

 Custom View Settings

Topic 1 - Question Set 1

Question #1

Topic 1

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: C

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>

Community vote distribution

C (100%)

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.

Select and Place:

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

Correct Answer:

Options

Answer

An Azure Key Vault

An Azure Storage account

An access policy

Azure Active Directory (AD)
Identity Protection

An Azure policy

A backup policy

Question #3

Topic 1

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>

Community vote distribution

A (69%)

B (31%)

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

Community vote distribution

B (100%)

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>

Community vote distribution

B (76%)

D (24%)

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>

Community vote distribution

D (100%)

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>

Community vote distribution

B (100%)

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 options from the list to the answer area.

Select and Place:

Options

Azure Cosmos DB connector

SendGrid action

Consumption plan

Azure Event Hubs binding

SendGrid binding

Answer

Options

Answer

Azure Cosmos DB connector

SendGrid action

Correct Answer:

Azure Event Hubs binding

Consumption plan

SendGrid binding

Question #9

Topic 1

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/>

Community vote distribution

B (62%)

A (38%)

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/>

Community vote distribution

C (89%) 11%

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>

Community vote distribution

A (100%)

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: BD

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>

Community vote distribution

BD (100%)

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 in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

Answer Area

Require MFA for Azure port... X

i Info Delete

* Name
MFA required for Azure portal access

Assignments

- Users and groups i >
0 users selected
- Cloud apps i >
0 apps selected
- Conditions i >
0 conditions selected

Access controls

- Grant i >
0 controls selected
- Session i >
0 controls selected

Answer Area

The screenshot shows the configuration page for a conditional access policy named 'MFA required for Azure portal access'. The 'Assignments' section is expanded, showing options for 'Users and groups', 'Cloud apps', and 'Conditions', each with a count of '0 users selected', '0 apps selected', and '0 conditions selected' respectively. The 'Access controls' section is also expanded, showing 'Grant' and 'Session' controls, both with a count of '0 controls selected'. The 'Info' and 'Delete' buttons are visible at the top.

Correct Answer:

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>

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

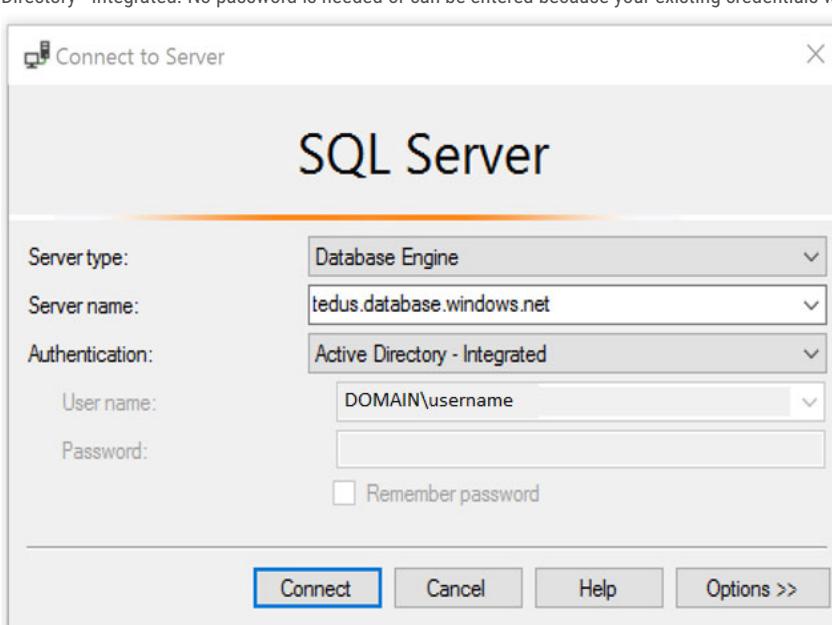
The following procedures show you how to connect to a SQL database with an Azure AD identity using SQL Server Management Studio or SQL Server Database Tools.

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.)

Community vote distribution

C (100%)

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: B

When soft-delete is enabled, resources marked as deleted 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>

Community vote distribution

B (100%)

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 in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

Answer Area

<code>az monitor metrics alert create -n myAlert -g myResourceGroup</code>	--scopes targetResourceId --condition " > 85"
5m	CPU Usage
--window size	Percentage CPU
--evaluation-frequency	avg Percentage CPU
--auto-mitigate	

Answer Area

<code>az monitor metrics alert create -n myAlert -g myResourceGroup</code>	--scopes targetResourceId --condition " > 85"
Correct Answer: 5m	CPU Usage
--window size	Percentage CPU
--evaluation-frequency	avg Percentage CPU
--auto-mitigate	

Reference:

<https://docs.microsoft.com/en-us/azure/monitor/metrics/alert>

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

Community vote distribution

B (100%)

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/>

Community vote distribution

A (100%)

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

Community vote distribution

B (100%)

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/>

Community vote distribution

C (100%)

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>

Community vote distribution

C (100%)

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>

Community vote distribution

B (100%)

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 -

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.

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>

Community vote distribution

B (83%)

D (17%)

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>

Community vote distribution

B (70%)

A (30%)

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>

Community vote distribution

A (73%)

B (27%)

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>

Community vote distribution

A (56%)

B (44%)

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>

Community vote distribution

B (69%)

A (31%)

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>

Community vote distribution

D (100%)

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: A

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>

Community vote distribution

D (66%)

B (34%)

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 from left to right and arrange them in the correct order.

Select and Place:

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.

Correct Answer:

Answer Area

Create a DataSource instance and set its Container property to the DataContainer.

Create a SearchIndexClient object to connect to the search index.

Set the DataSources property of the SearchServiceClient.

Create an IndexBatch that contains the documents which must be added.

Call the Documents.Index method of the SearchIndexClient and pass the IndexBatch.

Call the Documents.Suggest method of the SearchIndexClient and pass the DataSource.

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 #31

Topic 1

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>

Community vote distribution

C (54%)

D (46%)

HOTSPOT -

You are developing a C++ application that compiles to a native application named process.exe. The application accepts images as input and returns images in one of the following image formats: GIF, PNG, or JPEG.

You must deploy the application as an Azure Function.

You need to configure the function and host json files.

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

NOTE: Each correct selection is worth one point.

Hot Area:

`function.json`

```
{  
    "type": "http"  
    "platform": "gcm"  
    "datatype": "stream"  
    "path": "process.exe"
```

```
    "direction": "out",  
    "name" : "result"
```

}

`host.json`

```
"customHandler": { "description": {  
    "languageWorker": { "path": {  
        "extensions": { "worker": {  
            "extensionBundle": {
```

```
                "defaultExecutablePath": "process.exe"
```

,

```
    "enableForwardingHttpRequest": true  
    "enableForwardingHttpRequest": false
```

}

```

function.json
{
  "type": "http"
  "platform": "gcm"
  "datatype": "stream"
  "path": "process.exe"

  "direction": "out",
  "name" : "result"
}

```

host.json

Correct Answer:

```

  "customHandler": { "description": {
    "languageWorker": { "path": {
      "extensions": { "worker": {
        "extensionBundle": {
          "defaultExecutablePath": "process.exe"
        }
      }
    }
  }
  "enableForwardingHttpRequest": true
  "enableForwardingHttpRequest": false
}
}

```

Box 1: "type": "http"

Box 2: "customHandler": { "description":{

A custom handler is defined by configuring the host.json file with details on how to run the web server via the customHandler section.

The customHandler section points to a target as defined by the defaultExecutablePath.

Example:

```
"customHandler": {
  "description": {
```

"defaultExecutablePath": "handler.exe"

Box 3: "enableForwardingHttpRequest": false

Incorrect:

For HTTP-triggered functions with no additional bindings or outputs, you may want your handler to work directly with the HTTP request and response instead of the custom handler request and response payloads. This behavior can be configured in host.json using the enableForwardingHttpRequest setting.

At the root of the app, the host.json file is configured to run handler.exe and enableForwardingHttpRequest is set to true.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-functions/functions-custom-handlers>

HOTSPOT

You are developing an Azure Static Web app that contains training materials for a tool company. Each tool's training material is contained in a static web page that is linked from the tool's publicly available description page.

A user must be authenticated using Azure AD prior to viewing training.

You need to ensure that the user can view training material pages after authentication.

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

NOTE: Each correct selection is worth one point.

Answer Area

The screenshot shows a JSON configuration editor with the following structure:

```
{  
  "routes": [  
    "headers": [  
      "responseOverrides": {  
        "statusCode": 400,  
        "statusMessage": "Bad Request",  
        "headers": [{"name": "Content-Type", "value": "application/json"}],  
        "body": "The request is invalid."},  
        {"statusCode": 401,  
        "statusMessage": "Unauthorized",  
        "headers": [{"name": "WWW-Authenticate", "value": "Bearer"}, {"name": "Content-Type", "value": "application/json"}],  
        "body": "Authentication is required."},  
        {"statusCode": 403,  
        "statusMessage": "Forbidden",  
        "headers": [{"name": "Content-Type", "value": "application/json"}],  
        "body": "Access is denied."},  
        {"statusCode": 404,  
        "statusMessage": "Not Found",  
        "headers": [{"name": "Content-Type", "value": "application/json"}],  
        "body": "The requested resource was not found."}  
      ],  
      "navigationFallback": {  
        "to": "/index.html",  
        "statusCode": 302  
      }  
    }  
  ]  
}
```

The configuration includes routes for handling 400, 401, 403, and 404 errors. It also defines a navigation fallback rule to redirect to /index.html with a 302 status code. The 'routes' array is currently empty.

Answer Area

Correct Answer:

```
{
  "routes": [
    {
      "method": "POST",
      "path": "/auth/login",
      "status": 200,
      "body": {
        "message": "User logged in successfully"
      }
    },
    {
      "method": "GET",
      "path": "/auth/logout",
      "status": 200,
      "body": {
        "message": "User logged out successfully"
      }
    }
  ],
  "headers": [
    {
      "name": "Content-Type",
      "value": "application/json"
    }
  ],
  "responseOverrides": [
    {
      "status": 401,
      "body": {
        "message": "Unauthorized"
      }
    }
  ],
  "navigationFallback": {
    "enabled": true,
    "uri": "/index.html"
  }
}

"redirect": "/auth/login/?post_=add",
"statusCode": 302,
"referrer": "orig"
}
```

HOTSPOT

You are authoring a set of nested Azure Resource Manager templates to deploy Azure resources. You author an Azure Resource Manager template named mainTemplate.json that contains the following linked templates: linkedTemplate1.json, linkedTemplate2.json.

You add parameters to a parameters template file named mainTemplate.parameters.json. You save all templates on a local device in the C:\templates\ folder.

You have the following requirements:

- Store the templates in Azure for later deployment.
- Enable versioning of the templates.
- Manage access to the templates by using Azure RBAC.
- Ensure that users have read-only access to the templates.
- Allow users to deploy the templates.

You need to store the templates in Azure.

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

NOTE: Each correct selection is worth one point.

Answer Area

```
az ts create
storage account create
storage account update
blueprint artifact template create
--name templateStore \
--version "1.0" \
--resource-group templatesRG \
--location "eastus" \
--template-file "C:\templates\mainTemplate.json"
"mainTemplate.json
linkedTemplate1.json
linkedTemplate2.json
mainTemplate.parameters.json"
--tags Dept=HumanResources Environment=Production
```

Answer Area

Correct Answer:

```
az ts create
storage account create
storage account update
blueprint artifact template create

--name templateStore \
--version "1.0" \
--resource-group templatesRG \
--location "eastus" \

--template-file "C:\templates\mainTemplate.json"
linkedTemplate1.json
linkedTemplate2.json
mainTemplate.parameters.json

--tags Dept=HumanResources Environment=Production
```

HOTSPOT

You are developing a service where customers can report news events from a browser using Azure Web PubSub. The service is implemented as an Azure Function App that uses the JSON WebSocket subprotocol to receive news events.

You need to implement the bindings for the Azure Function App.

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

NOTE: Each correct selection is worth one point.

Answer Area

```
{  
  "bindings": [  
    {  
      "type": "user",  
      "direction": "in",  
      "name": "data",  
      "eventName": "message",  
      "eventType": "user"  
    }  
  ]  
}
```

Answer Area

```
{  
  "bindings": [  
    {  
      "type": "user",  
      "direction": "in",  
      "name": "data",  
      "eventName": "message",  
      "eventType": "user"  
    }  
  ]  
}
```

Correct Answer:

```
"direction": "in",  
"name": "data",  
"eventName": "message",  
"eventType": "user",  
"type": "webPubSubTrigger",  
"user": "user",  
"system": "system",  
"message": "message",  
"connected": "connected",  
"webPubSubTrigger": "webPubSubTrigger",  
"webPubSubConnection": "webPubSubConnection"}  
]  
}
```

HOTSPOT

You are building a software-as-a-service (SaaS) application that analyzes DNA data that will run on Azure virtual machines (VMs) in an availability zone. The data is stored on managed disks attached to the VM. The performance of the analysis is determined by the speed of the disk attached to the VM.

You have the following requirements:

- The application must be able to quickly revert to the previous day's data if a systemic error is detected.
- The application must minimize downtime in the case of an Azure datacenter outage.

You need to provision the managed disk for the VM to maximize performance while meeting the requirements.

Which type of Azure Managed Disk should you use? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Answer Area**Requirement Solution**

Disk type	<input type="button" value="▼"/>
Premium SSD Standard SSD Standard HDD	

Redundancy	<input type="button" value="▼"/>
Geo-redundant storage (GRS) Zone-redundant storage (ZRS) Locally-redundant storage (LRS)	

Answer Area**Requirement Solution**

Correct Answer:

Disk type	<input type="button" value="▼"/>
Premium SSD Standard SSD Standard HDD	

Redundancy	<input type="button" value="▼"/>
Geo-redundant storage (GRS) Zone-redundant storage (ZRS) Locally-redundant storage (LRS)	

HOTSPOT

You are developing an application that includes two Docker containers.

The application must meet the following requirements:

- The containers must not run as root.
- The containers must be deployed to Azure Container Instances by using a YAML file.
- The containers must share a lifecycle, resources, local network, and storage volume.
- The storage volume must persist through container crashes.
- The storage volume must be deployed on stop or restart of the containers.

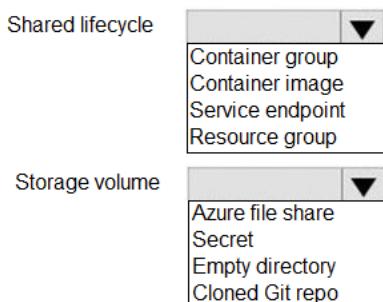
You need to configure Azure Container Instances for the application.

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

NOTE: Each correct selection is worth one point.

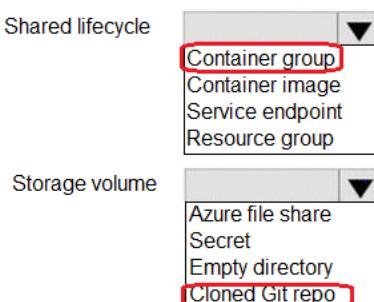
Answer Area

Configuration setting Configuration value

**Answer Area**

Configuration setting Configuration value

Correct Answer:



Question #1

Topic 2

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/>

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		
ScaledObject	Polling interval	
TriggerAuthentication	Azure Storage connection string	

Answer Area

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

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/
```

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	https://github.com/Contos/webapp
\$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
```

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

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

```
source config --name $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
az webapp --name $webappName --resource-group myResourceGroup --sku FREE
az appservice plan create
az webapp deployment
az group delete

az webapp create --name $webappName --resource-group myResourceGroup
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
az webapp deployment
az group delete

--resource-group myResourceGroup
--repo-url $gitrepo --branch master --manual-integration
git clone $gitrepo
--plan $webappName
```

Box 1: az appservice plan create

The azure group creates command successfully returns JSON result. Now we can use resource group to create a azure app service plan

Box 2: az webapp create -

Create a new web app..

Box 3: --plan \$webappName -

..with the serviceplan we created in step 1.

Box 4: az webapp deployment -

Continuous Delivery with GitHub. Example:

```
az webapp deployment source config --name firstsamplewebsite1 --resource-group websites --repo-url $gitrepo --branch master --git-token
$token
```

Box 5: --repo-url \$gitrepo --branch master --manual-integration

Reference:

<https://medium.com/@satish1v/devops-your-way-to-azure-web-apps-with-azure-cli-206ed4b3e9b1>

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>

Community vote distribution

B (67%)

A (33%)

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>

Community vote distribution

B (83%)

A (17%)

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>

Community vote distribution

A (70%)

B (30%)

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>

Community vote distribution

A (88%)

12%

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>

Community vote distribution

B (88%)

13%

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 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 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>

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

`az group create`

`az group update`

`az webapp update`

`az webapp create`

`az appservice plan create`

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

`az group create`

`az group create`

`az group update`

`az appservice plan create`

Correct Answer:

`az webapp update`

`az webapp create`

`az webapp create`

`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-ml/azure/app-service/containers/quickstart-docker-go?view=sql-server-ver15>

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

```
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

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>

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 Premium plan type.	
Create an access policy in Azure Key Vault for the application identity.	
Correct Answer: 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 2

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.

Community vote distribution

D (100%)

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.

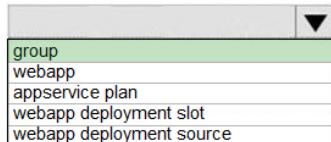
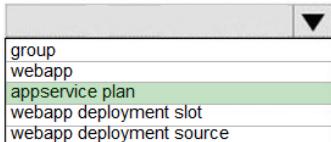
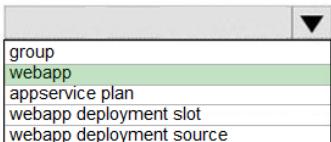
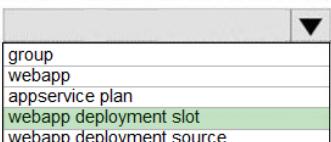
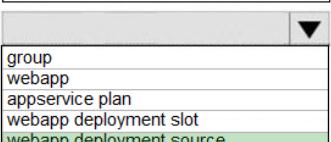
Hot Area:

Answer Area

gitrepo=https://github.com/Contoso/webapp							
webappname=businesswebapp							
resourcegroupname=BusinessAppResourceGroup							
az	<input type="button" value="▼"/>	create --location centralus --name \$resourcegroupname					
<table border="1"><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>		group	webapp	appservice plan	webapp deployment slot	webapp deployment source	
group							
webapp							
appservice plan							
webapp deployment slot							
webapp deployment source							
az	<input type="button" value="▼"/>	create --name \$webappname --resource-group \$resourcegroupname --sku S3					
<table border="1"><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>		group	webapp	appservice plan	webapp deployment slot	webapp deployment source	
group							
webapp							
appservice plan							
webapp deployment slot							
webapp deployment source							
az	<input type="button" value="▼"/>	create --name \$webappname --resource-group \$resourcegroupname --plan \$webappname					
<table border="1"><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>		group	webapp	appservice plan	webapp deployment slot	webapp deployment source	
group							
webapp							
appservice plan							
webapp deployment slot							
webapp deployment source							
az	<input type="button" value="▼"/>	create --name \$webappname --resource-group \$resourcegroupname --slot staging					
<table border="1"><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>		group	webapp	appservice plan	webapp deployment slot	webapp deployment source	
group							
webapp							
appservice plan							
webapp deployment slot							
webapp deployment source							
az	<input type="button" value="▼"/>	config --name \$webappname --resource-group \$resourcegroupname \ --slot staging --repo-url \$gitrepo --branch master --manual-integration					
<table border="1"><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>		group	webapp	appservice plan	webapp deployment slot	webapp deployment source	
group							
webapp							
appservice plan							
webapp deployment slot							
webapp deployment source							

Correct Answer:

Answer Area

gitrepo=https://github.com/Contoso/webapp	
webappname=businesswebapp	
resourcegroupname=BusinessAppResourceGroup	
az	create --location centralus --name \$resourcegroupname
	
az	create --name \$webappname --resource-group \$resourcegroupname --sku S3
	
az	create --name \$webappname --resource-group \$resourcegroupname --plan \$webappname
	
az	create --name \$webappname --resource-group \$resourcegroupname --slot staging
	
az	config --name \$webappname --resource-group \$resourcegroupname \ --slot staging --repo-url \$gitrepo --branch master --manual-integration
	

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>

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 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 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>

Community vote distribution

A (81%)

B (19%)

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

Community vote distribution

A (55%)

B (45%)

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>

Community vote distribution

B (67%)

A (33%)

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>

Community vote distribution

B (53%)

A (47%)

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>

Community vote distribution

B (100%)

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 -

Bx 4: COPY ./ .

Box 5: RUN powershell ./setupScript.ps1

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>

Community vote distribution

D (100%)

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":   
            "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))]",  
        "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": [
```

```
"[resourceId('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
```

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
------------	-----------

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
------------	-----------

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/functions/functions-bindings-storage-queue>

Question #25

Topic 2

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>

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.	<input type="checkbox"/> Azure PowerShell <input type="checkbox"/> Visual Studio command prompt <input type="checkbox"/> Azure Migrate <input type="checkbox"/> Azure Backup
Store images.	<input type="checkbox"/> Azure Blob Storage <input type="checkbox"/> Azure Data Lake Storage <input type="checkbox"/> Azure File Storage <input type="checkbox"/> Azure Table Storage

Correct Answer:

Answer Area

Action	Tool or service
Generalize the VM.	<input checked="" type="checkbox"/> Azure PowerShell <input type="checkbox"/> Visual Studio command prompt <input type="checkbox"/> Azure Migrate <input type="checkbox"/> Azure Backup
Store images.	<input checked="" type="checkbox"/> Azure Blob Storage <input type="checkbox"/> Azure Data Lake Storage <input type="checkbox"/> Azure File Storage <input type="checkbox"/> 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>

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>

Community vote distribution

BD (84%)

Other

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**Answer Area**

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.

