Grid handle messaging and events.

The solution uses Application Insights, Azure Monitor, and Azure Key Vault.

Architecture diagram -

The company has several applications and services that support their business. The company plans to implement serverless computing where possible. The overall architecture is shown below.



User authentication -

The following steps detail the user authentication process:

1. The user selects Sign in in the website.
2. The browser redirects the user to the Azure Active Directory (Azure AD) sign in page.

3. The user signs in.
4. Azure AD redirects the user's session back to the web application. The URL includes an access token.
5. The web application calls an API and includes the access token in the authentication header. The application ID is sent as the audience ('`aud`') claim in the access token.
6. The back-end API validates the access token.

Requirements -

Corporate website -

Communications and content must be secured by using SSL.

Communications must use HTTPS.

Data must be replicated to a secondary region and three availability zones.

Data storage costs must be minimized.

Azure Database for PostgreSQL -

The database connection string is stored in Azure Key Vault with the following attributes:

Azure Key Vault name: cpandlkeyvault

Secret name: PostgreSQLConn

Id: 80df3e46ffcd4f1cb187f79905e9a1e8

The connection information is updated frequently. The application must always use the latest information to connect to the database.

Azure Service Bus and Azure Event Grid

Azure Event Grid must use Azure Service Bus for queue-based load leveling.

Events in Azure Event Grid must be routed directly to Service Bus queues for use in buffering.

Events from Azure Service Bus and other Azure services must continue to be routed to Azure Event Grid for processing.

Security -

All SSL certificates and credentials must be stored in Azure Key Vault.

File access must restrict access by IP, protocol, and Azure AD rights.

All user accounts and processes must receive only those privileges which are essential to perform their intended function.

Compliance -

Auditing of the file updates and transfers must be enabled to comply with General Data Protection Regulation (GDPR). The file updates must be read-only, stored in the order in which they occurred, include only create, update, delete, and copy operations, and be retained for compliance reasons.

Issues -

Corporate website -

While testing the site, the following error message displays:

CryptographicException: The system cannot find the file specified.

Function app -

You perform local testing for the RequestUserApproval function. The following error message displays:

'Timeout value of 00:10:00 exceeded by function: RequestUserApproval'

The same error message displays when you test the function in an Azure development environment when you run the following Kusto query:

FunctionAppLogs -

```
| where FunctionName == "RequestUserApproval"
```

Logic app -

You test the Logic app in a development environment. The following error message displays:

'400 Bad Request'

Troubleshooting of the error shows an HttpTrigger action to call the RequestUserApproval function.

Code -

Corporate website -

Security.cs:

```
SC01 public class Security
SC02 {
SC03     var bytes = System.IO.File.ReadAllBytes("~/var/ssl/private");
SC04     var cert = new System.Security.Cryptography.X509Certificate2(bytes);
SC05     var certName = cert.FriendlyName;
SC06 }
```

Function app -

RequestUserApproval.cs:

```
RA01 public static class RequestUserApproval
RA02 {
RA03     [FunctionName("RequestUserApproval")]
RA04     public static async Task<IActionResult> Run(
RA05         [HttpTrigger(AuthorizationLevel.Function, "get", "post", Route = null)] HttpRequest req,
RA06         ILogger log)
RA06     {
RA07         log.LogInformation("RequestUserApproval function processed a request.");
RA08     ...
RA09     return ProcessRequest(req)
RA10     ? (ActionResult)new OkObjectResult($"User approval processed")
RA11     : new BadRequestObjectResult("Failed to process user approval");
RA12 }
RA13 private static bool ProcessRequest(HttpContext req)
RA14 {
RA15     ...
RA16 }
RA17 }
```

### Question

HOTSPOT -

You need to configure the Account Kind, Replication, and Storage tier options for the corporate website's Azure Storage account.

How should you complete the configuration? To answer, select the appropriate options in the dialog box in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

## Create storage account

Basics Networking Advanced Tags Review + create

Azure Storage is a Microsoft-managed service providing cloud storage that is highly available, secure, durable, scalable, and redundant. Azure Storage includes Azure Blobs (objects), Azure Data Lake Storage Gen2, Azure Files, Azure Queues, and Azure Tables. The cost of your storage account depends on the usage and the options you choose below.

### Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription \*

Visual Studio Enterprise

Resource group \*

(New) cplcorporatesite

[Create new](#)

### Instance details

The default deployment model is Resource Manager, which supports the latest Azure features. You may choose to deploy using the classic deployment model instead. [Choose classic deployment model](#)

Storage account name ⓘ \*

corporatewebsitecontent

Location \*

(US) East US

Performance ⓘ

Standard  Premium

Account kind ⓘ

StorageV2 (general purpose v2)  
Storage (general purpose v1)  
BlobStorage

Replication ⓘ

Locally-redundant storage (LRS)  
Zone-redundant storage (ZRS)  
Geo-redundant storage (GRS)  
Read-access geo-redundant storage (RA-GRS)  
Geo-zone-redundant storage (GZRS)  
Read-access geo-zone-redundant storage (RA-GZRS)

Access tier (default) ⓘ

Cool  Hot

**Correct Answer:**

## Create storage account

[Basics](#)   [Networking](#)   [Advanced](#)   [Tags](#)   [Review + create](#)

Azure Storage is a Microsoft-managed service providing cloud storage that is highly available, secure, durable, scalable, and redundant. Azure Storage includes Azure Blobs (objects), Azure Data Lake Storage Gen2, Azure Files, Azure Queues, and Azure Tables. The cost of your storage account depends on the usage and the options you choose below.

### Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *	Visual Studio Enterprise
Resource group *	(New) cplcorporatesite
	<a href="#">Create new</a>

### Instance details

The default deployment model is Resource Manager, which supports the latest Azure features. You may choose to deploy using the classic deployment model instead. [Choose classic deployment model](#)

Storage account name ⓘ *	corporatewebsitecontent
Location *	(US) East US
Performance ⓘ	<input checked="" type="radio"/> Standard <input type="radio"/> Premium
Account kind ⓘ	StorageV2 (general purpose v2) Storage (general purpose v1) BlobStorage
Replication ⓘ	Locally-redundant storage (LRS) Zone-redundant storage (ZRS) Geo-redundant storage (GRS) Read-access geo-redundant storage (RA-GRS) Geo-zone-redundant storage (GZRS) Read-access geo-zone-redundant storage (RA-GZRS)

Access tier (default) ⓘ

Cool

Hot

Account Kind: StorageV2 (general-purpose v2)

Scenario: Azure Storage blob will be used (refer to the exhibit). Data storage costs must be minimized.

### Topic 26 - Testlet 8

Storage.

Incorrect Answers:

- ☞ Block Blob Storage accounts: Storage accounts with premium performance characteristics for block blobs and append blobs. Recommended for scenarios with high transaction rates, or scenarios that use smaller objects or require consistently low storage latency.
- ☞ General-purpose v1 accounts: Legacy account type for blobs, files, queues, and tables. Use general-purpose v2 accounts instead when possible.

Replication: Geo-redundant Storage

Scenario: Data must be replicated to a secondary region and three availability zones.

Geo-redundant storage (GRS) copies your data synchronously three times within a single physical location in the primary region using LRS. It then copies your data asynchronously to a single physical location in the secondary region.

Question #1

Topic 26

## Introductory Info

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 questions in 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 section 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 in 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. 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 SaaS company that provides a solution for managing employee expenses. The solution consists of an ASP.NET Core Web API project that is deployed as an Azure Web App.

Overall architecture -

Employees upload receipts for the system to process. When processing is complete, the employee receives a summary report email that details the processing results. Employees then use a web application to manage their receipts and perform any additional tasks needed for reimbursement.

Receipt processing -

Employees may upload receipts in two ways:

Uploading using an Azure Files mounted folder

Uploading using the web application

Data Storage -

Receipt and employee information is stored in an Azure SQL database.

Documentation -

Employees are provided with a getting started document when they first use the solution. The documentation includes details on supported operating systems for

Azure File upload, and instructions on how to configure the mounted folder.

Solution details -

Users table -

Column	Description
UserId	unique identifier for an employee
ExpenseAccount	employees expense account number in the format 1234-123-1234
AllowedAmount	limit of allowed expenses before approval is needed
SupervisorId	unique identifier for employee's supervisor
SecurityPin	value used to validate user identity

Web Application -

You enable MSI for the Web App and configure the Web App to use the security principal name WebAppIdentity.

Processing -

Processing is performed by an Azure Function that uses version 2 of the Azure Function runtime. Once processing is completed, results are stored in Azure Blob

Storage and an Azure SQL database. Then, an email summary is sent to the user with a link to the processing report. The link to the report must

remain valid if the email is forwarded to another user.

#### Logging -

Azure Application Insights is used for telemetry and logging in both the processor and the web application. The processor also has TraceWriter logging enabled.

Application Insights must always contain all log messages.

#### Requirements -

##### Receipt processing -

Concurrent processing of a receipt must be prevented.

##### Disaster recovery -

Regional outage must not impact application availability. All DR operations must not be dependent on application running and must ensure that data in the DR region is up to date.

#### Security -

User's SecurityPin must be stored in such a way that access to the database does not allow the viewing of SecurityPins. The web application is the only system that should have access to SecurityPins.

All certificates and secrets used to secure data must be stored in Azure Key Vault.

You must adhere to the principle of least privilege and provide privileges which are essential to perform the intended function.

All access to Azure Storage and Azure SQL database must use the application's Managed Service Identity (MSI).

Receipt data must always be encrypted at rest.

All data must be protected in transit.

User's expense account number must be visible only to logged in users. All other views of the expense account number should include only the last segment, with the remaining parts obscured.

In the case of a security breach, access to all summary reports must be revoked without impacting other parts of the system.

#### Issues -

##### Upload format issue -

Employees occasionally report an issue with uploading a receipt using the web application. They report that when they upload a receipt using the Azure File Share, the receipt does not appear in their profile. When this occurs, they delete the file in the file share and use the web application, which returns a 500 Internal Server error page.

##### Capacity issue -

During busy periods, employees report long delays between the time they upload the receipt and when it appears in the web application.

##### Log capacity issue -

Developers report that the number of log messages in the trace output for the processor is too high, resulting in lost log messages.