Correct Answer:

Actions**Answer Area**

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

Create a new template deployment.



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

Export a Resource Manager template.

Export a Resource Manager template.

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

Create a new template deployment.

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

Modify the template by changing the storage account name and 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 2

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

Run Command

Serial console

Hybrid Runbook Worker

Custom Script Extension

Answer Area

Requirement

Firewall configuration

Supporting services script

Feature

Correct Answer:

Features

Serial console

Custom Script Extension

Answer Area

Requirement

Firewall configuration

Supporting services script

Feature

Run Command

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>

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"
```

<input type="checkbox"/> New-AzWebAppSlot	<input type="checkbox"/> New-AzWebApp	<input type="checkbox"/> New-AzAppServicePlan	<input type="checkbox"/> New-AzResourceGroup	-Name myResourceGroup -Location \$location
<input type="checkbox"/> New-AzWebAppSlot	<input type="checkbox"/> New-AzWebApp	<input type="checkbox"/> New-AzAppServicePlan	<input type="checkbox"/> New-AzResourceGroup	-Name \$webappname -Location \$location -ResourceGroupName myResourceGroup -Tier Standard
<input type="checkbox"/> New-AzWebAppSlot	<input type="checkbox"/> New-AzWebApp	<input type="checkbox"/> New-AzAppServicePlan	<input type="checkbox"/> New-AzResourceGroup	-Name \$webappname -Location \$location -AppServicePlan \$webappname -ResourceGroupName myResourceGroup
<input type="checkbox"/> New-AzWebAppSlot	<input type="checkbox"/> New-AzWebApp	<input type="checkbox"/> New-AzAppServicePlan	<input type="checkbox"/> New-AzResourceGroup	-Name \$webappname -ResourceGroupName myResourceGroup -Slot review
<input type="checkbox"/> New-AzWebAppSlot	<input type="checkbox"/> New-AzWebApp	<input type="checkbox"/> New-AzAppServicePlan	<input type="checkbox"/> 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"
```

<input type="checkbox"/> New-AzWebAppSlot	<input type="checkbox"/> New-AzWebApp	<input type="checkbox"/> New-AzAppServicePlan	<input checked="" type="checkbox"/> New-AzResourceGroup	-Name myResourceGroup -Location \$location
<input type="checkbox"/> New-AzWebAppSlot	<input type="checkbox"/> New-AzWebApp	<input checked="" type="checkbox"/> New-AzAppServicePlan	<input type="checkbox"/> New-AzResourceGroup	-Name \$webappname -Location \$location -ResourceGroupName myResourceGroup -Tier Standard
<input type="checkbox"/> New-AzWebAppSlot	<input checked="" type="checkbox"/> New-AzWebApp	<input type="checkbox"/> New-AzAppServicePlan	<input type="checkbox"/> New-AzResourceGroup	-Name \$webappname -Location \$location -AppServicePlan \$webappname -ResourceGroupName myResourceGroup
<input checked="" type="checkbox"/> New-AzWebAppSlot	<input type="checkbox"/> New-AzWebApp	<input type="checkbox"/> New-AzAppServicePlan	<input type="checkbox"/> New-AzResourceGroup	-Name \$webappname -ResourceGroupName myResourceGroup -Slot review
<input type="checkbox"/> New-AzWebAppSlot	<input type="checkbox"/> New-AzWebApp	<input type="checkbox"/> New-AzAppServicePlan	<input type="checkbox"/> 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

Topic 2

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>

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>

Community vote distribution

B (100%)

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>

Community vote distribution

B (80%)

A (20%)

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"/>

Correct Answer:

The AzScheduledQueryRuleSource is Heartbeat, not CPU.

Box 1: No -

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 2: Yes -

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

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 2

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:

Features

- Custom handler
- Extension bundle
- Trigger
- Runtime
- Policy
- Hosting plan

Answer Area

Requirement

Feature

Enable developers to write the functions by using the Rust language.

Feature

Declaratively connect to an Azure Blob Storage account.

Feature

Correct Answer:

Features

- Custom handler
- Extension bundle
- Runtime
- Policy
- Hosting plan

Answer Area

Requirement

Feature

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>

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	true
WEBSITE_LOCALCACHE_ENABLED	
DOTNET_HOSTING_OPTIMIZATION_CACHE	
WEBSITES_ENABLE_APP_SERVICE_STORAGE	
DIAGDATA	/home /local D:\home D:\local

Correct Answer:

Answer Area

App setting	Value
LOCALAPPDATA	true
WEBSITE_LOCALCACHE_ENABLED	
DOTNET_HOSTING_OPTIMIZATION_CACHE	
WEBSITES_ENABLE_APP_SERVICE_STORAGE	
DIAGDATA	/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>

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

* Protocol

HTTP HTTPS



Authentication certificates are not required for trusted Azure certificates for end to end ssl to work

* Port ⓘ

443



* Request timeout (seconds)

20

Override backend path ⓘ

Use for App service

Use custom probe ⓘ

OK

Reference:

<https://docs.microsoft.com/en-us/azure/application-gateway/application-gateway-web-app-overview>

Community vote distribution

AD (100%)

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>

Community vote distribution

B (100%)

You are developing a web application that runs as an Azure Web App. The web application stores data in Azure SQL Database and stores files in an Azure

Storage account. The web application makes HTTP requests to external services as part of normal operations.

The web application is instrumented with Application Insights. The external services are OpenTelemetry compliant.

You need to ensure that the customer ID of the signed in user is associated with all operations throughout the overall system.

What should you do?

- A. Add the customer ID for the signed in user to the CorrelationContext in the web application
- B. On the current SpanContext, set the Traceld to the customer ID for the signed in user
- C. Set the header Ocp-Apim-Trace to the customer ID for the signed in user
- D. Create a new SpanContext with the TraceFlags value set to the customer ID for the signed in user

Correct Answer: A

Reference:

<https://docs.microsoft.com/en-us/azure/azure-monitor/app/correlation>

Community vote distribution

A (79%)

B (21%)

HOTSPOT -

You are developing an Azure Function App. You develop code by using a language that is not supported by the Azure Function App host. The code language supports HTTP primitives.

You must deploy the code to a production Azure Function App environment.

You need to configure the app for deployment.

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**Configuration parameter****Configuration value****Publish**

Code
Docker Container

Runtime stack

Node.js
Python
PowerShell Core
Custom Handler

Version

14 LTS
7.0
custom

Answer Area

Configuration parameter	Configuration value
Publish	<div style="border: 1px solid black; padding: 5px; display: inline-block;">▼</div> <div style="border: 1px solid black; padding: 5px; display: inline-block; background-color: #e0f2e0;">Code</div> <div style="border: 1px solid black; padding: 5px; display: inline-block; background-color: #e0f2e0;">Docker Container</div>
Runtime stack	<div style="border: 1px solid black; padding: 5px; display: inline-block;">▼</div> <div style="border: 1px solid black; padding: 5px; display: inline-block; background-color: #e0f2e0;">Node.js</div> <div style="border: 1px solid black; padding: 5px; display: inline-block; background-color: #e0f2e0;">Python</div> <div style="border: 1px solid black; padding: 5px; display: inline-block; background-color: #e0f2e0;">PowerShell Core</div> <div style="border: 1px solid black; padding: 5px; display: inline-block; background-color: #e0f2e0;">Custom Handler</div>
Version	<div style="border: 1px solid black; padding: 5px; display: inline-block;">▼</div> <div style="border: 1px solid black; padding: 5px; display: inline-block; background-color: #e0f2e0;">14 LTS</div> <div style="border: 1px solid black; padding: 5px; display: inline-block; background-color: #e0f2e0;">7.0</div> <div style="border: 1px solid black; padding: 5px; display: inline-block; background-color: #e0f2e0;">custom</div>

Box 1: Docker container -

A custom handler can be deployed to every Azure Functions hosting option. If your handler requires operating system or platform dependencies (such as a language runtime), you may need to use a custom container. You can create and deploy your code to Azure Functions as a custom Docker container.

Box 2: PowerShell core -

When creating a function app in Azure for custom handlers, we recommend you select .NET Core as the stack. A "Custom" stack for custom handlers will be added in the future.

PowerShell Core (PSC) is based on the new .NET Core runtime.

Box 3: 7.0 -

On Windows: The Azure Az PowerShell module is also supported for use with PowerShell 5.1 on Windows.

On Linux: PowerShell 7.0.6 LTS, PowerShell 7.1.3, or higher is the recommended version of PowerShell for use with the Azure Az PowerShell module on all platforms.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-functions/functions-create-function-linux-custom-image> <https://docs.microsoft.com/en-us/powershell/azure/install-az-ps?view=azps-7.1.0>

DRAG DROP -

You provision virtual machines (VMs) as development environments.

One VM does not start. The VM is stuck in a Windows update process. You attach the OS disk for the affected VM to a recovery VM.

You need to correct the issue.

In which order should you perform the actions? 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**

Run the following command at an elevated command prompt:

```
dism /image:\ /get=packages > c:\temp\Patch.txt
```



Run the following command at an elevated command prompt:

```
dism /Image:<Attached OS disks>:\ /Remove  
Package /PackageName:<package name to delete>
```

Detach the OS disk and recreate the VM

Open C:\temp\Patch.txt file and locate the update that is in a pending state

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

Run the following command at an elevated command prompt:

```
dism /image:\ /get=packages > c:\temp\Patch.txt
```



Open C:\temp\Patch.txt file and locate the update that is in a pending state

Run the following command at an elevated command prompt:

```
dism /Image:<Attached OS disks>:\ /Remove  
Package /PackageName:<package name to delete>
```

Detach the OS disk and recreate the VM

Remove the update that causes the problem

1. Take a snapshot of the OS disk of the affected VM as a backup.
2. Attach the OS disk to a recovery VM.
3. Once the OS disk is attached on the recovery VM, run diskmgmt.msc to open Disk Management, and ensure the attached disk is ONLINE.
4. (Step 1) Open an elevated command prompt instance (Run as administrator). Run the following command to get the list of the update packages that are on the attached OS disk: `dism /image:<Attached OS disk>:\ /get-packages > c:\temp\Patch_level`
5. (Step 2) Open the C:\temp\Patch_level.txt file, and then read it from the bottom up. Locate the update that's in Install Pending or Uninstall Pending state.
6. Remove the update that caused the problem:
`dism /Image:<Attached OS disk>:\ /Remove-Package /PackageName:<PACK`
7. (Step 4) Detach the OS disk and recreate the VM. Then check whether the issue is resolved.

Reference:

<https://docs.microsoft.com/en-us/troubleshoot/azure/virtual-machines/troubleshoot-stuck-updating-boot-error>

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: Update the functionTimeout property of the host.json project file to 10 minutes.

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>

Community vote distribution

B (100%)

HOTSPOT -

You are developing an Azure Durable Function based application that processes a list of input values. The application is monitored using a console application that retrieves JSON data from an Azure Function diagnostic endpoint.

During processing a single instance of invalid input does not cause the function to fail. Invalid input must be available to the monitoring application.

You need to implement the Azure Durable Function and the monitoring console application.

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

NOTE: Each correct selection is worth one point.

Hot Area:

Answer Area

```
[FunctionName("App")]
public static async Task<List<string>> RunOrchestrator(
    [OrchestrationTrigger] IDurableOrchestrationContext context) {
    EntityId[] input = . . .
    int errIndex = . . .

    ;  

    context.SetOutput(input[errIndex])
    context.SetCustomStatus(input[errIndex])
    context.SignalEntity(input[errIndex], "error")
    await context.CallEntityAsync(input[errIndex], "error")
}

using (var client = new HttpClient())
{
    while (true)
    {
        var response = await client.GetAsync(" . . .");
        response.EnsureSuccessStatusCode();
        var json = await response.Content.ReadAsStringAsync();
        dynamic result = JsonConvert.DeserializeObject(json);
        if (result.runtimeStatus == "")  

            Failed  

            Awaited  

            Listening  

            Completed
        {
            return result.
        }
    }
}
```

input
output
runtimeStatus
customStatus

Answer Area

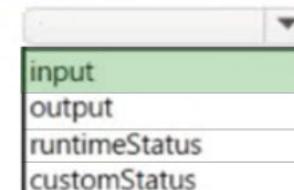
```
[FunctionName("App")]
public static async Task<List<string>> RunOrchestrator(
    [OrchestrationTrigger] IDurableOrchestrationContext context) {
    EntityId[] input = . . .
    int errIndex = . . .

    context.SetOutput(input[errIndex]);
    context.SetCustomStatus(input[errIndex]);
    context.SignalEntity(input[errIndex], "error");
    await context.CallEntityAsync(input[errIndex], "error");
}

using (var client = new HttpClient())
{
    while (true)
    {
        var response = await client.GetAsync("...");  

        response.EnsureSuccessStatusCode();
        var json = await response.Content.ReadAsStringAsync();
        dynamic result = JsonConvert.DeserializeObject(json);
        if (result.runtimeStatus == "Failed")
        {
            return result;
        }
    }
}
```

Correct Answer:



Box 1: await context.CallEntityAsync(input[errindex],"error")

Orchestration signals and calls an entity

Orchestrator functions can access entities by using APIs on the orchestration trigger binding.

Example:

```
[FunctionName("CounterOrchestration")]
public static async Task Run(
    [OrchestrationTrigger] IDurableOrchestrationContext context)
{
    var entityId = new EntityId(nameof(Counter), "myCounter");
    // Two-way call to the entity which returns a value - awaits the response int currentValue = await context.CallEntityAsync<int>(entityId, "Get");
```

Box 2: Failed -

During processing a single instance of invalid input does not cause the function to fail.

Note: RuntimeStatus: One of the following values:

Failed: The instance failed with an error.

Completed: The instance has completed normally.

Terminated: The instance was stopped abruptly.

Pending: The instance has been scheduled but has not yet started running.

Running: The instance has started running.

ContinuedAsNew: The instance has restarted itself with a new history. This state is a transient state.

Box 3: Input -

Invalid input must be available to the monitoring application.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-functions/durable/durable-functions-entities> <https://docs.microsoft.com/en-us/azure/azure-functions/durable/durable-functions-instance-management>

Question #44

Topic 2

You are developing an Azure Durable Function to manage an online ordering process.

The process must call an external API to gather product discount information.

You need to implement the Azure Durable Function.

Which Azure Durable Function types should you use? Each correct answer presents part of the solution.

NOTE: Each correct selection is worth one point.

- A. Orchestrator
- B. Entity
- C. Client
- D. Activity

Correct Answer: AB

The Durable Functions extension exposes a set of built-in HTTP APIs that can be used to perform management tasks on orchestrations, entities, and task hubs.

These HTTP APIs are extensibility webhooks that are authorized by the Azure Functions host but handled directly by the Durable Functions extension.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-functions/durable/durable-functions-http-api>

Community vote distribution

AD (88%)

13%

DRAG DROP -

You are authoring a set of nested Azure Resource Manager templates to deploy multiple Azure resources.

The templates must be tested before deployment and must follow recommended practices.

You need to validate and test the templates before deployment.

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

Tools	Answer Area	Requirement	Tool
Parameter file		Determine whether the templates follow recommended practices.	Tool
Template function			Tool
Azure Resource Manager test toolkit			
User-defined function		Test and validate changes that templates will make to the environment.	
What-if operation			
Azure Deployment Manager			

Correct Answer:

Tools	Answer Area	Requirement	Tool
Parameter file		Determine whether the templates follow recommended practices.	Azure Resource Manager test toolkit
Template function			
Azure Resource Manager test toolkit			
User-defined function		Test and validate changes that templates will make to the environment.	What-if operation
What-if operation			
Azure Deployment Manager			

Box 1: Azure Resource Manager test toolkit

Use ARM template test toolkit -

The Azure Resource Manager template (ARM template) test toolkit checks whether your template uses recommended practices. When your template isn't compliant with recommended practices, it returns a list of warnings with the suggested changes. By using the test toolkit, you can learn how to avoid common problems in template development.

Box 2: What-if operation -**ARM template deployment what-if operation**

Before deploying an Azure Resource Manager template (ARM template), you can preview the changes that will happen. Azure Resource Manager provides the what-if operation to let you see how resources will change if you deploy the template. The what-if operation doesn't make any changes to existing resources.

Instead, it predicts the changes if the specified template is deployed.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-resource-manager/templates/test-toolkit> <https://docs.microsoft.com/en-us/azure/azure-resource-manager/templates/deploy-what-if>

You develop Azure Durable Functions to manage vehicle loans.

The loan process includes multiple actions that must be run in a specified order. One of the actions includes a customer credit check process, which may require multiple days to process.

You need to implement Azure Durable Functions for the loan process.

Which Azure Durable Functions type should you use?

- A. orchestrator
- B. client
- C. entity
- D. activity

Correct Answer: A

Community vote distribution

A (100%)

HOTSPOT

You are developing an Azure Function app.

All functions in the app meet the following requirements:

- Run until either a successful run or until 10 run attempts occur.
- Ensure that there are at least 20 seconds between attempts for up to 15 minutes.

You need to configure the host.json file.

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.

Answer Area

```
{  
  "retry": {  
    "strategy": "exponentialBackoff",  
    "counterThreshold": 10,  
    "fixedDelay": 20000,  
    "maxRetryCount": 10,  
    "healthCheckInterval": 10000,  
    "healthCheckThreshold": 10000  
  },  
  "maximumInterval": "00:15:00",  
  "minimumInterval": "00:00:20"  
}
```

Answer Area

```
{  
  "retry",  
  "healthMonitor",  
  "singleton"  
}  
"strategy": "exponentialBackoff",  
  "counterThreshold",  
  "fixedDelay"  
}  
  "maxRetryCount",  
  "healthCheckInterval",  
  "healthCheckThreshold"  
}  
  "minimumInterval": "00:00:20",  
  "maximumInterval": "00:15:00"  
}  
}
```

Correct Answer:

Question #48

Topic 2

You develop Azure Web Apps for a commercial diving company. Regulations require that all divers fill out a health questionnaire every 15 days after each diving job starts.

You need to configure the Azure Web Apps so that the instance count scales up when divers are filling out the questionnaire and scales down after they are complete.

You need to configure autoscaling.

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

NOTE: Each correct selection is worth one point.

- A. Recurrence profile
- B. CPU usage-based autoscaling
- C. Fixed date profile
- D. Predictive autoscaling

Correct Answer: CD

Community vote distribution

AC (58%)

BD (26%)

AB (16%)

HOTSPOT

You are developing an online game that allows players to vote for their favorite photo that illustrates a word. The game is built by using Azure Functions and uses durable entities to track the vote count.

The voting window is 30 seconds. You must minimize latency.

You need to implement the Azure Function for voting.

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

Answer Area

```
[FunctionName("Vote")]
public static async Task<HttpResponseMessage> Run(
    [HttpTrigger("POST", Route = "pic/{id}")] HttpRequestMessage req,
    [DurableClient] IDurableEntityClient c,
    string id)
{
    var eid = new EntityId("pic", id);
    await c.
        (eid, "vote");
    return req.CreateResponse(HttpStatusCode.OK);
}
```

Answer Area

```
[FunctionName("Vote")]
public static async Task<HttpResponseMessage> Run(
    [HttpTrigger("POST", Route = "pic/{id}")] HttpRequestMessage req,
    [DurableClient] IDurableEntityClient c,
    [DurableClient] IDurableOrchestrationClient
)
{
    string id)

    var eid = new EntityId("pic", id);

    await c.
        (eid, "vote");

    return req.CreateResponse(HttpStatusCode.OK);
}
```

Correct Answer:

Question #1

Topic 3

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>

Question #2

Topic 3

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>

Community vote distribution

B (100%)

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 99.9% 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 \  
--resource-group $resourceGroupName \  
--max-interval 5 \  
--locations 'southcentralus=0 eastus=1 westus=2' \  
--default-consistency-level = $consistencylevel
```

Correct Answer:

--enable-virtual-network true \ --enable-automatic-failover true \ --kind 'GlobalDocumentDB' \ --kind 'MongoDB' \ --locations 'southcentralus=0 eastus=1 westus=2'
--

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>

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

The image shows three separate dropdown menus, each with a downward arrow icon in the top right corner. The first dropdown menu is associated with the command 'az appservice plan create --name websitePlan'. It contains four options: '--sku SHARED', '--tags container', '--sku B1 --hyper-v', and '--sku B1 --is-linux'. The second dropdown menu is associated with the command 'az webapp create --plan websitePlan'. It contains four options: '--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', and '--deployment-container-image-name images.azurecr.io/website:latest'. The third dropdown menu is associated with the command 'az webapp config'. It contains five options: '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', and '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
    -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
```

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.

Paremeter: --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>

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 x

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 i

▼

- 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

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

FetchAttributesAsync
UploadFileStream

Correct Answer:

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>

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>

Community vote distribution

B (100%)

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>

Community vote distribution

A (100%)

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	
Correct Answer: 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>

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>

Community vote distribution

C (100%)

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>

Community vote distribution

A (100%)

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`

Code segment 2

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

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`

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 = " [] " ;`
`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`
`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>

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

orderBy
sortOrder
ascending
descending
compositeIndexes

Answer Area

```
{  
    "automatic": true,  
    "ngMode": "Consistent",  
    "includedPaths": [  
        {  
            "path": "/**"  
        }  
    ], "excludedPaths": [],  
    " [ ]": [  
        [  
            {  
                "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",  
            }  
        ]  
    ]  
}
```

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="▼"/>
	3
	4
	6
	12
Partition Key	<input type="button" value="▼"/>
	Highway
	Department
	Timestamp
	VM name

Answer Area

Setting	Value
Number of partitions	<input type="button" value="▼"/>
Correct Answer:	3
	4
	6
	12
Partition Key	<input type="button" value="▼"/>
Box 1: 6 -	Highway
The number of partitions is specified at creation and must be between 2 and 32.	Department
There are 6 highways.	Timestamp
	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>

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

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
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>

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.



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.

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.

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>.azurededge.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>

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>

Community vote distribution

DE (100%)

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.

Container1 will contain two items.

Container2 will contain one item.

Answer Area

	Statements	Yes	No
Correct Answer:	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>

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**

Store the data from which the change feed is generated.

Component

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**

Store the data from which the change feed is generated.

Component

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>

HOTSPOT -

You are developing a web application that will use Azure Storage. Older data will be less frequently used than more recent data.

You need to configure data storage for the application. You have the following requirements:

- ⇒ Retain copies of data for five years.
- ⇒ Minimize costs associated with storing data that is over one year old.
- ⇒ Implement Zone Redundant Storage for application data.

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

Requirement	Solution
Configure an Azure Storage account	<div style="border: 1px solid black; padding: 5px; height: 150px; width: 100%;"><ul style="list-style-type: none">Implement Blob StorageImplement Azure Cosmos DBImplement Storage (general purpose v1)Implement StorageV2 (general purpose v2)</div>
Configure data retention	<div style="border: 1px solid black; padding: 5px; height: 150px; width: 100%;"><ul style="list-style-type: none">Snapshot blobs and move them to the archive tierSet a lifecycle management policy to move blobs to the cool tierUse AzCopy to copy the data to an on-premises device for backupSet a lifecycle management policy to move blobs to the archive tier</div>

Correct Answer:**Answer Area**

Requirement	Solution
Configure an Azure Storage account	<div style="border: 1px solid black; padding: 5px; height: 150px; width: 100%;"><ul style="list-style-type: none">Implement Blob StorageImplement Azure Cosmos DBImplement Storage (general purpose v1)Implement StorageV2 (general purpose v2)</div>
Configure data retention	<div style="border: 1px solid black; padding: 5px; height: 150px; width: 100%;"><ul style="list-style-type: none">Snapshot blobs and move them to the archive tierSet a lifecycle management policy to move blobs to the cool tierUse AzCopy to copy the data to an on-premises device for backupSet a lifecycle management policy to move blobs to the archive tier</div>

Reference:

<https://docs.microsoft.com/en-us/azure/storage/blobs/storage-blob-storage-tiers> <https://docs.microsoft.com/en-us/azure/storage/common/storage-redundancy?toc=/azure/storage/blobs/toc.json>

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 , TableQuery.GenerateFilterCondition (Email, QueryComparisons.Equal , "ssmith@contoso.com")
07 ));
08 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.

SaveScore will update and replace a record if one already exists with the same playerId and gameId.

Leader board data for the game will be automatically partitioned using gameId.

SaveScore will store the values for the gameId and playerId parameters in the database.

Answer Area

	Statements	Yes	No
	SaveScore will work with Cosmos DB.	<input checked="" type="radio"/>	<input type="radio"/>
Correct Answer:	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>

You develop and deploy a web application to Azure App Service. The application accesses data stored in an Azure Storage account. The account contains several containers with several blobs with large amounts of data. You deploy all Azure resources to a single region. You need to move the Azure Storage account to the new region. You must copy all data to the new region. What should you do first?

- A. Export the Azure Storage account Azure Resource Manager template
- B. Initiate a storage account failover
- C. Configure object replication for all blobs
- D. Use the AzCopy command line tool
- E. Create a new Azure Storage account in the current region
- F. Create a new subscription in the current region

Correct Answer: A

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 and finally, delete the resources in the source region.

To get started, export, and then modify a Resource Manager template.

Reference:

<https://docs.microsoft.com/en-us/azure/storage/common/storage-account-move?tabs=azure-portal>

Community vote distribution

A (100%)

HOTSPOT -

You are developing an application to collect the following telemetry data for delivery drivers: first name, last name, package count, item id, and current location coordinates. The app will store the data in Azure Cosmos DB.

You need to configure Azure Cosmos DB to query the data.

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

Configuration Parameter	Value
Azure Cosmos DB API	<input type="checkbox"/>
	Gremlin
	Table API
	Core (SQL)
Azure Cosmos DB partition key	<input type="checkbox"/>
	first name
	last name
	package count
	item id

Answer Area

Configuration Parameter	Value
Azure Cosmos DB API	<input type="checkbox"/>
	Gremlin
	Table API
Correct Answer:	Core (SQL)
Azure Cosmos DB partition key	<input type="checkbox"/>
	first name
	last name
	package count
	item id

Box 1: Core (SQL)

Core(SQL) API stores data in document format. It offers the best end-to-end experience as we have full control over the interface, service, and

the SDK client libraries. SQL API supports analytics and offers performance isolation between operational and analytical workloads.

Box 2: item id -

item id is a unique identifier and is suitable for the partition key.

Reference:

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

<https://docs.microsoft.com/en-us/azure/cosmos-db/partitioning-overview>

Question #25

Topic 3

DRAG DROP -

You are implementing an Azure solution that uses Azure Cosmos DB and the latest Azure Cosmos DB SDK. You add a change feed processor to a new container instance.

You attempt to read a batch of 100 documents. The process fails when reading one of the documents. The solution must monitor the progress of the change feed processor instance on the new container as the change feed is read. You must prevent the change feed processor from retrying the entire batch when one document cannot be read.

You need to implement the change feed processor to read the documents.

Which features should you use? To answer, drag the appropriate features to the cored 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 cored selection is worth one point.

Select and Place:

Features	Answer Area	
	Requirement	Feature
Change feed estimator	Monitor the progress of the change feed processor	
Dead-letter queue	Prevent the change feed processor from retrying the entire batch when one document cannot be read	
Deployment unit		
Lease container		

Correct Answer:

Features	Answer Area	
	Requirement	Feature
Change feed estimator	Monitor the progress of the change feed processor	Change feed estimator
Dead-letter queue	Prevent the change feed processor from retrying the entire batch when one document cannot be read	Dead-letter queue
Deployment unit		
Lease container		

Box 1: Change feed estimator -

You can use the change feed estimator to monitor the progress of your change feed processor instances as they read the change feed or use the life cycle notifications to detect underlying failures.

Box 2: Dead-letter queue -

To prevent your change feed processor from getting "stuck" continuously retrying the same batch of changes, you should add logic in your delegate code to write documents, upon exception, to a dead-letter queue. This design ensures that you can keep track of unprocessed changes while still being able to continue to process future changes. The dead-letter queue might be another Cosmos container. The exact data store does not matter, simply that the unprocessed changes are persisted.

Reference:

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

HOTSPOT -

You are developing an application that uses a premium block blob storage account. The application will process a large volume of transactions daily. You enable Blob storage versioning.

You are optimizing costs by automating Azure Blob Storage access tiers. You apply the following policy rules to the storage account. (Line numbers are included for reference only.)

```
01 {
02   "rules" : [
03     {
04       "name" : "versionRule",
05       "enabled" : true,
06       "type" : "Lifecycle",
07       "definition" : {
08         "actions" : {
09           "version" : {
10             "tierToCool" : {
11               "daysAfterCreationGreaterThanOrEqual" : 60
12             },
13             "delete" : {
14               "daysAfterCreationGreaterThanOrEqual" : 365
15             }
16           },
17         },
18         "filters" : {
19           "blobTypes" : [ "blockBlob" ], "prefixMatch" : [ "transactions" ]
20         }
21       }
22     }
23   ]
24 }
```

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
Block blobs prefixed with transactions will transition blobs that have not been modified in over 60 days to cool storage, and delete blobs not modified in 365 days	<input type="radio"/>	<input type="radio"/>
Blobs are moved to cool storage if they have not been accessed for 60 days	<input type="radio"/>	<input type="radio"/>
The policy rule tiers previous versions within a container named transactions that are 60 days or older to the cool tier and deletes previous versions that are 365 days or older	<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"/>

Correct Answer:

Answer Area

Statements	Yes	No
Block blobs prefixed with transactions will transition blobs that have not been modified in over 60 days to cool storage, and delete blobs not modified in 365 days	<input type="radio"/>	<input checked="" type="radio"/>
Blobs are moved to cool storage if they have not been accessed for 60 days	<input type="radio"/>	<input checked="" type="radio"/>
The policy rule tiers previous versions within a container named transactions that are 60 days or older to the cool tier and deletes previous versions that are 365 days or older	<input checked="" type="radio"/>	<input type="radio"/>
Blobs will automatically be tiered from cool back to hot if accessed again after being tiered to cool	<input checked="" type="radio"/>	<input type="radio"/>

Box 1: No -

Would be true if `daysAfterModificationGreater Than` was used, but here `daysAfterCreationGreater Than`

Box 2: No -

Would need to use the `daysAfterLastAccessTimeGreater Than` predicate.

Box 3: Yes -

Box 4: Yes -

With the lifecycle management policy, you can:

Transition blobs from cool to hot immediately when they are accessed, to optimize for performance.

Reference:

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

Question #27

Topic 3

An organization deploys Azure Cosmos DB.

You need to ensure that the index is updated as items are created, updated, or deleted.

What should you do?

- A. Set the indexing mode to Lazy.
- B. Set the value of the automatic property of the indexing policy to False.
- C. Set the value of the `EnableScanInQuery` option to True.
- D. Set the indexing mode to Consistent.

Correct Answer: D

Azure Cosmos DB supports two indexing modes:

Consistent: The index is updated synchronously as you create, update or delete items. This means that the consistency of your read queries will be the consistency configured for the account.

None: Indexing is disabled on the container.

Reference:

<https://docs.microsoft.com/en-us/azure/cosmos-db/index-policy>

Community vote distribution

D (100%)

You are developing a .Net web application that stores data in Azure Cosmos DB. The application must use the Core API and allow millions of reads and writes.

The Azure Cosmos DB account has been created with multiple write regions enabled. The application has been deployed to the East US2 and Central US regions.

You need to update the application to support multi-region writes.

What are two possible ways to achieve this goal? Each correct answer presents part of the solution.

NOTE: Each correct selection is worth one point.

- A. Update the ConnectionPolicy class for the Cosmos client and populate the PreferredLocations property based on the geo-proximity of the application.
- B. Update Azure Cosmos DB to use the Strong consistency level. Add indexed properties to the container to indicate region.
- C. Update the ConnectionPolicy class for the Cosmos client and set the UseMultipleWriteLocations property to true.
- D. Create and deploy a custom conflict resolution policy.
- E. Update Azure Cosmos DB to use the Session consistency level. Send the SessionToken property value from the FeedResponse object of the write action to the end-user by using a cookie.

Correct Answer: CD

C: The UseMultipleWriteLocations of the ConnectionPolicy class gets or sets the flag to enable writes on any locations (regions) for geo-replicated database accounts in the Azure Cosmos DB service.

Note: Once an account has been created with multiple write regions enabled, you must make two changes in your application to the ConnectionPolicy for the

Cosmos client to enable the multi-region writes in Azure Cosmos DB. Within the ConnectionPolicy, set UseMultipleWriteLocations to true and pass the name of the region where the application is deployed to ApplicationRegion. This will populate the PreferredLocations property based on the geo-proximity from location passed in. If a new region is later added to the account, the application does not have to be updated or redeployed, it will automatically detect the closer region and will auto-home on to it should a regional event occur.

Azure core API application " ConnectionPolicy class" cosmos db multiple write regions enabled

D: With multi-region writes, when multiple clients write to the same item, conflicts may occur. When a conflict occurs, you can resolve the conflict by using different conflict resolution policies.

Note: Conflict resolution policy can only be specified at container creation time and cannot be modified after container creation.

Reference:

<https://docs.microsoft.com/en-us/dotnet/api/microsoft.azure.documents.client.connectionpolicy> <https://docs.microsoft.com/en-us/azure/cosmos-db/sql/how-to-multi-master> <https://docs.microsoft.com/en-us/azure/cosmos-db/sql/how-to-manage-conflicts>

Community vote distribution

AC (78%)

CD (22%)

HOTSPOT -

You are developing a solution to store documents in Azure Blob storage. Customers upload documents to multiple containers. Documents consist of PDF, CSV,

Microsoft Office format and plain text files.

The solution must process millions of documents across hundreds of containers. The solution must meet the following requirements:

- ⇒ Documents must be categorized by a customer identifier as they are uploaded to the storage account.
- ⇒ Allow filtering by the customer identifier.
- ⇒ Allow searching of information contained within a document
- ⇒ Minimize costs.

You create and configure a standard general-purpose v2 storage account to support the solution.

You need to implement the solution.

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

NOTE: Each correct selection is worth one point.

Hot Area:

Answer Area

Requirement

Search and filter by customer identifier

Solution

	▼
Azure Cognitive Search	
Azure Blob index tags	
Azure Blob inventory policy	
Azure Blob metadata	

Search information inside documents

	▼
Azure Cognitive Search	
Azure Blob index tags	
Azure Blob inventory policy	
Azure Blob metadata	

Correct Answer:

Answer Area

Requirement

Search and filter by customer identifier

Solution

Azure Cognitive Search
Azure Blob index tags
Azure Blob inventory policy
Azure Blob metadata

Search information inside documents

Azure Cognitive Search
Azure Blob index tags
Azure Blob inventory policy
Azure Blob metadata

Box 1: Azure Blob index tags -

As datasets get larger, finding a specific object in a sea of data can be difficult. Blob index tags provide data management and discovery capabilities by using key-value index tag attributes. You can categorize and find objects within a single container or across all containers in your storage account. As data requirements change, objects can be dynamically categorized by updating their index tags. Objects can remain in-place with their current container organization.

Box 2: Azure Cognitive Search -

Only index tags are automatically indexed and made searchable by the native Blob Storage service. Metadata can't be natively indexed or searched. You must use a separate service such as Azure Search.

Azure Cognitive Search is the only cloud search service with built-in AI capabilities that enrich all types of information to help you identify and explore relevant content at scale. Use cognitive skills for vision, language, and speech, or use custom machine learning models to uncover insights from all types of content.

Reference:

<https://docs.microsoft.com/en-us/azure/storage/blobs/storage-manage-find-blobs> <https://azure.microsoft.com/en-us/services/search/>

HOTSPOT -

You are developing a web application by using the Azure SDK. The web application accesses data in a zone-redundant BlockBlobStorage storage account.

The application must determine whether the data has changed since the application last read the data. Update operations must use the latest data changes when writing data to the storage account.

You need to implement the update operations.

Which values should you use? To answer, select the appropriate option in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

Answer Area

Code evaluation	Value
HTTP Header value	<input type="checkbox"/> ETag <input type="checkbox"/> Last Modified <input type="checkbox"/> VersionId
Conditional header	<input type="checkbox"/> If-Match <input type="checkbox"/> If-Modified-Since <input type="checkbox"/> If-None-Match

Answer Area

Code evaluation	Value
HTTP Header value	<input type="checkbox"/> ETag <input checked="" type="checkbox"/> Last Modified <input type="checkbox"/> VersionId
Conditional header	<input type="checkbox"/> If-Match <input checked="" type="checkbox"/> If-Modified-Since <input type="checkbox"/> If-None-Match

Box 1: Last Modified -

The Last-Modified response HTTP header contains a date and time when the origin server believes the resource was last modified. It is used as a validator to determine if the resource is the same as the previously stored one. Less accurate than an ETag header, it is a fallback mechanism.

Box 2: If-Modified-Since -

Conditional Header If-Modified-Since:

A DateTime value. Specify this header to perform the operation only if the resource has been modified since the specified time.

Incorrect:

Not ETag/If-Match -

Conditional Header If-Match:

An ETag value. Specify this header to perform the operation only if the resource's ETag matches the value specified. For versions 2011-08-18 and newer, the

ETag can be specified in quotes.

Reference:

<https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Last-Modified> <https://docs.microsoft.com/en-us/rest/api/storageservices/specifying-conditional-headers-for-blob-service-operations>

Question #31

Topic 3

HOTSPOT -

An organization deploys a blob storage account. Users take multiple snapshots of the blob storage account over time.