#### Application code -

##### Processing.cs -

```

PC01 public static class Processing
PC02 {
PC03     public static class Function
PC04     {
PC05         [FunctionName("IssueWork")]
PC06         public static async Task Run([TimerTrigger("0 */5 * * * *")] TimerInfo timer, ILogger
log)
PC07         {
PC08             var container = await GetCloudBlobContainer();
PC09             foreach (var fileItem in await ListFiles())
PC10             {
PC11                 var file = new CloudFile(fileItem.StorageUri.PrimaryUri);
PC12                 var ms = new MemoryStream();
PC13                 await file.DownloadToStreamAsync(ms);
PC14                 var blob = container.GetBlockBlobReference(fileItem.Uri.ToString());
PC15                 await blob.UploadFromStreamAsync(ms);
PC16
PC17             }
PC18         }
PC19         private static CloudBlockBlob GetDRBlob(CloudBlockBlob sourceBlob)
PC20         {
PC21             . .
PC22         }
PC23         private static async Task<CloudBlobContainer> GetCloudBlobContainer()
PC24         {
PC25             var cloudBlobClient = new CloudBlobClient(new Uri(" . . ."), await GetCredentials());
PC26
PC27             await cloudBlobClient.GetRootContainerReference().CreateIfNotExistsAsync();
PC28             return cloudBlobClient.GetRootContainerReference();
PC29         }
PC30         private static async Task<StorageCredentials> GetCredentials()
PC31         {
PC32             . .
PC33         }
PC34         private static async Task<List<IListFileItem>> ListFiles()
PC35         {
PC36             . .
PC37         }
PC37         private KeyVaultClient _keyVaultClient = new KeyVaultClient(" . . .");
PC38     }
PC39 }

```

Database.cs -

```

DB01 public class Database
DB02 {
DB03     private string ConnectionString =
DB04
DB05     public async Task<object> LoadUserDetails(string userId)
DB06     {
DB07
DB08         return await policy.ExecuteAsync(async () =>
DB09         {
DB10             using (var connection = new SqlConnection(ConnectionString))
DB11             {
DB12                 await connection.OpenAsync();
DB13                 using (var command = new SqlCommand(" . . .", connection))
DB14                     using (var reader = command.ExecuteReader())
DB15                     {
DB16                         . .
DB17                     }
DB18                 }
DB19             });
DB20         }
DB21     }

```

ReceiptUploader.cs -

```

RU01 public class ReceiptUploader
RU02 {
RU03     public async Task UploadFile(string file, byte[] binary)
RU04     {
RU05         var httpClient = new HttpClient();
RU06         var response = await httpClient.PutAsync("...", new ByteArrayContent(binary));
RU07         while (ShouldRetry(response))
RU08         {
RU09             response = await httpClient.PutAsync("...", new ByteArrayContent(binary));
RU10         }
RU11     }
RU12     private bool ShouldRetry(HttpStatusCode response)
RU13     {
RU14     }
RU15 }
RU16 }
```

ConfigureSSE.ps1 -

```

CS01 $storageAccount = Get-AzureRmStorageAccount -ResourceGroupName "..." -AccountName "..."
CS02 $keyVault = Get-AzureRmKeyVault -VaultName "..."
CS03 $key = Get-AzureKeyVaultKey -VaultName $keyVault.VaultName -Name "..."
CS04 Set-AzureRmKeyVaultAccessPolicy ` 
CS05 -VaultName $keyVault.VaultName ` 
CS06 -ObjectId $storageAccount.Identity.PrincipalId ` 
CS07 
CS08 
CS09 Set-AzureRmStorageAccount ` 
CS10 -ResourceGroupName $storageAccount.ResourceGroupName ` 
CS11 -AccountName $storageAccount.StorageAccountName ` 
CS12 -EnableEncryptionService File ` 
CS13 -KeyvaultEncryption ` 
CS14 -KeyName $key.Name 
CS15 -KeyVersion $key.Version ` 
CS16 -KeyVaultUri $keyVault.VaultUri
```

### Question

DRAG DROP -

You need to add code at line PC32 in Processing.cs to implement the GetCredentials method in the Processing class.

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 one point.

Select and Place:

#### Code segments

```

MSITokenProvider("...", null)
tp.GetAccessTokenAsync("...")

AzureServiceTokenProvider()

StringTokenProvider("storage", "msi")
tp.GetAuthenticationHeaderAsync(CancellationToken.None)
```

#### Answer Area

```

var tp = new _____ code segment
var t = new TokenCredential(await _____ code segment );
return new StorageCredentials(t);
```

**Correct Answer:****Code segments**

```
MSITokenProvider("...", null)  
  
StringTokenProvider("storage", "msi")  
  
tp.GetAuthenticationHeaderAsync(CancellationToken.None)
```

**Answer Area**

```
var tp = new AzureServiceTokenProvider()  
  
var t = new TokenCredential(await tp.GetAccessTokenAsync("..."));  
  
return new StorageCredentials(t);
```

Box 1: AzureServiceTokenProvider()

Box 2: tp.GetAccessTokenAsync(..)