You need to delete all snapshots of the blob storage account. You must not delete the blob storage account itself.

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

```
Delete (Azure.Storage.Blobs.Models.DeleteSnapshotsOption  
snapshotsOption = Azure.Storage.Blobs.Models.
```

DeleteIfExists
DeleteSnapshotsOption
WithSnapshot
WithSnapshotCore

IncludeSnapshots
None
OnlySnapshots

Correct Answer:

Answer Area

```
Delete (Azure.Storage.Blobs.Models.DeleteSnapshotsOption  
snapshotsOption = Azure.Storage.Blobs.Models.
```

DeleteIfExists
DeleteSnapshotsOption
WithSnapshot
WithSnapshotCore

IncludeSnapshots
None
OnlySnapshots

Box 1: DeleteSnapshotsOption -

Sample code in powershell:

```
//dont forget to add the include snapshots :)  
await batchClient.DeleteBlobsAsync(listofURIforBlobs,  
Azure.Storage.Blobs.Models.DeleteSnapshotsOption.IncludeSnapshots);
```

Sample code in .Net:

```
// Create a batch with three deletes  
BlobBatchClient batchClient = service.GetBlobBatchClient();  
BlobBatch batch = batchClient.CreateBatch();  
batch.DeleteBlob(foo.Uri, DeleteSnapshotsOption.IncludeSnapshots); batch.DeleteBlob(bar.Uri, DeleteSnapshotsOption.OnlySnapshots);  
batch.DeleteBlob(baz.Uri);  
// Submit the batch  
batchClient.SubmitBatch(batch);
```

Box 2: OnlySnapshots -

Reference:

<https://docs.microsoft.com/en-us/dotnet/api/overview/azure/storage.blobs.batch-readme>

<https://stackoverflow.com/questions/39471212/programmatically-delete-azure-blob-storage-objects-in-bulks>

HOTSPOT -

An organization deploys a blob storage account. Users take multiple snapshots of the blob storage account over time.

You need to delete all snapshots of the blob storage account. You must not delete the blob storage account itself.

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

<code>delete_blob (</code>	<code>)</code>
<input type="button" value="delete_container"/>	<input type="button" value="False"/>
<input type="button" value="delete_snapshots"/>	<input type="button" value="Include"/>
<input type="button" value="snapshot_blob"/>	<input type="button" value="Only"/>
<input type="button" value="snapshots_present"/>	

Correct Answer:

Answer Area

<code>delete_blob (</code>	<code>)</code>
<input type="button" value="delete_container"/>	<input type="button" value="False"/>
<input checked="" type="button" value="delete_snapshots"/>	<input checked="" type="button" value="Include"/>
<input type="button" value="snapshot_blob"/>	<input checked="" type="button" value="Only"/>
<input type="button" value="snapshots_present"/>	

Box 1: `delete_snapshots` -

Delete only the snapshot (blob itself is retained)

`blob_client.delete_blob(delete_snapshots="only")`

Box 2: `only` -

Reference:

https://github.com/Azure/azure-sdk-for-python/blob/main/sdk/storage/azure-storage-blob/samples/blob_samples_common.py

HOTSPOT

You are developing an application that monitors data added to an Azure Blob storage account.

You need to process each change made to the storage account.

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.

Answer Area

```
var changeFeedClient = new BlobServiceClient("...").GetChangeFeedClient();
var x = default(string);
while (true)
{
    var changeFeed = changeFeedClient. 
        GetChanges()
        GetChangesAsync()
        GetChanges(x).AsPages()
        GetChanges(x).GetEnumerator()

    foreach (var c in changeFeed)
    {
        x = c. 
            ContinuationToken
            GetRawResponse().ReasonPhrase
            Values.Max(x => x.EventTime).ToString()
            Values.Min(x => x.EventTime).ToString()

        ProcessChanges(c.Values);
    }
}
```

Answer Area

```
var changeFeedClient = new BlobServiceClient("...").GetChangeFeedClient();
var x = default(string);
while (true)
{
    var changeFeed = changeFeedClient. 
        GetChanges()
        GetChangesAsync()
        GetChanges(x).AsPages()
        GetChanges(x).GetEnumerator()
```

Correct Answer:

```
foreach (var c in changeFeed)
{
    x = c. 
        ContinuationToken
        GetRawResponse().ReasonPhrase
        Values.Max(x => x.EventTime).ToString()
        Values.Min(x => x.EventTime).ToString()

    ProcessChanges(c.Values);
}
```

HOTSPOT

You develop an application that sells AI generated images based on user input. You recently started a marketing campaign that displays unique ads every second day.

Sales data is stored in Azure Cosmos DB with the date of each sale being stored in a property named 'whenFinished'.

The marketing department requires a view that shows the number of sales for each unique ad.

You need to implement the query for the view.

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

NOTE: Each correct selection is worth one point.

```
SELECT
    max(c.whenFinished)
    sum(c.whenFinished)
    count(c.whenFinished)

    DateTimeBin(c.whenFinished, 'day', 2)
    DateTimePart(c.whenFinished, 'day', 2)
    DateTimeBin(c.whenFinished, 'hour', 12)
    DateTimePart(c.whenFinished, 'hour', 12)

FROM c
group by
    DateTimeBin(c.whenFinished, 'day', 2)
    DateTimePart(c.whenFinished, 'day', 2)
    DateTimeBin(c.whenFinished, 'hour', 12)
    DateTimePart(c.whenFinished, 'hour', 12)
```

Correct Answer:

```
SELECT
    max(c.whenFinished)
    sum(c.whenFinished)
    count(c.whenFinished)

    DateTimeBin(c.whenFinished, 'day', 2)
    DateTimePart(c.whenFinished, 'day', 2)
    DateTimeBin(c.whenFinished, 'hour', 12)
    DateTimePart(c.whenFinished, 'hour', 12)

FROM c
group by
    DateTimeBin(c.whenFinished, 'day', 2)
    DateTimePart(c.whenFinished, 'day', 2)
    DateTimeBin(c.whenFinished, 'hour', 12)
    DateTimePart(c.whenFinished, 'hour', 12)
```

HOTSPOT

You implement an Azure solution to include Azure Cosmos DB, the latest Azure Cosmos DB SDK, and the Core (SQL) API. You also implement a change feed processor on a new container instance by using the Azure Functions trigger for Azure Cosmos DB.

A large batch of documents continues to fail when reading one of the documents in the batch. The same batch of documents is continuously retried by the triggered function and a new batch of documents must be read.

You need to implement the change feed processor to read the documents.

Which feature should you implement? To answer, select the appropriate features in the answer area.

NOTE: Each correct selection is worth one point.

Requirement

Read a new batch of documents while keeping track of the failing batch of documents.

Feature

- Lease container
- Dead-letter queue
- Life-cycle notifications
- Change feed estimator

Handle errors in the change feed processor.

- Lease container
- Dead-letter queue
- Life-cycle notifications
- Change feed estimator

Requirement

Read a new batch of documents while keeping track of the failing batch of documents.

Feature

- Lease container
- Dead-letter queue
- Life-cycle notifications
- Change feed estimator

Correct Answer:

Handle errors in the change feed processor.

- Lease container
- Dead-letter queue**
- Life-cycle notifications
- Change feed estimator

You are developing an application to store business-critical data in Azure Blob storage.

The application must meet the following requirements:

- Data must not be modified or deleted for a user-specified interval.
- Data must be protected from overwrites and deletes.
- Data must be written once and allowed to be read many times.

You need to protect the data in the Azure Blob storage account.

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

NOTE: Each correct selection is worth one point.

- A. Configure a time-based retention policy for the storage account.
- B. Create an account shared-access signature (SAS).
- C. Enable the blob change feed for the storage account.
- D. Enable version-level immutability support for the storage account.
- E. Enable point-in-time restore for containers in the storage account.
- F. Create a service shared-access signature (SAS).

Correct Answer: AF

Community vote distribution

AD (91%) 9%

You are updating an application that stores data on Azure and uses Azure Cosmos DB for storage. The application stores data in multiple documents associated with a single username.

The application requires the ability to update multiple documents for a username in a single ACID operation.

You need to configure Azure Cosmos DB.

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

NOTE: Each correct selection is worth one point.

- A. Create a collection sharded on username to store documents.
- B. Configure Azure Cosmos DB to use the Gremlin API.
- C. Create an unsharded collection to store documents.
- D. Configure Azure Cosmos DB to use the MongoDB API.

Correct Answer: CD

Community vote distribution

CD (100%)

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.

```
database_name = 'MyDatabase'  
database = client.create_database_if_not_exists(id=database_name)
```
- B.

```
client = CosmosClient(endpoint, key)
```
- C.

```
container_name = 'MyContainer'  
container = database.create_container_if_not_exists(  
id=container_name, partition_key=PartitionKey(path="/lastName"), offer_throughput=400 )
```

Correct Answer: C

Community vote distribution

B (100%)

You develop a web application that provides access to legal documents that are stored on Azure Blob Storage with version-level immutability policies. Documents are protected with both time-based policies and legal hold policies. All time-based retention policies have the AllowProtectedAppendWrites property enabled.

You have a requirement to prevent the user from attempting to perform operations that would fail only when a legal hold is in effect and when all other policies are expired.

You need to meet the requirement.

Which two operations should you prevent? Each correct answer presents a complete solution.

NOTE: Each correct selection is worth one point.

- A. adding data to documents
- B. deleting documents
- C. creating documents
- D. overwriting existing documents

Correct Answer: BD

Community vote distribution

BD (100%)

HOTSPOT

You provisioned an Azure Cosmos DB for NoSQL account named account1 with the default consistency level.

You plan to configure the consistency level on a per request basis. The level needs to be set for consistent prefix for read and write operations to account1.

You need to identify the resulting consistency level for read and write operations.

Which levels should you configure? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Answer Area

Operation type	Resulting consistency level
Read operations	<input type="checkbox"/> strong <input type="checkbox"/> session <input checked="" type="checkbox"/> consistent prefix
Write operations	<input type="checkbox"/> strong <input type="checkbox"/> session <input checked="" type="checkbox"/> consistent prefix

Answer Area

Operation type	Resulting consistency level
Read operations	<input type="checkbox"/> strong <input type="checkbox"/> session <input checked="" type="checkbox"/> consistent prefix
Write operations	<input type="checkbox"/> strong <input checked="" type="checkbox"/> session <input type="checkbox"/> consistent prefix

DRAG DROP

You are developing an application to store millions of images in Azure blob storage. The images are uploaded to an Azure blob storage container named companyimages contained in an Azure blob storage account named companymedia. The stored images are uploaded with multiple blob index tags across multiple blobs in the container.

You must find all blobs whose tags match a search expression in the container. The search expression must evaluate an index tag named status with a value of final.

You need to construct the GET method request URI.

How should you complete the URI? To answer, drag the appropriate parameters to the correct request URI targets. Each parameter 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.

Parameters

- Status='Final'
- Status <= 'Final'
- companymedia
- companyimages

Answer Area

`https:// [] .blob.core.windows.net/ [] ?restype=container&comp=blobs&where= []`

Answer Area**Correct Answer:**

`https:// [] .blob.core.windows.net/ [] ?restype=container&comp=blobs&where= [] Status='Final'`

Question #1

Topic 4

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/>

Community vote distribution

C (100%)

Question #2

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 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/>

Community vote distribution

B (75%)

A (25%)

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/>

Community vote distribution

A (100%)

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/>

Community vote distribution

A (70%)

B (30%)

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
Access policy
Purge protection
Soft delete
Shared access signature

Answer Area

Action	Feature
Enable retention period and accidental deletion.	Feature
Enforce retention period and accidental deletion.	Feature

Correct Answer:

Features
Access policy
Purge protection
Soft delete
Shared access signature

Answer Area

Action	Feature
Enable retention period and accidental deletion.	Soft delete
Enforce retention period and accidental deletion.	Purge protection

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>

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

Community vote distribution

AC (61%)

AD (39%)

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:

Settings
client_id
profile
delegated
application
user_impersonation

Answer Area

API	Permission	Type
Azure Storage	Setting	Setting
Microsoft Graph	User.Read	Setting

Correct Answer:

Settings
client_id
profile
delegated
application
user_impersonation

Answer Area

API	Permission	Type
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/>

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.      ();  


|                   |
|-------------------|
| UseAuthentication |
| UseStaticFiles    |
| UseSession        |
| UseCookiePolicy   |



    app.      ();  


|                     |
|---------------------|
| UseAuthorization    |
| UseHttpsRedirection |
| UseSession          |
| UseCookiePolicy     |



    app.      ();  


|                          |
|--------------------------|
| 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>

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>

Community vote distribution

A (79%)

C (21%)

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

Community vote distribution

A (92%)

8%

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 from in Azure Storage Blob storage.

Community vote distribution

B (80%)

A (20%)

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 from in Azure Storage Blob storage.

Community vote distribution

B (100%)

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
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
```

Correct Answer:

```
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
```

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>

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>

Community vote distribution

C (100%)

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**Answer Area**

```
$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
```

Correct Answer:**Powershell commands****Answer Area**

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

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



```
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-a/Azure/key-vault/key-vault-key-rotation-log-monitoring>

Question #16

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 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/bs-latn-a/Azure/active-directory/managed-identities-azure-resources/tutorial-windows-vm-access-arm>

Community vote distribution

B (100%)

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>

Community vote distribution

B (80%) A (20%)

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

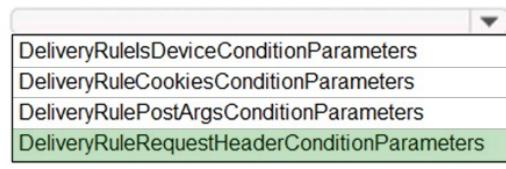
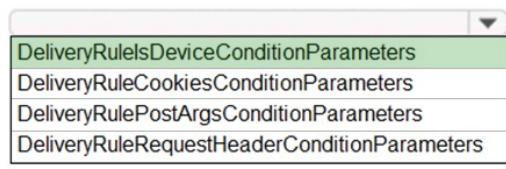
```
"conditions": [ {
    "name": "IsDevice",
    "parameters": {
        "@odata.type": "#Microsoft.Azure.Cdn.Models.
        "operator": "Equal",
        "matchValues": [ " "
            " "
            " "
            " "
        ],
        "selector": " "
            " "
            " "
            " "
    }
}, {
    "name": "RequestHeader",
    "parameters": {
        "@odata.type": "#Microsoft.Azure.Cdn.Models.
        "operator": "Contains",
        "selector": " "
            " "
            " "
            " "
    }
}
],
"matchValues": [ "
    " "
    " "
    " "
    " "
]
```

The screenshot shows the Azure Resource Manager template editor. It displays a JSON configuration for an Azure CDN rule. The 'conditions' section contains two entries. The first entry uses the 'IsDevice' condition with an 'Equal' operator, matching values like 'iOS', 'Mobile', 'iPhone', and 'Desktop'. The second entry uses the 'RequestHeader' condition with a 'Contains' operator, matching values like 'FROM', 'PRAGMA', 'X-POWERED-BY', and 'HTTP_USER_AGENT'. Both sections have dropdown menus for selecting parameters and operators. To the right of the editor, there are two dropdown menus showing lists of available condition parameters: 'DeliveryRuleIsDeviceConditionParameters' and 'DeliveryRuleRequestHeaderConditionParameters', each containing four items: 'DeliveryRuleIsDeviceConditionParameters', 'DeliveryRuleCookiesConditionParameters', 'DeliveryRulePostArgsConditionParameters', and 'DeliveryRuleRequestHeaderConditionParameters'.

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": " ",  
            " ",  
            " ",  
            " " ]  
        } }  
    ]  
}  
]  
"matchValues": [ " ",  
    " ",  
    " ",  
    " " ]
```



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>

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 groups 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/>

Community vote distribution

B (78%) A (22%)

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>

Community vote distribution

A (90%) 10%

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": "All",
  "groupMembershipClaims": true
  ...
}
```

Answer Area

```
{
  ...
  "appId": "d61126e3-089b-4adb-b721-d5023213df7d",
  "displayName": "internal",
  "optionalClaims": "All",
  "groupMembershipClaims": 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 #22

Topic 4

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>

Community vote distribution

B (100%)

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>

Community vote distribution

A (100%)

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>

Community vote distribution

C (70%)

D (30%)

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>

Community vote distribution

D (100%)

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
Correct Answer: 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 be 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>

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	Answer Area
DefaultAzureCredential	<code>string var1 = Environment.GetEnvironmentVariable("KEY_VAULT_URI");</code>
ClientSecretCredential	<code>var var2 = new [] (new Uri(var1), new [] ());</code>
CloudClients	
SecretClient	

Correct Answer:

Code segments	Answer Area
	<code>string var1 = Environment.GetEnvironmentVariable("KEY_VAULT_URI");</code>
ClientSecretCredential	<code>var var2 = new [SecretClient] (new Uri(var1), new [DefaultAzureCredential] ());</code>
CloudClients	

Box 1: SecretClient -

Box 2: DefaultAzureCredential -

In below example, the name of your key vault is expanded to the key vault URL, 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>

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>

Community vote distribution

AB (100%)

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>

Community vote distribution

AD (100%)

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>

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>

Community vote distribution

A (100%)

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

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.

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 #33

Topic 4

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: AC

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>

Community vote distribution

CD (40%) D (20%) A (20%) Other

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

	Yes	No
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**

	Yes	No
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 $0.1 \times \text{Tmax} \leq T \leq \text{Tmax}$.

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>

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();  



```

Correct Answer:**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();  



```

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>

HOTSPOT -

You develop a containerized application. You plan to deploy the application to a new Azure Container instance by using a third-party continuous integration and continuous delivery (CI/CD) utility.

The deployment must be unattended and include all application assets. The third-party utility must only be able to push and pull images from the registry. The authentication must be managed by Azure Active Directory (Azure AD). The solution must use the principle of least privilege.

You need to ensure that the third-party utility can access the registry.

Which authentication 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

Authentication	Option
Registry authentication method	<div style="border: 1px solid black; padding: 5px; width: fit-content;"> Service principal Individual identity Repository-scoped access token Managed identity for Azure resources </div>
RBAC role	<div style="border: 1px solid black; padding: 5px; width: fit-content;"> AcrPull Owner AcrPush Contributor </div>

Answer Area

Authentication	Option
Registry authentication method	<div style="border: 1px solid black; padding: 5px; width: fit-content;"> Service principal Individual identity Repository-scoped access token Managed identity for Azure resources </div>
RBAC role	<div style="border: 1px solid black; padding: 5px; width: fit-content;"> AcrPull Owner AcrPush Contributor </div>

Correct Answer:

- ⇒ Service principal - Applications and container orchestrators can perform unattended, or "headless," authentication by using an Azure Active Directory (Azure AD) service principal.
- ⇒ Incorrect Answers:
 - ⇒ Individual AD identity does not support unattended push/pull
 - ⇒ Repository-scoped access token is not integrated with AD identity
 - ⇒ Managed identity for Azure resources is used to authenticate to an Azure container registry from another Azure resource.

Box 1: Service principal -

Applications and container orchestrators can perform unattended, or "headless," authentication by using an Azure Active Directory (Azure AD) service principal.

Incorrect Answers:

- ⇒ Individual AD identity does not support unattended push/pull
- ⇒ Repository-scoped access token is not integrated with AD identity
- ⇒ Managed identity for Azure resources is used to authenticate to an Azure container registry from another Azure resource.

Box 2: AcrPush -

AcrPush provides pull/push permissions only and meets the principle of least privilege.

Incorrect Answers:

- ⇒ AcrPull only allows pull permissions it does not allow push permissions.
-

⇒ Owner and Contributor allow pull/push permissions but does not meet the principle of least privilege.

Reference:

<https://docs.microsoft.com/en-us/azure/container-registry/container-registry-authentication?tabs=azure-cli> <https://docs.microsoft.com/en-us/azure/container-registry/container-registry-roles?tabs=azure-cli>

Question #37

Topic 4

You deploy an Azure App Service web app. You create an app registration for the app in Azure Active Directory (Azure AD) and Twitter.

The app must authenticate users and must use SSL for all communications. The app must use Twitter as the identity provider.

You need to validate the Azure AD request in the app code.

What should you validate?

- A. ID token header
- B. ID token signature
- C. HTTP response code
- D. Tenant ID

Correct Answer: A

Reference:

<https://docs.microsoft.com/en-us/azure/storage/common/storage-auth-aad-app?tabs=dotnet>

Community vote distribution

B (88%) 12%

Question #38

Topic 4

A development team is creating a new REST API. The API will store data in Azure Blob storage. You plan to deploy the API to Azure App Service.

Developers must access the Azure Blob storage account to develop the API for the next two months. The Azure Blob storage account must not be accessible by the developers after the two-month time period.

You need to grant developers access to the Azure Blob storage account.

What should you do?

- A. Generate a shared access signature (SAS) for the Azure Blob storage account and provide the SAS to all developers.
- B. Create and apply a new lifecycle management policy to include a last accessed date value. Apply the policy to the Azure Blob storage account.
- C. Provide all developers with the access key for the Azure Blob storage account. Update the API to include the Coordinated Universal Time (UTC) timestamp for the request header.
- D. Grant all developers access to the Azure Blob storage account by assigning role-based access control (RBAC) roles.

Correct Answer: A

Reference:

<https://docs.microsoft.com/en-us/azure/storage/common/storage-sas-overview>

Community vote distribution

A (100%)

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.

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 #40

Topic 4

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>

Community vote distribution

BC (100%)

DRAG DROP -

An organization plans to deploy Azure storage services.

You need to configure shared access signature (SAS) for granting access to Azure Storage.

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

SAS types**Account-level****Service-level****User delegation****Answer Area****Requirement**

Delegate access to resources in one or more of the storage services

Delegate access to a resource in a single storage service

Secure a resource by using Azure AD credentials

SAS type**Correct Answer:****SAS types****Account-level****Service-level****User delegation****Answer Area****Requirement**

Delegate access to resources in one or more of the storage services

Delegate access to a resource in a single storage service

Secure a resource by using Azure AD credentials

SAS type**Account-level****Service-level****User delegation****Reference:**

<https://docs.microsoft.com/en-us/azure/storage/common/storage-sas-overview>

HOTSPOT -

You are developing an ASP.NET Core app that includes feature flags which are managed by Azure App Configuration. You create an Azure App Configuration store named AppFeatureflagStore as shown in the exhibit:

Key	Label	State	Description	Last modified	...
Export	Export	Off	On	Ability to export data.	6/11/2020, 9:13:26 ...

You must be able to use the feature in the app by using the following markup:

```
<feature name="Export">
<li class="nav-item">
    <a class="nav-link text-dark" asp-area="" asp-controller="Home" asp-action="Export">Export Data</a>
</li>
</feature>
```

You need to update the app to use the feature flag.

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

Code section	Value
Controller attribute	<input type="checkbox"/> FeatureGate <input type="checkbox"/> Route <input type="checkbox"/> ServiceFilter <input type="checkbox"/> TypeFilter
Startup method	<input type="checkbox"/> AddAzureAppConfiguration <input type="checkbox"/> AddControllersWithViews <input type="checkbox"/> AddUserSecrets
AppConfig endpoint setting	<input type="checkbox"/> https://appfeatureflagstore.azure.net <input type="checkbox"/> https://appfeatureflagstore.vault.azure.net <input type="checkbox"/> https://export.azure.net <input type="checkbox"/> https://export.vault.azure.net

Correct Answer:

Answer Area

Code section	Value
Controller attribute	<div style="background-color: #e0f2e0; padding: 5px;">FeatureGate</div> <div style="border-top: 1px solid black; padding: 2px;">Route</div> <div style="border-top: 1px solid black; padding: 2px;">ServiceFilter</div> <div style="border-top: 1px solid black; padding: 2px;">TypeFilter</div>
Startup method	<div style="background-color: #e0f2e0; padding: 5px;">AddAzureAppConfiguration</div> <div style="border-top: 1px solid black; padding: 2px;">AddControllersWithViews</div> <div style="border-top: 1px solid black; padding: 2px;">AddUserSecrets</div>
AppConfig endpoint setting	<div style="background-color: #e0f2e0; padding: 5px;">https://appfeatureflagstore.azureconfig.io</div> <div style="border-top: 1px solid black; padding: 2px;">https://appfeatureflagstore.vault.azure.net</div> <div style="border-top: 1px solid black; padding: 2px;">https://export.azureconfig.io</div> <div style="border-top: 1px solid black; padding: 2px;">https://export.vault.azure.net</div>

Box 1: FeatureGate -

You can use the FeatureGate attribute to control whether a whole controller class or a specific action is enabled.

Box 2: AddAzureAppConfiguration -

The extension method AddAzureAppConfiguration is used to add the Azure App Configuration Provider.

Box 3: https://appfeatureflagstore.azureconfig.io

You need to request the access token with resource=https://<yourstorename>.azconfig.io

Reference:

<https://docs.microsoft.com/en-us/azure/azure-app-configuration/use-feature-flags-dotnet-core>

<https://csharp.christiannagel.com/2020/05/19/azureappconfiguration/> <https://stackoverflow.com/questions/61899063/how-to-use-azure-app-configuration-rest-api>

HOTSPOT -

You have a single page application (SPA) web application that manages information based on data returned by Microsoft Graph from another company's Azure Active Directory (Azure AD) instance.

Users must be able to authenticate and access Microsoft Graph by using their own company's Azure AD instance.

You need to configure the application manifest for the app registration.

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

NOTE: Each correct selection is worth one point.

Hot Area:

```
{  
  "oauth2AllowImplicitFlow": 

|       |   |
|-------|---|
|       | ▼ |
| add   |   |
| false |   |
| spa   |   |
| true  |   |

,  
  "": [ {  
    "addIns":  
    "orgRestrictions":  
    "availableToOtherTenants":  
    "requiredResourceAccess":  
      "resourceAppId": "00000003-0000-0000-c000-000000000000",  
      "resourceAccess": [ {  
        "id": "24a6cdd6-fab1-4aaf-91b8-3cc8225e90d0",  
        "type": "Scope"  
      } ] },  
    "signInAudience": "

|                                    |   |
|------------------------------------|---|
| All                                | ▼ |
| AzureADMyOrg                       |   |
| AzureADMultipleOrgs                |   |
| AzureADandPersonalMicrosoftAccount |   |

"  
  }]
```

Correct Answer:

```
{  
  "oauth2AllowImplicitFlow": 

|       |
|-------|
| ▼     |
| add   |
| false |
| spa   |
| true  |

,  
  "resourceAccess": [  
    {  
      "id": "24a6cdd6-fab1-4aaf-91b8-3cc8225e90d0",  
      "type": "Scope"  
    }]  
  ],  
  "signInAudience": 

|                                    |
|------------------------------------|
| ▼                                  |
| All                                |
| AzureADMyOrg                       |
| AzureADMultipleOrgs                |
| AzureADandPersonalMicrosoftAccount |

  
}  
}
```

Box 1: true -

The oauth2AllowImplicitFlow attribute 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.

In implicit flow, the app receives tokens directly from the Azure Active Directory (Azure AD) authorize endpoint, without any server-to-server exchange. All authentication logic and session handling is done entirely in the JavaScript client with either a page redirect or a pop-up box.

Box 2: requiredResourceAccess -

With dynamic consent, requiredResourceAccess drives the admin consent experience and the user consent experience for users who are using static consent.

However, this parameter doesn't drive the user consent experience for the general case. resourceAppId is the unique identifier for the resource that the app requires access to. This value should be equal to the appId declared on the target resource app. resourceAccess is an array that lists the OAuth2.0 permission scopes and app roles that the app requires from the specified resource. Contains the id and type values of the specified resources.

Example:

```
"requiredResourceAccess": [  
  {  
    "resourceAppId": "00000002-0000-0000-0000-000000000000",  
    "resourceAccess": [  
      {  
        "id": "311a71cc-e848-46a1-bdf8-97ff7156d8e6",  
        "type": "Scope"  
      }  
    ]  
  }  
],
```

Incorrect Answers:

- ⇒ The legacy attribute availableToOtherTenants is no longer supported.
- ⇒ The addIns attribute defines custom behavior that a consuming service can use to call an app in specific contexts. For example, applications that can render file streams may set the addIns property for its "FileHandler" functionality. This parameter will let services like Microsoft 365 call the application in the context of a document the user is working on.

```

Example:
"addIns": [
{
"id": "968A844F-7A47-430C-9163-07AE7C31D407",
"type": "FileHandler",
"properties": [
{
"key": "version",
"value": "2"
}
]
}
],

```

Box 3: AzureADMyOrg -

The signInAudience attribute specifies what Microsoft accounts are supported for the current application. Supported values are:

- ⇒ AzureADMyOrg - Users with a Microsoft work or school account in my organization's Azure AD tenant (for example, single tenant)
- ⇒ AzureADMultipleOrgs - Users with a Microsoft work or school account in any organization's Azure AD tenant (for example, multi-tenant)
- ⇒ AzureADandPersonalMicrosoftAccount - Users with a personal Microsoft account, or a work or school account in any organization's Azure AD tenant

Reference:

<https://docs.microsoft.com/en-us/azure/active-directory/develop/reference-app-manifest> <https://docs.microsoft.com/en-us/azure/active-directory/develop/v2-oauth2-implicit-grant-flow>

Question #44

Topic 4

You manage a data processing application that receives requests from an Azure Storage queue.

You need to manage access to the queue. You have the following requirements:

- ⇒ Provide other applications access to the Azure queue.
- ⇒ Ensure that you can revoke access to the queue without having to regenerate the storage account keys.
- ⇒ Specify access at the queue level and not at the storage account level.

Which type of shared access signature (SAS) should you use?

- A. Service SAS with a stored access policy
- B. Account SAS
- C. User Delegation SAS
- D. Service SAS with ad hoc SAS

Correct Answer: A

A service SAS is secured with the storage account key. A service SAS delegates access to a resource in only one of the Azure Storage services:

Blob storage,

Queue storage, Table storage, or Azure Files.

Stored access policies give you the option to revoke permissions for a service SAS without having to regenerate the storage account keys.

Incorrect Answers:

Account SAS: Account SAS is specified at the account level. It is secured with the storage account key.

User Delegation SAS: A user delegation SAS applies to Blob storage only.

Reference:

<https://docs.microsoft.com/en-us/azure/storage/common/storage-sas-overview>

Community vote distribution

A (100%)

HOTSPOT -

You are developing an application to store and retrieve data in Azure Blob storage. The application will be hosted in an on-premises virtual machine (VM). The VM is connected to Azure by using a Site-to-Site VPN gateway connection. The application is secured by using Azure Active Directory (Azure AD) credentials.

The application must be granted access to the Azure Blob storage account with a start time, expiry time, and read permissions. The Azure Blob storage account access must use the Azure AD credentials of the application to secure data access. Data access must be able to be revoked if the client application security is breached.

You need to secure the application access to Azure Blob storage.

Which security features 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

Component	Security Feature
Application (Client)	<input type="checkbox"/> Storage Account Access Key <input type="checkbox"/> System-assigned Managed Identity <input type="checkbox"/> Shared access signature (SAS) token
Azure Storage (Server)	<input type="checkbox"/> Stored Access Policy <input type="checkbox"/> User-assigned Managed Identity <input type="checkbox"/> Cross-Origin Resource Sharing (CORS)

Correct Answer:

Answer Area

Component	Security Feature
Application (Client)	<p>Storage Account Access Key</p> <p>System-assigned Managed Identity</p> <p>Shared access signature (SAS) token</p>
Azure Storage (Server)	<p>Stored Access Policy</p> <p>User-assigned Managed Identity</p> <p>Cross-Origin Resource Sharing (CORS)</p>

Box 1: Shared access signature (SAS) token

When your application design requires shared access signatures for access to Blob storage, use Azure AD credentials to create a user delegation SAS when possible for superior security.

Box 2: Stored access policy -

Stored access policies give you the option to revoke permissions for a service SAS without having to regenerate the storage account keys.

A shared access signature can take one of the following two forms:

⇒ Service SAS with stored access policy. A stored access policy is defined on a resource container, which can be a blob container, table, queue, or file share.

The stored access policy can be used to manage constraints for one or more service shared access signatures. When you associate a service SAS with a stored access policy, the SAS inherits the constraints – the start time, expiry time, and permissions – defined for the stored access policy.

⇒ Ad hoc SAS.

Reference:

<https://docs.microsoft.com/en-us/azure/storage/common/storage-sas-overview>

You are building a web application that uses the Microsoft identity platform for user authentication.

You are implementing user identification for the web application.

You need to retrieve a claim to uniquely identify a user.

Which claim type should you use?

- A. aud
- B. nonce
- C. oid
- D. idp

Correct Answer: C

oid -The object identifier for the user in Azure AD. This value is the immutable and non-reusable identifier of the user. Use this value, not email, as a unique identifier for users; email addresses can change. If you use the Azure AD Graph API in your app, object ID is that value used to query profile information.

Incorrect:

Not A: aud - Who the token was issued for. This will be the application's client ID.

Reference:

<https://docs.microsoft.com/en-us/azure/architecture/multitenant-identity/claims>

Community vote distribution

C (100%)

You are developing an Azure Function that calls external APIs by providing an access token for the API. The access token is stored in a secret named token in an

Azure Key Vault named mykeyvault.

You need to ensure the Azure Function can access to the token. Which value should you store in the Azure Function App configuration?

- A. KeyVault:mykeyvault;Secret:token
- B. App:Settings:Secret:mykeyvault:token
- C. AZUREKVCNNSTR_ <https://mykeyveault.vault.ezure.net/secrets/token/>
- D. @Microsoft.KeyVault(SecretUri=https://mykeyvault.vault.azure.net/secrets/token/)

Correct Answer: D

Add Key Vault secrets reference in the Function App configuration.

Syntax: @Microsoft.KeyVault(SecretUri={copied identifier for the username secret})

Reference:

<https://daniel-krzyczkowski.github.io/Integrate-Key-Vault-Secrets-With-Azure-Functions/>

Community vote distribution

D (94%)

6%

A company maintains multiple web and mobile applications. Each application uses custom in-house identity providers as well as social identity providers.

You need to implement single sign-on (SSO) for all the applications.

What should you do?

- A. Use Azure Active Directory B2C (Azure AD B2C) with custom policies.
- B. Use Azure Active Directory B2B (Azure AD B2B) and enable external collaboration.
- C. Use Azure Active Directory B2C (Azure AD B2C) with user flows.
- D. Use Azure Active Directory B2B (Azure AD B2B).

Correct Answer: B

You can add Google as an identity provider for B2B guest users.

Federation with SAML/WS-Fed identity providers for guest users.

Make sure your organization's external collaboration settings are configured such that you're allowed to invite guests.

Note 1: As a user who is assigned any of the limited administrator directory roles, you can use the Azure portal to invite B2B collaboration users. You can invite guest users to the directory, to a group, or to an application. After you invite a user through any of these methods, the invited user's account is added to Azure

Active Directory (Azure AD), with a user type of Guest.

Note 2: Direct federation in Azure Active Directory is now referred to as SAML/WS-Fed identity provider (IdP) federation.

Reference:

<https://docs.microsoft.com/en-us/azure/active-directory/external-identities/google-federation> <https://docs.microsoft.com/en-us/azure/active-directory/external-identities/add-users-administrator>

Community vote distribution

A (79%)

C (21%)

You develop a Python application for image rendering that uses GPU resources to optimize rendering processes. You deploy the application to an Azure Container Instances (ACI) Linux container.

The application requires a secret value to be passed when the container is started. The value must only be accessed from within the container. You need to pass the secret value.

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 environment variable Set the secureValue property to the secret value.
- B. Add the secret value to the container image. Use a managed identity.
- C. Add the secret value to the application code Set the container startup command.
- D. Add the secret value to an Azure Blob storage account. Generate a SAS token.
- E. Mount a secret volume containing the secret value in a secrets file.

Correct Answer: AE

A: Secure environment variables -

Another method (another than a secret volume) for providing sensitive information to containers (including Windows containers) is through the use of secure environment variables.

E: Use a secret volume to supply sensitive information to the containers in a container group. The secret volume stores your secrets in files within the volume, accessible by the containers in the container group. By storing secrets in a secret volume, you can avoid adding sensitive data like SSH keys or database credentials to your application code.

Reference:

<https://docs.microsoft.com/en-us/azure/container-instances/container-instances-volume-secret>

Community vote distribution

AE (100%)

You are developing a user portal for a company.

You need to create a report for the portal that lists information about employees who are subject matter experts for a specific topic. You must ensure that administrators have full control and consent over the data.

Which technology should you use?

- A. Microsoft Graph data connect
- B. Microsoft Graph API
- C. Microsoft Graph connectors

Correct Answer: A

Data Connect grants a more granular control and consent model: you can manage data, see who is accessing it, and request specific properties of an entity. This enhances the Microsoft Graph model, which grants or denies applications access to entire entities.

Microsoft Graph Data Connect augments Microsoft Graph's transactional model with an intelligent way to access rich data at scale. The data covers how workers communicate, collaborate, and manage their time across all the applications and services in Microsoft 365.

Incorrect:

Not B: The Microsoft Graph API is a RESTful web API that enables you to access Microsoft Cloud service resources. After you register your app and get authentication tokens for a user or service, you can make requests to the Microsoft Graph API.

A simplistic definition of a Graph API is an API that models the data in terms of nodes and edges (objects and relationships) and allows the client to interact with multiple nodes in a single request.

Not C: Microsoft Graph connectors, your organization can index third-party data so that it appears in Microsoft Search results.

With Microsoft Graph connectors, your organization can index third-party data so that it appears in Microsoft Search results.

Reference:

<https://docs.microsoft.com/en-us/graph/data-connect-concept-overview>

Community vote distribution

A (100%)

HOTSPOT -

You are a developer building a web site using a web app. The web site stores configuration data in Azure App Configuration.

Access to Azure App Configuration has been configured to use the identity of the web app for authentication. Security requirements specify that no other authentication systems must be used.

You need to load configuration data from Azure App Configuration.

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

```
Host .CreateDefaultBuilder(args)
    .ConfigureWebHostDefaults(wb =>
{
    wb.ConfigureAppConfiguration((hc, config) =>
    {
        var settings = config.Build();
        config.          (options =>
            AddAzureKeyVault
            DefaultAzureCredential
            ChainedTokenCredential
            ManagedIdentityCredential
            AddAzureAppConfiguration
        options.Connect(new Uri(settings["AppConfig:Endpoint"]),
            new          ());
            AddAzureKeyVault
            DefaultAzureCredential
            ChainedTokenCredential
            ManagedIdentityCredential
            AddAzureAppConfiguration
        });
    });
})
```

Answer Area

```
Host .CreateDefaultBuilder(args)
    .ConfigureWebHostDefaults(wb =>
{
    wb.ConfigureAppConfiguration((hc, config) =>
    {
        var settings = config.Build();
        config. (options =>
AddAzureKeyVault
DefaultAzureCredential
ChainedTokenCredential
ManagedIdentityCredential
AddAzureAppConfiguration

        options.Connect(new Uri(settings["AppConfig:Endpoint"]),
            new ());
AddAzureKeyVault
DefaultAzureCredential
ChainedTokenCredential
ManagedIdentityCredential
AddAzureAppConfiguration

    });
})
});
```

Box 1: AddAzureAppConfiguration -

Load data from App Configuration, code example:

```
public static IHostBuilder CreateHostBuilder(string[] args) =>
    Host.CreateDefaultBuilder(args)
        .ConfigureWebHostDefaults(webBuilder =>
            webBuilder.ConfigureAppConfiguration((hostingContext, config) =>
{
    var settings = config.Build();
    config.AddAzureAppConfiguration(options =>
{
    Etc.
```

Box 2: ManagedIdentityCredential

Use managed identities to access App Configuration

If you want to use a user-assigned managed identity, be sure to specify the clientId when creating the ManagedIdentityCredential.