Acquiring an access token is then quite easy. Example code:

```
private async Task<string> GetAccessTokenAsync()  
{  
    var tokenProvider = new AzureServiceTokenProvider();  
    return await tokenProvider.GetAccessTokenAsync("https://storage.azure.com/");  
}
```

Reference:

<https://joonasw.net/view/azure-ad-authentication-with-azure-storage-and-managed-service-identity>

Question #2

**Introductory Info****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 questions in 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 section 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 in 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. 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 SaaS company that provides a solution for managing employee expenses. The solution consists of an ASP.NET Core Web API project that is deployed as an Azure Web App.

**Overall architecture -**

Employees upload receipts for the system to process. When processing is complete, the employee receives a summary report email that details the processing results. Employees then use a web application to manage their receipts and perform any additional tasks needed for reimbursement.

**Receipt processing -**

Employees may upload receipts in two ways:

Uploading using an Azure Files mounted folder

Uploading using the web application

**Data Storage -**

Receipt and employee information is stored in an Azure SQL database.

**Documentation -**

Employees are provided with a getting started document when they first use the solution. The documentation includes details on supported operating systems for

Azure File upload, and instructions on how to configure the mounted folder.

**Solution details -****Users table -**

Column	Description
UserId	unique identifier for an employee
ExpenseAccount	employees expense account number in the format 1234-123-1234
AllowedAmount	limit of allowed expenses before approval is needed
SupervisorId	unique identifier for employee's supervisor
SecurityPin	value used to validate user identity

**Web Application -**

You enable MSI for the Web App and configure the Web App to use the security principal name WebAppIdentity.

**Processing -**

Processing is performed by an Azure Function that uses version 2 of the Azure Function runtime. Once processing is completed, results are stored in Azure Blob

Storage and an Azure SQL database. Then, an email summary is sent to the user with a link to the processing report. The link to the report must

remain valid if the email is forwarded to another user.

#### Logging -

Azure Application Insights is used for telemetry and logging in both the processor and the web application. The processor also has TraceWriter logging enabled.

Application Insights must always contain all log messages.

#### Requirements -

##### Receipt processing -

Concurrent processing of a receipt must be prevented.

##### Disaster recovery -

Regional outage must not impact application availability. All DR operations must not be dependent on application running and must ensure that data in the DR region is up to date.

#### Security -

User's SecurityPin must be stored in such a way that access to the database does not allow the viewing of SecurityPins. The web application is the only system that should have access to SecurityPins.

All certificates and secrets used to secure data must be stored in Azure Key Vault.

You must adhere to the principle of least privilege and provide privileges which are essential to perform the intended function.

All access to Azure Storage and Azure SQL database must use the application's Managed Service Identity (MSI).

Receipt data must always be encrypted at rest.

All data must be protected in transit.

User's expense account number must be visible only to logged in users. All other views of the expense account number should include only the last segment, with the remaining parts obscured.

In the case of a security breach, access to all summary reports must be revoked without impacting other parts of the system.

#### Issues -

##### Upload format issue -

Employees occasionally report an issue with uploading a receipt using the web application. They report that when they upload a receipt using the Azure File Share, the receipt does not appear in their profile. When this occurs, they delete the file in the file share and use the web application, which returns a 500 Internal Server error page.

##### Capacity issue -

During busy periods, employees report long delays between the time they upload the receipt and when it appears in the web application.

##### Log capacity issue -

Developers report that the number of log messages in the trace output for the processor is too high, resulting in lost log messages.

#### Application code -

##### Processing.cs -

```

PC01 public static class Processing
PC02 {
PC03     public static class Function
PC04     {
PC05         [FunctionName("IssueWork")]
PC06         public static async Task Run([TimerTrigger("0 */5 * * * *")] TimerInfo timer, ILogger
log)
PC07         {
PC08             var container = await GetCloudBlobContainer();
PC09             foreach (var fileItem in await ListFiles())
PC10             {
PC11                 var file = new CloudFile(fileItem.StorageUri.PrimaryUri);
PC12                 var ms = new MemoryStream();
PC13                 await file.DownloadToStreamAsync(ms);
PC14                 var blob = container.GetBlockBlobReference(fileItem.Uri.ToString());
PC15                 await blob.UploadFromStreamAsync(ms);
PC16
PC17             }
PC18         }
PC19         private static CloudBlockBlob GetDRBlob(CloudBlockBlob sourceBlob)
PC20         {
PC21             . .
PC22         }
PC23         private static async Task<CloudBlobContainer> GetCloudBlobContainer()
PC24         {
PC25             var cloudBlobClient = new CloudBlobClient(new Uri(" . . ."), await GetCredentials());
PC26
PC27             await cloudBlobClient.GetRootContainerReference().CreateIfNotExistsAsync();
PC28             return cloudBlobClient.GetRootContainerReference();
PC29         }
PC30         private static async Task<StorageCredentials> GetCredentials()
PC31         {
PC32             . .
PC33         }
PC34         private static async Task<List<IListFileItem>> ListFiles()
PC35         {
PC36             . .
PC37         }
PC37         private KeyVaultClient _keyVaultClient = new KeyVaultClient(" . . .");
PC38     }
PC39 }

```

Database.cs -

```

DB01 public class Database
DB02 {
DB03     private string ConnectionString =
DB04
DB05     public async Task<object> LoadUserDetails(string userId)
DB06     {
DB07
DB08         return await policy.ExecuteAsync(async () =>
DB09         {
DB10             using (var connection = new SqlConnection(ConnectionString))
DB11             {
DB12                 await connection.OpenAsync();
DB13                 using (var command = new SqlCommand(" . . .", connection))
DB14                     using (var reader = command.ExecuteReader())
DB15                     {
DB16                         . .
DB17                     }
DB18                 }
DB19             });
DB20         }
DB21     }

```

ReceiptUploader.cs -

```

RU01 public class ReceiptUploader
RU02 {
RU03     public async Task UploadFile(string file, byte[] binary)
RU04     {
RU05         var httpClient = new HttpClient();
RU06         var response = await httpClient.PutAsync("...", new ByteArrayContent(binary));
RU07         while (ShouldRetry(response))
RU08         {
RU09             response = await httpClient.PutAsync("...", new ByteArrayContent(binary));
RU10         }
RU11     }
RU12     private bool ShouldRetry(HttpStatusCode response)
RU13     {
RU14     }
RU15 }
RU16 }
```

ConfigureSSE.ps1 -

```

CS01 $storageAccount = Get-AzureRmStorageAccount -ResourceGroupName "..." -AccountName "..."
CS02 $keyVault = Get-AzureRmKeyVault -VaultName "..."
CS03 $key = Get-AzureKeyVaultKey -VaultName $keyVault.VaultName -Name "..."
CS04 Set-AzureRmKeyVaultAccessPolicy ` 
CS05 -VaultName $keyVault.VaultName ` 
CS06 -ObjectId $storageAccount.Identity.PrincipalId ` 
CS07 
CS08 
CS09 Set-AzureRmStorageAccount ` 
CS10 -ResourceGroupName $storageAccount.ResourceGroupName ` 
CS11 -AccountName $storageAccount.StorageAccountName ` 
CS12 -EnableEncryptionService File ` 
CS13 -KeyvaultEncryption ` 
CS14 -KeyName $key.Name 
CS15 -KeyVersion $key.Version ` 
CS16 -KeyVaultUri $keyVault.VaultUri
```

## Question

DRAG DROP -

You need to ensure disaster recovery requirements are met.

What code should you add at line PC16?

To answer, drag the appropriate code fragments to the correct locations. Each code fragment 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:

Values
true
SingleTransferContext
ShouldTransferCallbackAsync
false
DirectoryTransferContext
ShouldOverwriteCallbackAsync

### Answer Area

```

var copyOptions = new CopyOptions { };
var context = new Value = (source, destination) => Task.FromResult(true);
context. Value = (source, destination) => Task.FromResult(true);
await TransferManager.CopyAsync(blob, GetDRBlob(blob), isServiceCopy: Value
, context: context, options:copyOptions);
```

**Correct Answer:**

Values
true
SingleTransferContext
ShouldOverwriteCallbackAsync

**Answer Area**

```
var copyOptions = new CopyOptions { };
var context = new DirectoryTransferContext = (source, destination) => Task.FromResult(true);
context. ShouldTransferCallbackAsync = (source, destination) => Task.FromResult(true);
await TransferManager.CopyAsync(blob, GetDRBlob(blob), isServiceCopy: false
, context: context, options:copyOptions);
```

Scenario: Disaster recovery. Regional outage must not impact application availability. All DR operations must not be dependent on application running and must ensure that data in the DR region is up to date.

Box 1: DirectoryTransferContext -

We transfer all files in the directory.

**Topic 27 - Testlet 9**

Box 2: ShouldTransferCallbackAsync

The DirectoryTransferContext.ShouldTransferCallbackAsync delegate callback is invoked to tell whether a transfer should be done.

Box 3: False -

If you want to use the retry policy in Copy, and want the copy can be resume if break in the middle, you can use SyncCopy (isServiceCopy = false).

Note that if you choose to use service side copy ('isServiceCopy' set to true), Azure (currently) doesn't provide SLA for that. Setting 'isServiceCopy' to false will download the source blob loca

Reference:

<https://docs.microsoft.com/en-us/azure/storage/common/storage-use-data-movement-library> <https://docs.microsoft.com/en-us/dotnet/api/microsoft.windowsazure.storage.datamovement.directorytransfercontext.shouldtransfercallbackasync?view=azure-dotnet>

Question #1

## Introductory Info

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

LabelMaker app -

Coho Winery produces, bottles, and distributes a variety of wines globally. You are a developer implementing highly scalable and resilient applications to support online order processing by using Azure solutions.

Coho Winery has a LabelMaker application that prints labels for wine bottles. The application sends data to several printers. The application consists of five modules that run independently on virtual machines (VMs). Coho Winery plans to move the application to Azure and continue to support label creation.

External partners send data to the LabelMaker application to include artwork and text for custom label designs.

Requirements. Data -

You identify the following requirements for data management and manipulation:

Order data is stored as nonrelational JSON and must be queried using SQL.

Changes to the Order data must reflect immediately across all partitions. All reads to the Order data must fetch the most recent writes.

Requirements. Security -

You have the following security requirements:

Users of Coho Winery applications must be able to provide access to documents, resources, and applications to external partners.

External partners must use their own credentials and authenticate with their organization's identity management solution.