```
config.AddAzureAppConfiguration(options =>
{
    options.Connect(new Uri(settings["AppConfig:Endpoint"]), new ManagedIdentityCredential("<your_clientId>"))
});
```

Full code sample:

```
public static IHostBuilder CreateHostBuilder(string[] args) =>
    Host.CreateDefaultBuilder(args)
        .ConfigureWebHostDefaults(webBuilder =>
            webBuilder.ConfigureAppConfiguration((hostingContext, config) =>
{
    var settings = config.Build();
    config.AddAzureAppConfiguration(options =>
        options.Connect(new Uri(settings["AppConfig:Endpoint"]), new ManagedIdentityCredential()));
});
```

.UseStartup<Startup>());

Reference:

<https://docs.microsoft.com/en-us/azure/azure-app-configuration/howto-integrate-azure-managed-service-identity?tabs=core5x&pivots=framework-dotnet>

HOTSPOT -

You are building an application that stores sensitive customer data in Azure Blob storage. The data must be encrypted with a key that is unique for each customer.

If the encryption key has been corrupted it must not be used for encryption.

You need to ensure that the blob is encrypted.

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

```
Uri blobUri = ... ; TokenCredential c = ...
byte[] key = ...; string verify = ...
var x = new AesManaged(key);
AsnEncodedData(key)
CustomerProvidedKey(key)
BlobContainerEncryptionScopeOptions { DefaultEncryptionScope = key }

if ( x.IV == verify
    x.RawData == verify
    x.Encryption == verify
    x.PreventEncryptionScopeOverride == verify ) {

    var o = new BlobClientOptions()
    {
        = x
        Version
        Transport
EncryptionScope
CustomerProvidedKey
    };
    var blobClient = new BlobClient(blobUri, c, o);
}
}
```

Answer Area

```
Uri blobUri = ... ; TokenCredential c = ...
byte[] key = ...; string verify = ...
var x = new AesManaged(key);
AsnEncodedData(key)
CustomerProvidedKey(key)
BlobContainerEncryptionScopeOptions { DefaultEncryptionScope = key }

if ( x.IV == verify
    x.RawData == verify
x.Encryption == verify
x.PreventEncryptionScopeOverride == verify ) {

    var o = new BlobClientOptions()
    {
        = x
Version
Transport
EncryptionScope
CustomerProvidedKey
    };
    var blobClient = new BlobClient(blobUri, c, o);
}
}
```

Correct Answer:

Box 1: CustomerProvidedKey(key)

The data must be encrypted with a key that is unique for each customer.

Sample code:

```
async static Task UploadBlobWithClientKey(Uri blobUri,
Stream data,
byte[] key,
string keySha256)
{
    // Create a new customer-provided key.
    // Key must be AES-256.
    var cpk = new CustomerProvidedKey(key);
```

Box 2: Encryption -

CustomerProvidedKey.EncryptionKey Property

Sample code continued:

```
// Check the key's encryption hash.
if (cpk.EncryptionKeyHash != keySha256)
{
    throw new InvalidOperationException("The encryption key is corrupted.");
}
```

Box 3: CustomerProvidedKey -

Sample code continued:

// Specify the customer-provided key on the options for the client.

```
BlobClientOptions options = new BlobClientOptions()
{
```

CustomerProvidedKey = cpk -

};

// Create the client object with options specified.

```
BlobClient blobClient = new BlobClient(
```

blobUri,

new DefaultAzureCredential(),

options);

Incorrect:

* Version - Gets the BlobClientOptions.ServiceVersion of the service API used when making requests.

Transport - The HttpPipelineTransport to be used for this client.

Reference:

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

HOTSPOT

You are developing a web application that uses the Microsoft Identity platform for user and resource authentication. The web application called several REST APIs.

You are implementing various authentication and authorization flows for the web application.

You need to validate the claims in the authentication token.

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

NOTE: Each correct selection is worth one point.

Answer Area

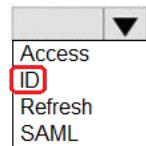
Requirement	Token type
Identify users for the application by using a JWT token that contains claims.	<input type="checkbox"/> Access <input type="checkbox"/> ID <input type="checkbox"/> Refresh <input checked="" type="checkbox"/> SAML
Identify the permissions granted to APIs by using a JWT token that contains claims.	<input type="checkbox"/> Access <input type="checkbox"/> ID <input type="checkbox"/> Refresh <input checked="" type="checkbox"/> SAML
Provide the web application with long-term access to resources on behalf of users without requiring interaction with those users.	<input type="checkbox"/> Access <input type="checkbox"/> ID <input checked="" type="checkbox"/> Refresh <input type="checkbox"/> SAML
Provide XML representations of claims that can be consumed by applications that use WS-Federation.	<input type="checkbox"/> Access <input type="checkbox"/> ID <input checked="" type="checkbox"/> Refresh <input type="checkbox"/> SAML

Answer Area

Requirement

Identify users for the application by using a JWT token that contains claims.

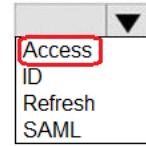
Token type



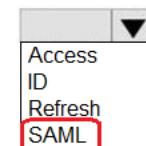
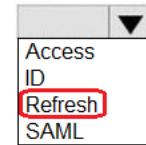
Identify the permissions granted to APIs by using a JWT token that contains claims.

Correct Answer:

Provide the web application with long-term access to resources on behalf of users without requiring interaction with those users.



Provide XML representations of claims that can be consumed by applications that use WS-Federation.



HOTSPOT

You are developing a content management application for technical manuals. The application is deployed as an Azure Static Web app.

Authenticated users can view pages under/manuals but only contributors can access the page /manuals/new.html.

You need to configure the routing for the web app.

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

NOTE: Each correct selection is worth one point.

Answer Area

```
"routes": [
{
  "route": "/",
  "allowedRoles": []
},
{
  "route": "/manuals*",
  "allowedRoles": []
},
{
  "route": "/contributors",
  "allowedRoles": []
},
{
  "route": "/authenticated",
  "allowedRoles": []
},
{
  "route": "/manuals/new.html",
  "allowedRoles": []
}]
```

Answer Area

```
"routes": [
{
  "route": "/",
  "allowedRoles": [
    "/manuals*",
    "contributors",
    "authenticated",
    "/manuals/new.html"
  ]
},
{
  "route": "/",
  "allowedRoles": [
    "/manuals*",
    "contributors",
    "authenticated",
    "/manuals/new.html"
  ]
}]
```

Correct Answer:

```
"routes": [
{
  "route": "/",
  "allowedRoles": [
    "/manuals*",
    "contributors",
    "authenticated",
    "/manuals/new.html"
  ]
},
{
  "route": "/",
  "allowedRoles": [
    "/manuals*",
    "contributors",
    "authenticated",
    "/manuals/new.html"
  ]
}]
```

Question #55

Topic 4

You are developing a web application that uses the Microsoft identity platform for user and resource authentication. The web application calls several REST APIs.

A REST API call must read the user's calendar. The web application requires permission to send an email as the user.

You need to authorize the web application and the API.

Which parameter should you use?

- A. tenant
- B. code_challenge
- C. state
- D. client_id
- E. scope

Correct Answer: E

Community vote distribution

E (100%)

HOTSPOT

You develop and deploy a web app to Azure App service. The web app allows users to authenticate by using social identity providers through the Azure B2C service. All user profile information is stored in Azure B2C.

You must update the web app to display common user properties from Azure B2C to include the following information:

- Email address
- Job title
- First name
- Last name
- Office location

You need to implement the user properties in the web app.

Which code library and API should you use? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Requirement	Value
API to access user properties	<input type="checkbox"/> Microsoft Graph <input type="checkbox"/> Azure AD Graph <input type="checkbox"/> Azure Key Vault <input type="checkbox"/> Azure AD entitlement management
Code library to interface to Azure AD B2C	<input type="checkbox"/> Microsoft Authentication Library (MSAL) <input type="checkbox"/> Microsoft Azure Key Vault SDK <input type="checkbox"/> Azure Identity library

Requirement	Value
API to access user properties	<input checked="" type="checkbox"/> Microsoft Graph <input type="checkbox"/> Azure AD Graph <input type="checkbox"/> Azure Key Vault <input type="checkbox"/> Azure AD entitlement management
Code library to interface to Azure AD B2C	<input checked="" type="checkbox"/> Microsoft Authentication Library (MSAL) <input type="checkbox"/> Microsoft Azure Key Vault SDK <input type="checkbox"/> Azure Identity library

HOTSPOT

You develop and deploy the following staticwebapp.config.json file to the app_location value specified in the workflow file of an Azure Static Web app:

```
{
  "routes": [
    {
      "route": "/api/*",
      "methods": ["GET"],
      "allowedRoles": ["registeredusers"]
    },
    {
      "route": "/api/*",
      "methods": ["PUT", "POST", "PATCH", "DELETE"],
      "allowedRoles": ["administrator"]
    },
    {
      "route": "/login",
      "rewrite": "./auth/login/github"
    },
    {
      "route": "./auth/login/twitter",
      "statusCode": 404
    },
    {
      "route": "/logout",
      "redirect": "./auth/logout"
    }
  ],
  "navigationFallback": {
    "rewrite": "index.html",
    "exclude": ["/images/*.{png,jpg,gif}", "/css/*"]
  },
  "responseOverrides": {
    "400": {
      "rewrite": "/invalid-invitation-error.html"
    },
    "401": {
      "redirect": "./auth/login/aad",
      "statusCode": 302
    },
    "403": {
      "rewrite": "/forbidden.html"
    },
    "404": {
      "rewrite": "/404.html"
    }
  },
  "mimeType": {
    ".json": "text/json"
  }
}
```

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

NOTE: Each correct selection is worth one point.

Answer Area		Yes	No
Statements			
Unauthenticated users are challenged to authenticate with GitHub.	<input type="radio"/>	<input type="radio"/>	
A non-existent file in the /images/ folder will generate a 404 response code.	<input type="radio"/>	<input type="radio"/>	
HTTP GET method requests from authenticated users in the role named registeredusers are sent to the API folder.	<input type="radio"/>	<input type="radio"/>	
Authenticated users that are not in the role named registeredusers and unauthenticated users are served a 401 HTTP error when accessing the API folder.	<input type="radio"/>	<input type="radio"/>	

Question #58	Topic 4
<p>You develop and deploy an Azure App Service web app named App1. You create a new Azure Key Vault named Vault1. You import several API keys, passwords, certificates, and cryptographic keys into Vault1.</p> <p>You need to grant App1 access to Vault1 and automatically rotate credentials. Credentials must not be stored in code.</p> <p>What should you do?</p> <p>A. Enable App Service authentication for App1. Assign a custom RBAC role to Vault1. B. Add a TLS/SSL binding to App1. C. Upload a self-signed client certificate to Vault1. Update App1 to use the client certificate. D. Assign a managed identity to App1.</p>	
<p>Correct Answer: D</p>	
<p><i>Community vote distribution</i></p>	
<p>D (100%)</p>	

You are developing a Java application to be deployed in Azure. The application stores sensitive data in Azure Cosmos DB.

You need to configure Always Encrypted to encrypt the sensitive data inside the application.

What should you do first?

- A. Create a new container to include an encryption policy with the JSON properties to be encrypted.
- B. Create a customer-managed key (CMK) and store the key in a new Azure Key Vault instance.
- C. Create a data encryption key (DEK) by using the Azure Cosmos DB SDK and store the key in Azure Cosmos DB.
- D. Create an Azure AD managed identity and assign the identity to a new Azure Key Vault instance.

Correct Answer: B

Community vote distribution

B (100%)

HOTSPOT

You develop a web app that interacts with Azure Active Directory (Azure AD) groups by using Microsoft Graph.

You build a web page that shows all Azure AD groups that are not of the type 'Unified'.

You need to build the Microsoft Graph query for the page.

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

NOTE: Each correct selection is worth one point.

Answer Area

<https://graph.microsoft.com/v1.0/groups?>

filter
search
contain

=
groupTypes/any(s:s ne 'Unified')
not groupTypes/contains('Unified')
not groupTypes/any(s:s eq 'Unified')
groupTypes/contains('Unified') eq false

&\$
\$top=true
\$count=true
\$filter=nested
\$consistencylevel=eventual

Answer Area

<https://graph.microsoft.com/v1.0/groups?>

filter
search
contain

=
groupTypes/any(s:s ne 'Unified')
not groupTypes/contains('Unified')
not groupTypes/any(s:s eq 'Unified')
groupTypes/contains('Unified') eq false

Correct Answer:

&\$
\$top=true
\$count=true
\$filter=nested
\$consistencylevel=eventual

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.

Code segments

DefaultAzureCredential
ClientSecretCredential
CloudClients
SecretClient

Answer area

```
def get_secret():

    var1 = os.environ.get("KEY_VAULT_URI")

    var2 = [ ] ( var1 , [ ] () )

    var3 = var2.get_secret("secret" )

    return "secret value {}" .format(var3.value)
```

Answer area**Correct Answer:**

```
def get_secret():

    var1 = os.environ.get("KEY_VAULT_URI")

    var2 = [ SecretClient ( var1 , [ DefaultAzureCredential ] () )

    var3 = var2.get_secret("secret" )

    return "secret value {}" .format(var3.value)
```

HOTSPOT

You are a developer building a web site using a web app. The web site stores configuration data in Azure App Configuration.

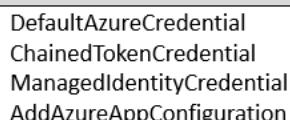
Access to Azure App Configuration has been configured to use the identity of the web app for authentication. Security requirements specify that no other authentication systems must be used.

You need to load configuration data from Azure App Configuration.

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

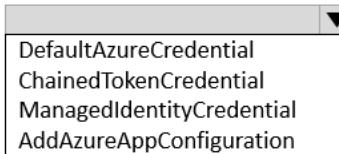
NOTE: Each correct selection is worth one point.

```
from azure.identity import
```



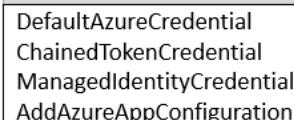
- DefaultAzureCredential
- ChainedTokenCredential
- ManagedIdentityCredential
- AddAzureAppConfiguration

```
from azure.app_configuration import
```



- DefaultAzureCredential
- ChainedTokenCredential
- ManagedIdentityCredential
- AddAzureAppConfiguration

```
credential =
```



- DefaultAzureCredential
- ChainedTokenCredential
- ManagedIdentityCredential
- AddAzureAppConfiguration

```
()
```

```
client =
```

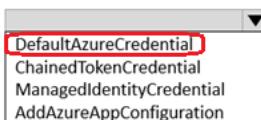


- DefaultAzureCredential
- ChainedTokenCredential
- ManagedIdentityCredential
- AddAzureAppConfiguration

```
(base_url="...", credential=credential)
```

Correct Answer:

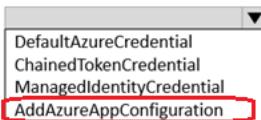
```
credential =
```



- DefaultAzureCredential
- ChainedTokenCredential
- ManagedIdentityCredential
- AddAzureAppConfiguration

```
()
```

```
client =
```



- DefaultAzureCredential
- ChainedTokenCredential
- ManagedIdentityCredential
- AddAzureAppConfiguration

```
(base_url="...", credential=credential)
```

You are developing several microservices to deploy to a new Azure Kubernetes Service cluster. The microservices manage data stored in Azure Cosmos DB and Azure Blob storage. The data is secured by using customer-managed keys stored in Azure Key Vault.

You must automate key rotation for all Azure Key Vault keys and allow for manual key rotation. Keys must rotate every three months. Notifications of expiring keys must be sent before key expiry.

You need to configure key rotation and enable key expiry notifications.

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

NOTE: Each correct selection is worth one point.

- A. Create and configure a new Azure Event Grid instance.
- B. Configure Azure Key Vault alerts.
- C. Create and assign an Azure Key Vault access policy.
- D. Create and configure a key rotation policy during key creation.

Correct Answer: AC

Question #1

Topic 5

DRAG DROP -

You develop a web app that uses the 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.

Correct Answer:

Actions**Answer Area**

Configure the web app to the Premium App Service tier.

Configure the web app to the Standard App Service tier.

Configure the web app to the Standard App Service tier.

Enable autoscaling on the web app.

Enable autoscaling on the web app.

Add a Scale rule.

Add a Scale rule.

Add a Scale rule.

Switch to an Azure App Services consumption plan.

Configure a Scale condition.

Configure a Scale condition.

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 #2

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.

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 #3

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.

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>

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.";  
}
```

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>

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>

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:

Features

- Users
- Funnels
- Impact
- Retention
- User Flows

Answer Area**Requirement**

Which pages visited by users most often correlate to a product purchase?

Feature

- Feature
- Feature
- Feature
- Feature

How does load time of the product display page affect a user's decision to purchase a product?

Which events most influence a user's decision to continue to use the application?

Are there places in the application that users often perform repetitive actions?

Correct Answer:**Features**

- Users
- Funnels
- Impact
- Retention
- User Flows

Answer Area**Requirement**

Which pages visited by users most often correlate to a product purchase?

Feature

- Users
- Impact
- Retention
- User Flows

How does load time of the product display page affect a user's decision to purchase a product?

Which events most influence a user's decision to continue to use the application?

Are there places in the application that users often perform repetitive actions?

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 #7

Topic 5

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>

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 presents 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>

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>

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>

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

Correct Answer:

```
let startTimestamp = ago(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
```

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>

HOTSPOT -

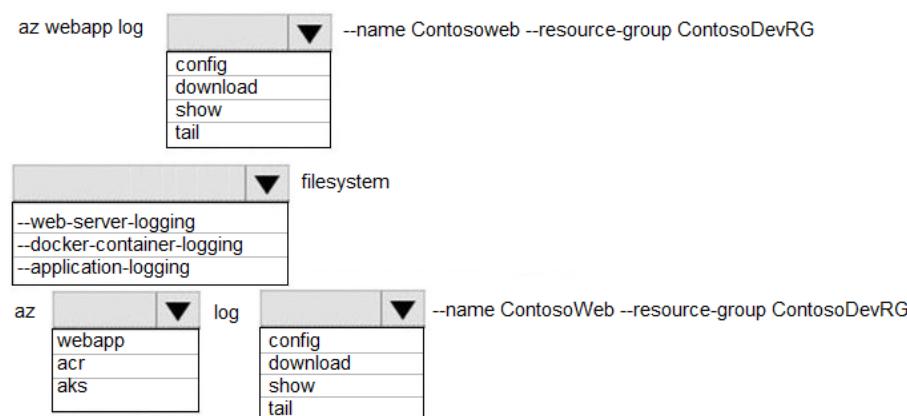
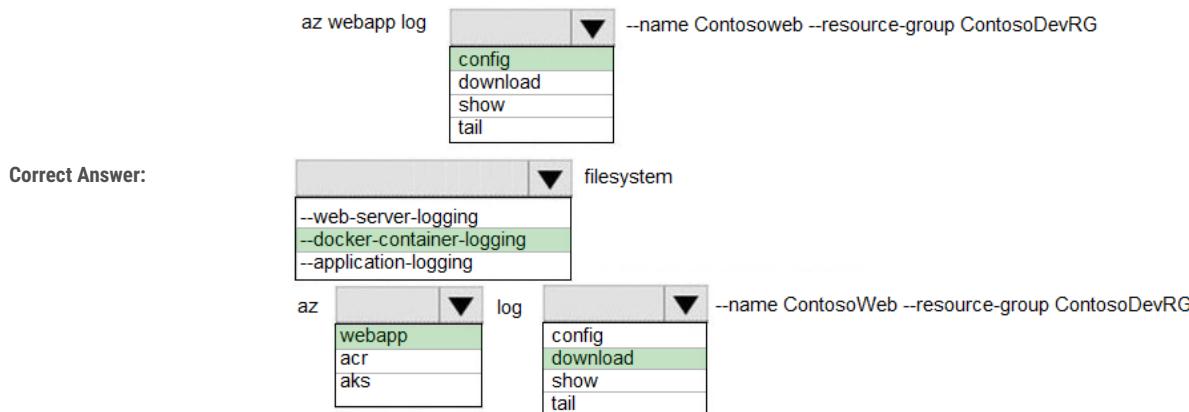
You plan to deploy a web app to App Service on Linux. You create an App Service plan. You create and push a custom Docker image that contains the web app to Azure Container Registry.