External partner logins must be audited monthly for application use by a user account administrator to maintain company compliance.

Storage of e-commerce application settings must be maintained in Azure Key Vault.

E-commerce application sign-ins must be secured by using Azure App Service authentication and Azure Active Directory (AAD).

Conditional access policies must be applied at the application level to protect company content.

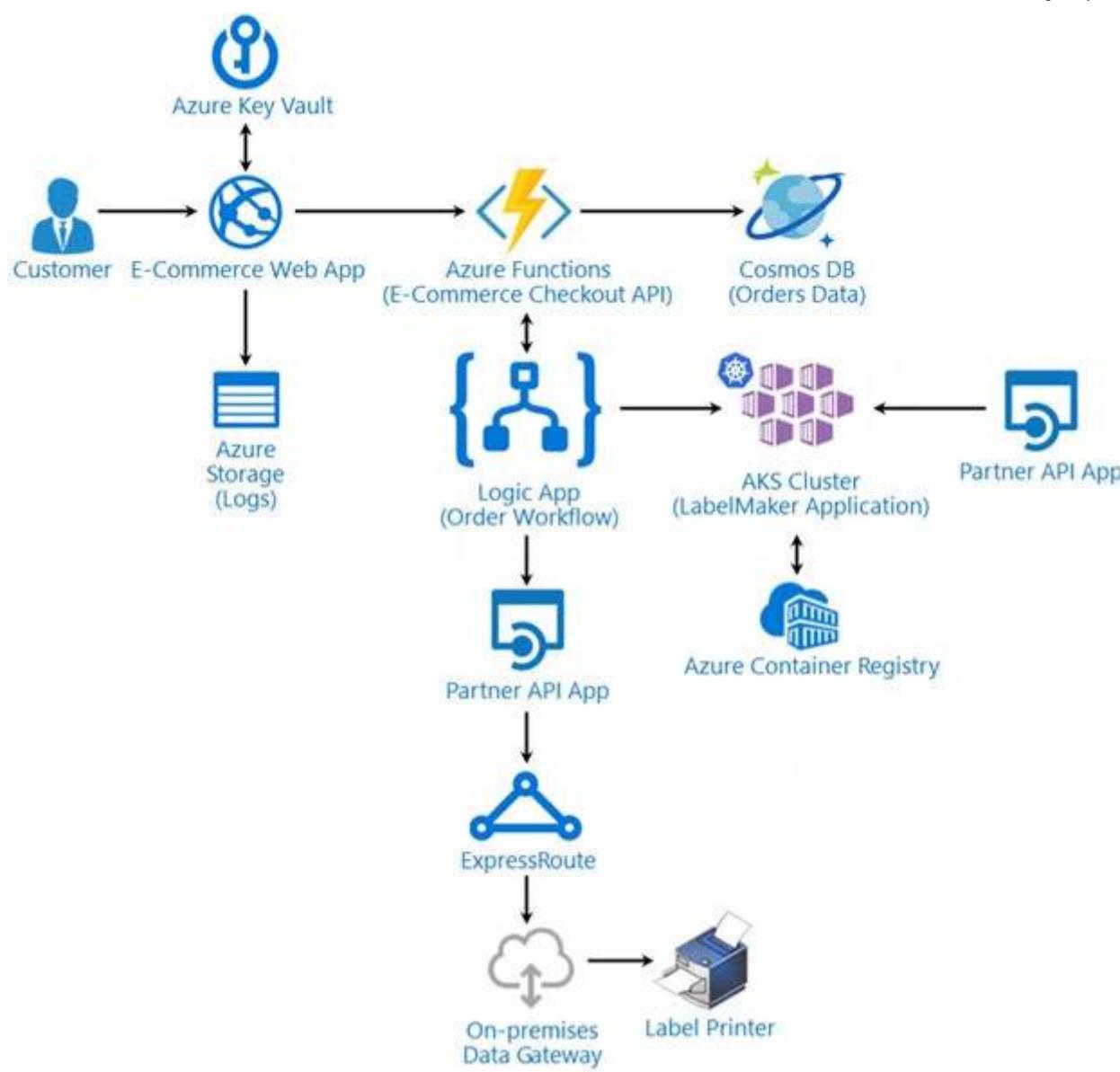
The LabelMaker application must be secured by using an AAD account that has full access to all namespaces of the Azure Kubernetes Service (AKS) cluster.

Requirements. LabelMaker app -

Azure Monitor Container Health must be used to monitor the performance of workloads that are deployed to Kubernetes environments and hosted on Azure Kubernetes Service (AKS).

You must use Azure Container Registry to publish images that support the AKS deployment.

Architecture -



#### Issues -

Calls to the Printer API App fail periodically due to printer communication timeouts.

Printer communication timeouts occur after 10 seconds. The label printer must only receive up to 5 attempts within one minute.

The order workflow fails to run upon initial deployment to Azure.

#### Order.json -

Relevant portions of the app files are shown below. Line numbers are included for reference only.

This JSON file contains a representation of the data for an order that includes a single item.