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**

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>

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>

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> <inbound> <base /> <cache-lookup caching-type="
Public	<input style="width: 100px; height: 20px; border: 1px solid black; margin-right: 10px;" type="text"/> Target
Private	" downstream-caching-type = "
Internal	<input style="width: 100px; height: 20px; border: 1px solid black; margin-right: 10px;" type="text"/> Target
External	</cache-lookup> </inbound>
Authorization	</policies>

Correct Answer:

Targets	Answer Area
Expect	<policies> <inbound> <base /> <cache-lookup caching-type="
Public	<input style="width: 100px; height: 20px; border: 1px solid black; margin-right: 10px;" type="text"/> Internal
Private	" downstream-caching-type = "
Internal	<input style="width: 100px; height: 20px; border: 1px solid black; margin-right: 10px;" type="text"/> Private
External	</cache-lookup> </inbound>
Authorization	</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 #15

Topic 5

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>

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>

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	Answer Area
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.	

(Left) (Right) (Up) (Down)

Correct Answer:

Actions	Answer Area
	Register the application with the Microsoft identity platform.
	Build a client by using the client app ID.
(Left)	Create an authentication provider.
(Right)	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/sdk/create-client>

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.
- (Left arrow) Add a diagnostic setting to the Azure Logic App. (Right arrow)

**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>

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.

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>

Question #20

Topic 5

You develop and deploy an Azure App Service web app. The app is deployed to multiple regions and uses Azure Traffic Manager. Application Insights is enabled for the app.

You need to analyze app uptime for each month.

Which two solutions will achieve the goal? Each correct answer presents a complete solution.

NOTE: Each correct selection is worth one point.

- A. Azure Monitor logs
- B. Application Insights alerts
- C. Azure Monitor metrics
- D. Application Insights web tests

Correct Answer: BD

Reference:

<https://azure.microsoft.com/en-us/blog/creating-a-web-test-alert-programmatically-with-application-insights/>

DRAG DROP -

You develop and deploy an Azure App Service web app. The web app accesses data in an Azure SQL database.

You must update the web app to store frequently used data in a new Azure Cache for Redis Premium instance.

You need to implement the Azure Cache for Redis features.

Which feature should you implement? To answer, drag the appropriate feature 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:

Answer Area

Features	Requirement	Feature
horizontal partitioning	Create a data structure for storing collections of related items	Feature
channel	Create a data structure for the most recently accessed cache items	Feature
list	Send messages through a high-performance publisher/subscriber mechanism	Feature
set		

Correct Answer:**Answer Area**

Features	Requirement	Feature
horizontal partitioning	Create a data structure for storing collections of related items	set
channel	Create a data structure for the most recently accessed cache items	list
list	Send messages through a high-performance publisher/subscriber mechanism	channel
set		

Reference:

<https://www.red-gate.com/simple-talk/development/dotnet-development/overview-of-azure-cache-for-redis/> <https://docs.microsoft.com/en-us/azure/architecture/best-practices/caching>

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
```

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>

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;"><p>Bypass cache</p><p>Override</p><p>Set if missing</p></div>
Cache expiration duration	<div style="border: 1px solid black; padding: 5px;"><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;"><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	<table border="1"><tr><td>Bypass cache</td></tr><tr><td>Override</td></tr><tr><td>Set if missing</td></tr></table>	Bypass cache	Override	Set if missing	
Bypass cache					
Override					
Set if missing					
Correct Answer: Cache expiration duration	<table border="1"><tr><td>1 second</td></tr><tr><td>1 minute</td></tr><tr><td>1 hour</td></tr><tr><td>1 day</td></tr></table>	1 second	1 minute	1 hour	1 day
1 second					
1 minute					
1 hour					
1 day					
Query string caching behavior	<table border="1"><tr><td>Ignore query strings</td></tr><tr><td>Bypass caching for query strings</td></tr><tr><td>Cache every unique URL</td></tr></table>	Ignore query strings	Bypass caching for query strings	Cache every unique URL	
Ignore query strings					
Bypass caching for query strings					
Cache every unique URL					

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>

HOTSPOT -

You are developing an ASP.NET Core time sheet application that runs as an Azure Web App. Users of the application enter their time sheet information on the first day of every month.

The application uses a third-party web service to validate data.

The application encounters periodic server errors due to errors that result from calling a third-party web server. Each request to the third-party server has the same chance of failure.

You need to configure an Azure Monitor alert to detect server errors unrelated to the third-party service. You must minimize false-positive alerts.

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:

```
"type": "Microsoft.Insights/metricAlerts",
"properties": {
  "criteria": {
    "odata.type": "...",
    "allOf": [
      {
        "criterionType": "DynamicThresholdCriterion",
        "singleResourceMultipleMetricCriteria": [
          "Http4xx",
          "Http5xx"
        ]
      },
      {
        "alertSensitivity": "Low"
      }
    ]
  }
}
```

```
"type": "Microsoft.Insights/metricAlerts",
"properties": {
  "criteria": {
    "odata.type": "...",
    "allOf": [
      {
        "criterionType": "DynamicThresholdCriterion",
        "singleResourceMultipleMetricCriteria": [
          "Http4xx",
          "Http5xx"
        ]
      },
      {
        "alertSensitivity": "Low"
      }
    ]
  }
}
```

Correct Answer:

Box 1: DynamicThresholdCriterion

Box 2: Http5xx -

Server errors are in the 5xx range.

Client errors are in the 4xx range

Box 3: Low -

Reference:

<https://docs.microsoft.com/en-us/azure/azure-monitor/alerts/alerts-dynamic-thresholds>

You are developing a web application that uses Azure Cache for Redis. You anticipate that the cache will frequently fill and that you will need to evict keys.

You must configure Azure Cache for Redis based on the following predicted usage pattern: A small subset of elements will be accessed much more often than the rest.

You need to configure the Azure Cache for Redis to optimize performance for the predicted usage pattern.

Which two eviction policies will achieve the goal?

NOTE: Each correct selection is worth one point.

- A. noeviction
- B. allkeys-lru
- C. volatile-lru
- D. allkeys-random
- E. volatile-ttl
- F. volatile-random

Correct Answer: BC

B: The allkeys-lru policy evict keys by trying to remove the less recently used (LRU) keys first, in order to make space for the new data added. Use the allkeys-lru policy when you expect a power-law distribution in the popularity of your requests, that is, you expect that a subset of elements will be accessed far more often than the rest.

C: volatile-lru: evict keys by trying to remove the less recently used (LRU) keys first, but only among keys that have an expire set, in order to make space for the new data added.

Note: The allkeys-lru policy is more memory efficient since there is no need to set an expire for the key to be evicted under memory pressure.

Reference:

<https://redis.io/topics/lru-cache>

DRAG DROP -

An organization has web apps hosted in Azure.

The organization wants to track events and telemetry data in the web apps by using Application Insights.

You need to configure the web apps for Application Insights.

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

Select and Place:

Actions	Answer Area
Configure the Azure App Service SDK for the app	
Configure the Application Insights SDK in the app	
Copy the connection string	
Create an Azure Machine Learning workspace	
Create an Application Insights resource	

Correct Answer:

Actions	Answer Area
Configure the Azure App Service SDK for the app	
Create an Azure Machine Learning workspace	

Step 1: Create an Application Insights resource

Creating an Application Insights workspace-based resource is a prerequisite.

Step 2: Copy the connection string

A connection string identifies the resource that you want to associate with your telemetry data. It also allows you to modify the endpoints that your resource will use as a destination for your telemetry. You'll need to copy the connection string and add it to your application's code or to an environment variable.

Step 3: Configure the Application Insights SDK in the app

The Application Insights SDK for ASP.NET Core can monitor your applications no matter where or how they run.

Install the Application Insights SDK NuGet package for ASP.NET Core.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-monitor/app/asp-net-core>

An organization hosts web apps in Azure. The organization uses Azure Monitor.

You discover that configuration changes were made to some of the web apps.

You need to identify the configuration changes.

Which Azure Monitor log should you review?

- A. AppServiceAppLogs
- B. AppServiceEnvironmentPlatformLogs
- C. AppServiceConsoleLogs
- D. AppServiceAuditLogs

Correct Answer: B

The log type AppServiceEnvironmentPlatformLogs handles the App Service Environment: scaling, configuration changes, and status logs.

Incorrect:

AppServiceAppLogs contains logs generated through your application.

AppServiceAuditLogs logs generated when publishing users successfully log on via one of the App Service publishing protocols.

Reference:

<https://docs.microsoft.com/en-us/azure/app-service/troubleshoot-diagnostic-logs>

You develop and deploy an Azure App Service web app to a production environment. You enable the Always On setting and the Application Insights site extensions.

You deploy a code update and receive multiple failed requests and exceptions in the web app.

You need to validate the performance and failure counts of the web app in near real time.

Which Application Insights tool should you use?

- A. Profiler
- B. Smart Detection
- C. Live Metrics Stream
- D. Application Map
- E. Snapshot Debugger

Correct Answer: C

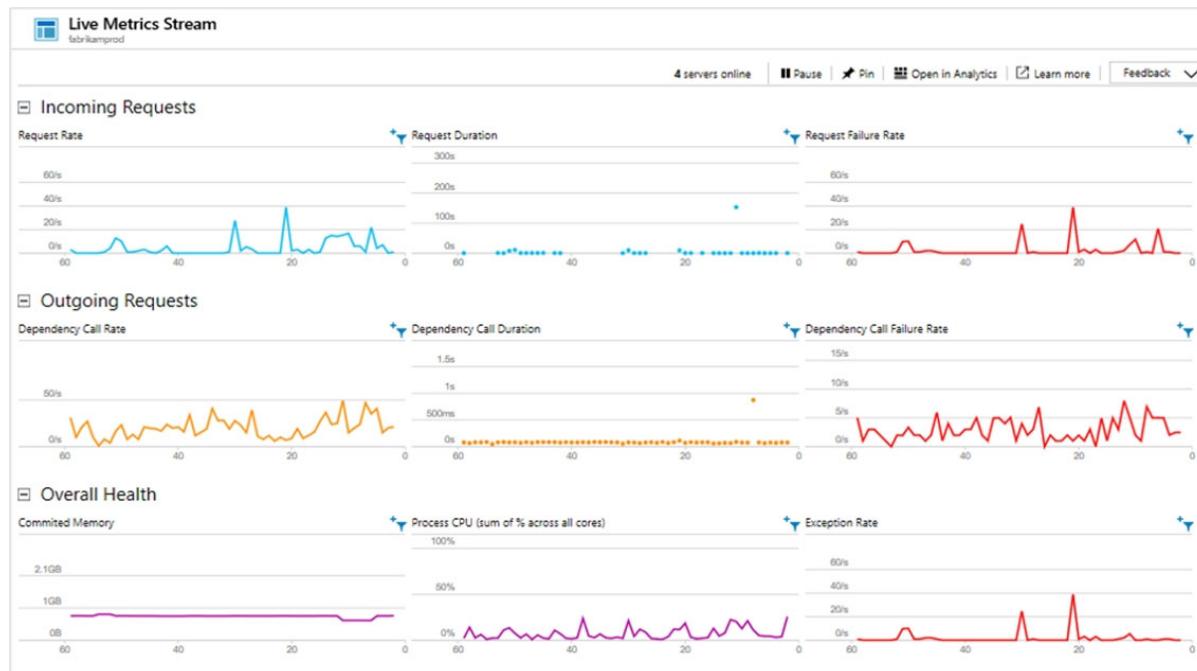
Live Metrics Stream -

Deploying the latest build can be an anxious experience. If there are any problems, you want to know about them right away, so that you can back out if necessary. Live Metrics Stream gives you key metrics with a latency of about one second.

With Live Metrics Stream, you can:

* Validate a fix while it's released, by watching performance and failure counts.

* Etc.



Incorrect:

* Profiler

Azure Application Insights Profiler provides performance traces for applications running in production in Azure. Profiler:

Captures the data automatically at scale without negatively affecting your users.

Helps you identify the hot code path spending the most time handling a particular web request.

* Snapshot debugger

When an exception occurs, you can automatically collect a debug snapshot from your live web application. The snapshot shows the state of source code and variables at the moment the exception was thrown. The Snapshot Debugger in Azure Application Insights monitors exception telemetry from your web app. It collects snapshots on your top-throwing exceptions so that you have the information you need to diagnose issues in production.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-monitor/app/live-stream>

HOTSPOT -

You deploy an ASP.NET web app to Azure App Service.

You must monitor the web app by using Application Insights.

You need to configure Application Insights to meet the requirements.

Which feature 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**Requirement**

Automatically warn you of potential performance problems and failure anomalies in the web app.

Feature

Smart Detection
Snapshot Debugger
Profiler
Multi-step test

Automatically collect the state of the source code and variables when an exception is thrown in the web app.

Smart Detection
Snapshot Debugger
Profiler
Multi-step test

Capture performance traces of the web app without negatively affecting users of the web app.

Smart Detection
Snapshot Debugger
Profiler
Multi-step test

Correct Answer:

Answer Area**Requirement**

Automatically warn you of potential performance problems and failure anomalies in the web app.

Feature

Smart Detection
Snapshot Debugger
Profiler
Multi-step test

Automatically collect the state of the source code and variables when an exception is thrown in the web app.

Smart Detection
Snapshot Debugger
Profiler
Multi-step test

Capture performance traces of the web app without negatively affecting users of the web app.

Smart Detection
Snapshot Debugger
Profiler
Multi-step test

Box 1: Smart Detection -

Smart detection automatically warns you of potential performance problems and failure anomalies in your web application. It performs proactive analysis of the telemetry that your app sends to Application Insights. If there is a sudden rise in failure rates, or abnormal patterns in client or server performance, you get an alert. This feature needs no configuration. It operates if your application sends enough telemetry.

Box 2: Snapshot Debugger -

When an exception occurs, you can automatically collect a debug snapshot from your live web application. The snapshot shows the state of source code and variables at the moment the exception was thrown. The Snapshot Debugger in Azure Application Insights monitors exception telemetry from your web app. It collects snapshots on your top-throwing exceptions so that you have the information you need to diagnose issues in production.

Box 3: Profiler -

Azure Application Insights Profiler provides performance traces for applications running in production in Azure. Profiler:

Captures the data automatically at scale without negatively affecting your users.

Helps you identify the **hot** code path spending the most time handling a particular web request.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-monitor/app/proactive-diagnostics> <https://docs.microsoft.com/en-us/azure/azure-monitor/snapshot-debugger/snapshot-debugger> <https://docs.microsoft.com/en-us/azure/azure-monitor/profiler/profiler-overview>

Question #31

Topic 5

You are building a web application that performs image analysis on user photos and returns metadata containing objects identified. The image analysis is very costly in terms of time and compute resources. You are planning to use Azure Redis Cache so duplicate uploads do not need to be reprocessed.

In case of an Azure data center outage, metadata loss must be kept to a minimum.

You need to configure the Azure Redis cache instance.

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

NOTE: Each correct selection is worth one point.

- A. Configure Azure Redis with AOF persistence.
- B. Configure Azure Redis with RDB persistence.
- C. Configure second storage account for persistence.
- D. Set backup frequency to the minimum value.

Correct Answer: BD

RDB persistence - When you use RDB persistence, Azure Cache for Redis persists a snapshot of your cache in a binary format. The snapshot is saved in an

Azure Storage account. The configurable backup frequency determines how often to persist the snapshot. If a catastrophic event occurs that disables both the primary and replica cache, the cache is reconstructed using the most recent snapshot.

Note: Azure Cache for Redis supports zone redundant configurations in the Premium and Enterprise tiers. A zone redundant cache can place its nodes across different Azure Availability Zones in the same region. It eliminates data center or AZ outage as a single point of failure and increases the overall availability of your cache.

Incorrect:

Not A: Zone redundancy doesn't support AOF persistence or work with geo-replication currently.

Not C: No need for a second storage account.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-cache-for-redis/cache-how-to-premium-persistence>

You are developing an Azure-based web application. The application goes offline periodically to perform offline data processing. While the application is offline, numerous Azure Monitor alerts fire which result in the on-call developer being paged. The application must always log when the application is offline for any reason. You need to ensure that the on-call developer is not paged during offline processing. What should you do?

- A. Add Azure Monitor alert processing rules to suppress notifications.
- B. Disable Azure Monitor Service Health Alerts during offline processing.
- C. Create an Azure Monitor Metric Alert.
- D. Build an Azure Monitor action group that suppresses the alerts.

Correct Answer: D

You can use alert processing rules to add action groups or remove (suppress) action groups from your fired alerts.

Reference:

<https://docs.microsoft.com/en-us/azure/azure-monitor/alerts/alerts-action-rules>

You are developing an online game that includes a feature that allows players to interact with other players on the same team within a certain distance. The calculation to determine the players in range occurs when players move and are cached in an Azure Cache for Redis instance.

The system should prioritize players based on how recently they have moved and should not prioritize players who have logged out of the game.

You need to select an eviction policy.

Which eviction policy should you use?

- A. allkeys-lru
- B. volatile-lru
- C. allkeys-ifu
- D. volatile-ttl

Correct Answer: A

You develop an Azure App Service web app and deploy to a production environment. You enable Application Insights for the web app.

The web app is throwing multiple exceptions in the environment.

You need to examine the state of the source code and variables when the exceptions are thrown.

Which Application Insights feature should you configure?

- A. Smart detection
- B. Profiler
- C. Snapshot Debugger
- D. Standard test

Correct Answer: C

DRAG DROP

You develop and deploy a Java application to Azure. The application has been instrumented by using the Application Insights SDK.

The telemetry data must be enriched and processed before it is sent to the Application Insights service.

You need to modify the telemetry data.

Which Application Insights SDK 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.

Features
Sampling
Telemetry initializer
Telemetry processor
Telemetry channel

Answer Area	
Requirement	Feature
Reduce the volume of telemetry without affecting statistics.	<input type="text"/>
Enrich telemetry with additional properties or override an existing one.	<input type="text"/>
Completely replace or discard a telemetry item.	<input type="text"/>

Correct Answer:

Answer Area

Requirement

Reduce the volume of telemetry without affecting statistics.

Feature

Sampling

Enrich telemetry with additional properties or override an existing one.

Telemetry initializer

Completely replace or discard a telemetry item.

Telemetry processor

HOTSPOT

You develop new functionality in a web application for a company that provides access to seismic data from around the world. The seismic data is stored in Redis Streams within an Azure Cache for Redis instance.

The new functionality includes a real-time display of seismic events as they occur.

You need to implement the Azure Cache for Redis command to receive seismic data.

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

NOTE: Each correct selection is worth one point.

Answer Area

XLEN
XREAD
XRANGE

BLOCK 0
COUNT 0
BLOCK -1
COUNT -1

STREAMS seismicData

\$
(&
(0-0 +
(0-0 -

Answer Area

Correct Answer:

XLEN
XREAD
XRANGE

BLOCK 0
COUNT 0
BLOCK -1
COUNT -1

STREAMS seismicData

\$
(&
(0-0 +
(0-0 -

You develop an ASP.NET Core app that uses Azure App Configuration. You also create an App Configuration containing 100 settings.

The app must meet the following requirements:

- Ensure the consistency of all configuration data when changes to individual settings occur.
- Handle configuration data changes dynamically without causing the application to restart.
- Reduce the overall number of requests made to App Configuration APIs.

You must implement dynamic configuration updates in the app.

What are two ways to achieve this goal? Each correct answer presents part of the solution.

NOTE: Each correct selection is worth one point.

- A. Create and register a sentinel key in the App Configuration store. Set the refreshAll parameter of the Register method to true.
- B. Increase the App Configuration cache expiration from the default value.
- C. Decrease the App Configuration cache expiration from the default value.
- D. Create and configure Azure Key Vault. Implement the Azure Key Vault configuration provider.
- E. Register all keys in the App Configuration store. Set the refreshAll parameter of the Register method to false.
- F. Create and implement environment variables for each App Configuration store setting.

Correct Answer: AB

HOTSPOT

You develop and deploy an Azure App Service web app that connects to Azure Cache for Redis as a content cache. All resources have been deployed to the East US 2 region.

The security team requires the following audit information from Azure Cache for Redis:

- The number of Redis client connections from an associated IP address.
- Redis operations completed on the content cache.
- The location (region) in which the Azure Cach3e for Redis instance was accessed.

The audit information must be captured and analyzed by a security team application deployed to the Central US region.

You need to log information on all client connections to the cache.

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

NOTE: Each correct selection is worth one point.

Answer Area

Requirement	Configuration value
Store log information.	<input type="checkbox"/> Log Analytics workspace <input type="checkbox"/> Blob Storage account <input type="checkbox"/> Data Lake Storage Gen2 Storage account <input type="checkbox"/> Event hub
Enable client connection logging.	<input type="checkbox"/> Diagnostic setting <input type="checkbox"/> Managed identity <input type="checkbox"/> App registration <input type="checkbox"/> Environment variable

Answer Area**Requirement**

Store log information.

Configuration value

- Log Analytics workspace
- Blob Storage account
- Data Lake Storage Gen2 Storage account
- Event hub

Correct Answer:

Enable client connection logging.

- Diagnostic setting
- Managed identity
- App registration
- Environment variable

Question #39

Topic 5

You develop and deploy a web app to Azure App Service. The Azure App Service uses a Basic plan in a single region.

Users report that the web app is responding slow. You must capture the complete call stack to help identify performance issues in the code. Call stack data must be correlated across app instances. You must minimize cost and impact to users on the web app.

You need to capture the telemetry.

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

NOTE: Each correct selection is worth one point.

- A. Restart all apps in the App Service plan.
- B. Enable Application Insights site extensions.
- C. Upgrade the Azure App Service plan to Premium.
- D. Enable Profiler.
- E. Enable the Always On setting for the app service.
- F. Enable Snapshot debugger.
- G. Enable remote debugging.

Correct Answer: DEF

You are building an application to track cell towers that are available to phones in near real time. A phone will send information to the application by using the Azure Web PubSub service. The data will be processed by using an Azure Functions app. Traffic will be transmitted by using a content delivery network (CDN).

The Azure function must be protected against misconfigured or unauthorized invocations.

You need to ensure that the CDN allows for the Azure function protection.

Which HTTP header should be on the allowed list?

- A. Authorization
- B. WebHook-Request-Callback
- C. Resource
- D. WebHook-Request-Origin

Correct Answer: D

Topic 6 - Question Set 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 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>

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 order processing and financial transactions.

Note: An event is a lightweight notification of a condition or a state change. Event hubs is usually used reacting to status changes.

Reference:

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

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	Enterprise Integration Pack
Code View Editor	Edit definitions in JSON	Code View Editor
Enterprise Integration Pack	Visually add functionality	Logic Apps Designer

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-enterprise-integration-overview](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>

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>

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}", peekedMessage.AsString);
}
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 #6

Topic 6

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>

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>

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>

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

.
.
.

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.
        SendWindowsNativeNotificationAsync
        SubmitNotificationHubJobAsync
        ScheduleNotificationAsync
        SendAppleNativeNotificationAsync
        ...
}
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>

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>

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
HTTP(s) endpoint	Target
Basic	Gateway credentials
Client cert	

Correct Answer:

Values	Answer Area
	Configuration parameter
HTTP(s) endpoint	Target
Basic	Gateway credentials

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>

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;"> SAS tokens Key authentication Management Access Control </div>
Topic publishing	<div style="border: 1px solid black; padding: 5px; width: fit-content;"> ValidationCode handshake ValidationURL handshake JWT token </div>

Answer Area

Option	Value
WebHook event delivery	<div style="border: 1px solid black; padding: 5px; width: fit-content;"> SAS tokens Key authentication Management Access Control </div>
Topic publishing	<div style="border: 1px solid black; padding: 5px; width: fit-content;"> ValidationCode handshake ValidationURL handshake 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>

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>

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>

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/>

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>

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>

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
      " conflict = " ) ) " >
    <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
      " conflict = " ) ) " >
    <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>

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	Answer Area
raw	<code>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);</code>
windows	
windowsphone	
application/xml	
application/json	
application/octet-stream	

Correct Answer:

Code segments	Answer Area
raw	<code>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);</code>
windowsphone	
application/xml	
application/json	

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(HttpMethod.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 #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 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>

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**

Event Source

Technology

Technology

Event Receiver

Technology

Event Handler

Technology

Correct Answer:**Technologies**

Azure Event Hub

Azure Blob Storage

Azure App Service

Answer Area**Object**

Event Source

Technology

Azure Event Grid

Event Receiver

Azure Logic App

Event Handler

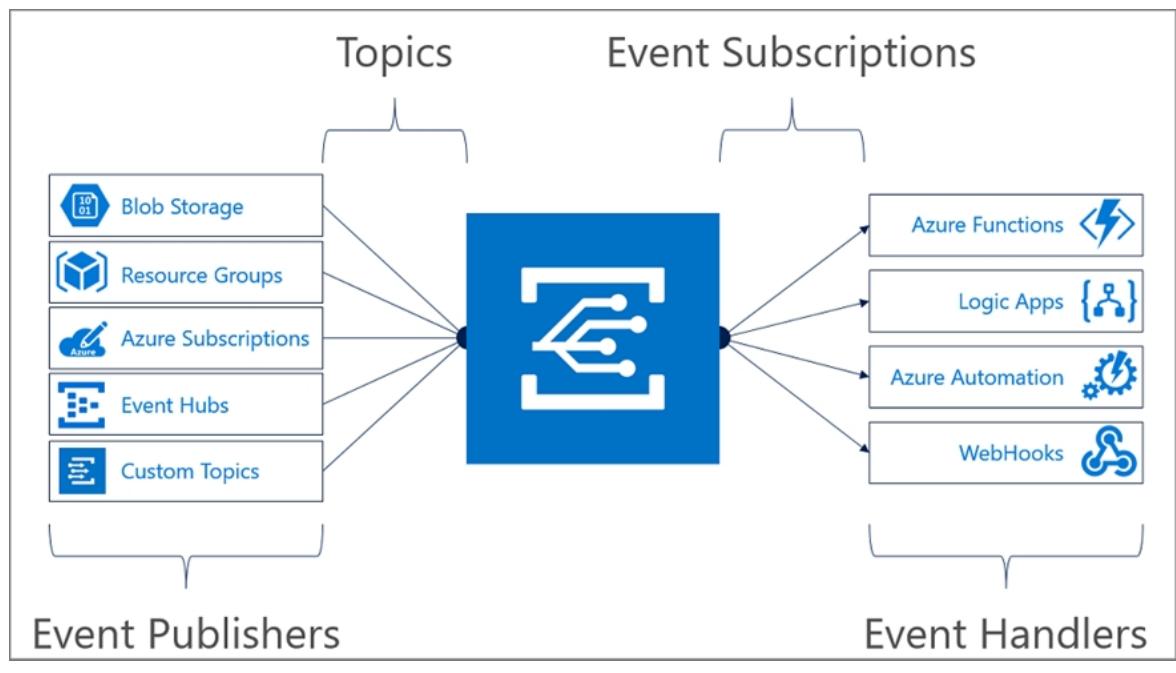
Azure Service Bus

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, EventHubs, Storage Queue, WebHook endpoint, or many other supported Azure Services.

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 #22

Topic 6

You are creating an app that will use CosmosDB for data storage. The app will process batches of relational data.

You need to select an API for the app.

Which API should you use?

- A. MongoDB API
- B. Table API
- C. SQL API
- D. Cassandra API

Correct Answer: Incorrect Answer:

For relational data you will need the SQL API

A: The MongoDB API is not used for relational data.

B: The Table API only supports data in the key/value format

D: The Cassandra API only supports OLTP (Online Transactional Processing) and not batch processing.

Reference:

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

HOTSPOT -

You are developing a .NET application that communicates with Azure Storage.

A message must be stored when the application initializes.

You need to implement the message.

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