Order.json -

### Order.json

```
01 {
02   "id" : 1,
03   "customers" : [
04     {
05       "familyName" : "Doe",
06       "givenName" : "John",
07       "customerid" : 5
08     }
09   ],
10   "line_items" : [
11     {
12       "fulfillable_quantity" : 1,
13       "id": 6,
14       "price" : "199.99" ,
15       "product_id" : 7513594,
16       "quantity": 1,
17       "requires_shipping" : true ,
18       "sku": "SFC-342-N" ,
19       "title" : "Surface Go" ,
20       "vendor" : "Microsoft" ,
21       "name" : "Surface Go - 8GB" ,
22       "taxable" : true ,
23       "tax_lines" : [
24         {
25           "title" : "State Tax" ,
26           "price" : "3.98" ,
27           "rate" : 0.06
28         }
29       ],
30       "total_discount" : "5.00" ,
31       "discount_allocations" : [
32         {
33           "amount" : "5.00" ,
34           "discount_application_index" : 2
35         }
36       ]
37 }
```

### Question

HOTSPOT -

You need to configure Azure Cosmos DB.

Which settings should you use? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

### Answer Area

Setting	Value
---------	-------

Consistency Level

Strong
Bounded-staleness
Session
Eventual

API

SQL
MongoDB
Graph
Table

## Answer Area

Setting	Value
---------	-------

Consistency Level

Strong
Bounded-staleness
Session
Eventual

Correct Answer:

API

SQL
MongoDB
Graph
Table

Box 1: Strong -

When the consistency level is set to strong, the staleness window is equivalent to zero, and the clients are guaranteed to read the latest committed value of the write operation.

Scenario: Changes to the Order data must reflect immediately across all partitions. All reads to the Order data must fetch the most recent writes.

Note: You can choose from five well-defined models on the consistency spectrum. From strongest to weakest, the models are: Strong, Bounded staleness,

Session, Consistent prefix, Eventual

Box 2: SQL -

Scenario: You identify the following requirements for data management and manipulation:

Order data is stored as nonrelational JSON and must be queried using Structured Query Language (SQL).

Question #2

## Introductory Info

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 questions in 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 section 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 in 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. When you are ready to answer a question, click the Question button to return to the question.

LabelMaker app -

Coho Winery produces, bottles, and distributes a variety of wines globally. You are a developer implementing highly scalable and resilient applications to support online order processing by using Azure solutions.

Coho Winery has a LabelMaker application that prints labels for wine bottles. The application sends data to several printers. The application consists of five modules that run independently on virtual machines (VMs). Coho Winery plans to move the application to Azure and continue to support label creation.

External partners send data to the LabelMaker application to include artwork and text for custom label designs.

Requirements. Data -

You identify the following requirements for data management and manipulation:

Order data is stored as nonrelational JSON and must be queried using SQL.

Changes to the Order data must reflect immediately across all partitions. All reads to the Order data must fetch the most recent writes.

Requirements. Security -

You have the following security requirements:

Users of Coho Winery applications must be able to provide access to documents, resources, and applications to external partners.

External partners must use their own credentials and authenticate with their organization's identity management solution.

External partner logins must be audited monthly for application use by a user account administrator to maintain company compliance.

Storage of e-commerce application settings must be maintained in Azure Key Vault.

E-commerce application sign-ins must be secured by using Azure App Service authentication and Azure Active Directory (AAD).

Conditional access policies must be applied at the application level to protect company content.

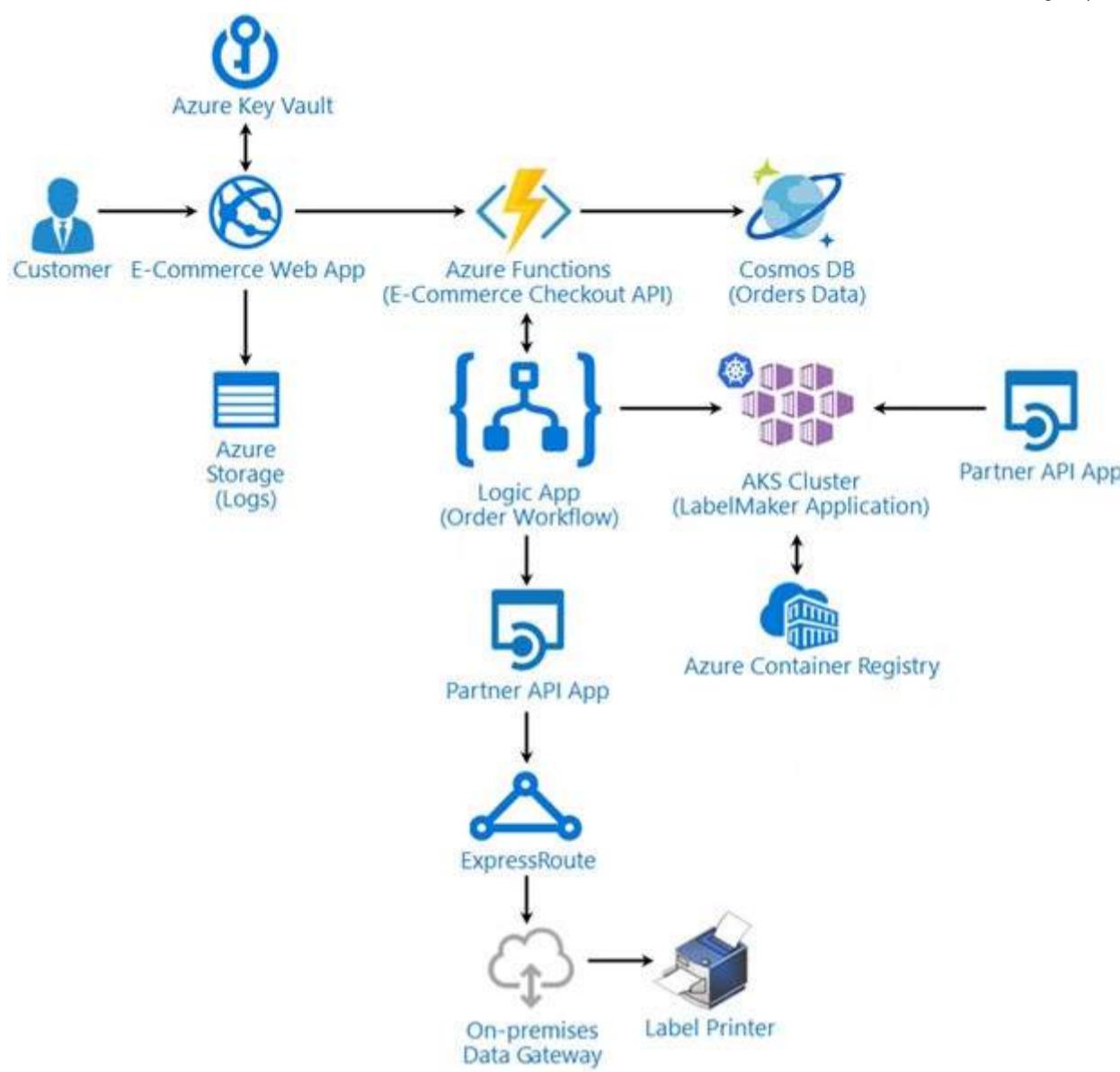
The LabelMaker application must be secured by using an AAD account that has full access to all namespaces of the Azure Kubernetes Service (AKS) cluster.

Requirements. LabelMaker app -

Azure Monitor Container Health must be used to monitor the performance of workloads that are deployed to Kubernetes environments and hosted on Azure Kubernetes Service (AKS).

You must use Azure Container Registry to publish images that support the AKS deployment.

Architecture -



#### Issues -

Calls to the Printer API App fail periodically due to printer communication timeouts.

Printer communication timeouts occur after 10 seconds. The label printer must only receive up to 5 attempts within one minute.

The order workflow fails to run upon initial deployment to Azure.

#### Order.json -

Relevant portions of the app files are shown below. Line numbers are included for reference only.

This JSON file contains a representation of the data for an order that includes a single item.

Order.json -

### Order.json

```
01 {
02   "id" : 1,
03   "customers" : [
04     {
05       "familyName" : "Doe",
06       "givenName" : "John",
07       "customerid" : 5
08     }
09   ],
10   "line_items" : [
11     {
12       "fulfillable_quantity" : 1,
13       "id": 6,
14       "price" : "199.99" ,
15       "product_id" : 7513594,
16       "quantity": 1,
17       "requires_shipping" : true ,
18       "sku": "SFC-342-N" ,
19       "title" : "Surface Go" ,
20       "vendor" : "Microsoft" ,
21       "name" : "Surface Go - 8GB" ,
22       "taxable" : true ,
23       "tax_lines" : [
24         {
25           "title" : "State Tax" ,
26           "price" : "3.98" ,
27           "rate" : 0.06
28         }
29       ],
30       "total_discount" : "5.00" ,
31       "discount_allocations" : [
32         {
33           "amount" : "5.00" ,
34           "discount_application_index" : 2
35         }
36     ]
37 }
```

### Question

HOTSPOT -

You need to retrieve all order line items from Order.json and sort the data alphabetically by the city.

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

NOTE: Each correct selection is worth one point.

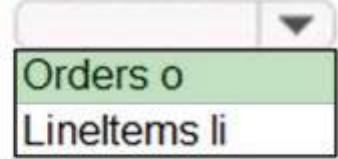
Hot Area:

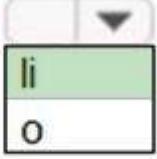
### Answer Area

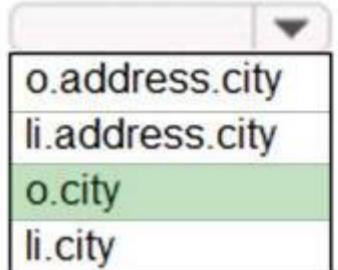
```
SELECT li.id AS lineitemid, li.price
FROM Orders o
JOIN Lineltems li
ON li.o
ORDER BY o.address.city ASC
```

**Answer Area**

SELECT li.id AS lineitemid, li.price

FROM  Orders o  
LineItems li

JOIN  IN  li  
o

ORDER BY  ASC  
o.address.city  
li.address.city  
o.city  
li.city

Correct Answer:

Box 1: orders o -

Scenario: Order data is stored as nonrelational JSON and must be queried using SQL.

Box 2:li -

Box 3: o.line\_items -