```
CloudStorageAccount storageAccount = CloudStorageAccount.Parse  
(CloudConfigurationManager.GetSetting ("StorageConnectionString"));
```

pVar1 = storageAccount.	();
CloudQueueClient	CreateCloudQueueClient
CloudTableClient	CreateCloudTableClient
CloudQueue	GetQueueReference
CloudTable	GetTableReference

```
pVar2 = pVar1.
```

("contoso-storage");
CreateCloudQueueClient
CreateCloudTableClient
GetQueueReference
GetTableReference

```
try  
{  
    await pVar2.CreateIfNotExistsAsync();  
}  
catch (StorageException x)  
{  
    throw;  
}  
CloudQueueMessage cloudQueueMessage = new CloudQueueMessage("App Launch: <iUserID>");  
await pVar2.AddMessageAsync(cloudQueueMessage);
```

Correct Answer:

Answer Area

```
CloudStorageAccount storageAccount = CloudStorageAccount.Parse
(CloudConfigurationManager.GetSetting ("StorageConnectionString"));

pVar1 = storageAccount. ▼ ( );
CloudQueueClient
CloudTableClient
CloudQueue
CloudTable

pVar2 = pVar1. ▼ ( "contoso-storage");
CloudQueueClient
CloudTableClient
CloudQueue
CloudTable

CreateCloudQueueClient
CreateCloudTableClient
GetQueueReference
GetTableReference

try
{
    await pVar2.CreateIfNotExistsAsync ();
}
catch (StorageException x)
{
    throw;
}
CloudQueueMessage cloudQueueMessage = new CloudQueueMessage("App Launch: <iUserID>");
await pVar2.AddMessageAsync(cloudQueueMessage);
```

Reference:

<https://docs.microsoft.com/en-us/azure/storage/queues/storage-dotnet-how-to-use-queues?tabs=dotnetv11>

HOTSPOT -

A software as a service (SaaS) company provides document management services. The company has a service that consists of several Azure web apps. All

Azure web apps run in an Azure App Service Plan named PrimaryASP.

You are developing a new web service by using a web app named ExcelParser. The web app contains a third-party library for processing Microsoft Excel files.

The license for the third-party library stipulates that you can only run a single instance of the library.

You need to configure the service.

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

```
Set-AzAppServicePlan `  
    -ResourceGroupName $rg `  
    -Name "PrimaryASP" `
```

NumberOfSites 1	▼
PerSiteScaling \$true	▼
TargetWorkerCount = 1	▼
MaxNumberOfWorkers = 1	▼
SiteConfig.NumberOfWorkers = 1	▼

```
$app = Get-AzWebApp `  
    -ResourceGroupName $rg `  
    -Name "ExcelParser"
```

```
$app.

|                                |   |
|--------------------------------|---|
| NumberOfSites 1                | ▼ |
| PerSiteScaling \$true          | ▼ |
| TargetWorkerCount = 1          | ▼ |
| MaxNumberOfWorkers = 1         | ▼ |
| SiteConfig.NumberOfWorkers = 1 | ▼ |


```

```
Set-AzWebApp $app
```

Answer Area

```
Set-AzAppServicePlan ` 
    -ResourceGroupName $rg ` 
    -Name "PrimaryASP" `
```

▼
NumberOfSites 1
PerSiteScaling \$true
TargetWorkerCount = 1
MaxNumberOfWorkers = 1
SiteConfig.NumberOfWorkers = 1

Correct Answer:

```
$app = Get-AzWebApp ` 
    -ResourceGroupName $rg ` 
    -Name "ExcelParser"
```

\$app.▼
NumberOfSites 1
PerSiteScaling \$true
TargetWorkerCount = 1
MaxNumberOfWorkers = 1
SiteConfig.NumberOfWorkers = 1

```
Set-AzWebApp $app
```

Reference:

<https://docs.microsoft.com/en-us/azure/app-service/manage-scale-per-app>

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 requirements. 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	Answer Area	Policy type
Inbound	Requirement Support alternative input parameters.	policy type
Outbound	Requirement Remove formatting text from responses.	policy type
Backend	Requirement Provide additional context to back-end services.	policy type

Correct Answer:

Policy types	Answer Area	Policy type
Inbound	Requirement Support alternative input parameters.	Inbound
Outbound	Requirement Remove formatting text from responses.	Outbound
Backend	Requirement Provide additional context to back-end services.	Inbound

Reference:

<https://docs.microsoft.com/en-us/azure/api-management/api-management-howto-policies> <https://docs.microsoft.com/en-us/azure/api-management/api-management-transformation-policies#forward-context-information-to-the-backend-service>

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:

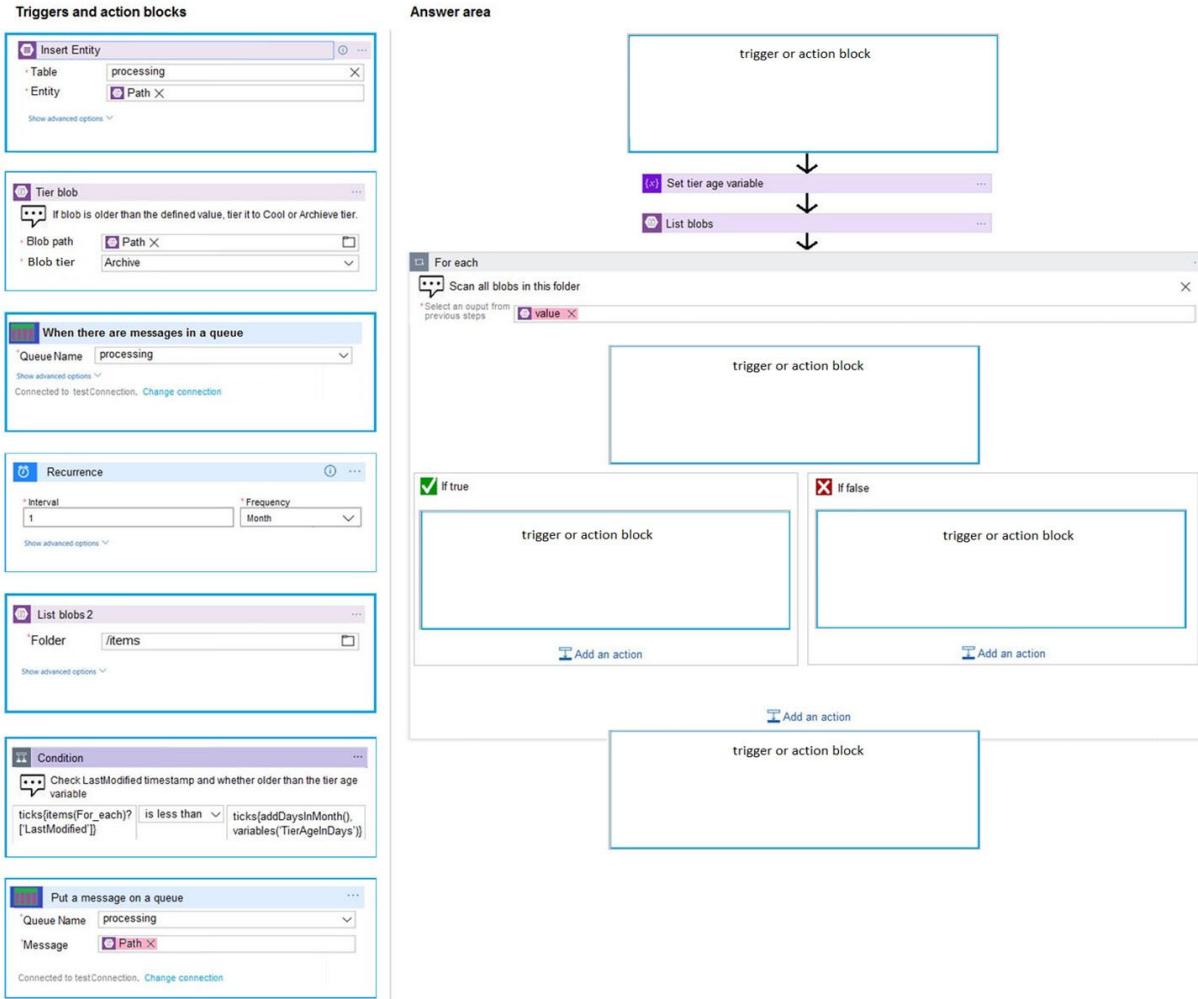
<https://docs.microsoft.com/en-us/azure/service-bus-messaging/service-bus-azure-and-service-bus-queues-compared-contrasted>

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

Box 1: Recurrence..

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 #28

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 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 #29

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 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>

You are building a loyalty program for a major snack producer. When customers buy a snack at any of 100 participating retailers the event is recorded in Azure.

Event Hub. Each retailer is given a unique identifier that is used as the primary identifier for the loyalty program.

Retailers must be able to be added or removed at any time. Retailers must only be able to record sales for themselves.

You need to ensure that retailers can record sales.

What should you do?

- A. Use publisher policies for retailers.
- B. Create a partition for each retailer.
- C. Define a namespace for each retailer.

Correct Answer: A

Event Hubs enables granular control over event publishers through publisher policies. Publisher policies are run-time features designed to facilitate large numbers of independent event publishers. With publisher policies, each publisher uses its own unique identifier when publishing events to an event hub.

Incorrect:

Not C: An Event Hubs namespace is a management container for event hubs (or topics, in Kafka parlance). It provides DNS-integrated network endpoints and a range of access control and network integration management features such as IP filtering, virtual network service endpoint, and Private Link.

Reference:

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

DRAG DROP -

You develop and deploy a web app to Azure App Service in a production environment. You scale out the web app to four instances and configure a staging slot to support changes.

You must monitor the web app in the environment to include the following requirements:

- ☞ Increase web app availability by re-routing requests away from instances with error status codes and automatically replace instances if they remain in an error state after one hour.

- ☞ Send web server logs, application logs, standard output, and standard error messaging to an Azure Storage blob account.

You need to configure Azure App Service.

Which values should you use? To answer, drag the appropriate configuration value to the correct requirements. Each configuration 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:

Configuration values

- Health check
- Diagnostic setting
- Deployment slot
- Autoscale rule
- Zone redundancy

Answer Area**Requirement**

Increase availability

Configuration value

Configuration value

Send logs

Configuration value

Correct Answer:

Configuration values

- Health check
- Diagnostic setting
- Deployment slot
- Autoscale rule
- Zone redundancy

Answer Area**Requirement**

Increase availability

Configuration value

Health check

Diagnostic setting

Box 1: Health check -

Health check increases your application's availability by re-routing requests away from unhealthy instances, and replacing instances if they remain unhealthy. Your

App Service plan should be scaled to two or more instances to fully utilize Health check.

Box 2: Diagnostic setting -

Azure provides built-in diagnostics to assist with debugging an App Service app.

With the new Azure Monitor integration, you can create Diagnostic Settings to send logs to Storage Accounts, Event Hubs and Log Analytics.

Reference:

<https://docs.microsoft.com/en-us/azure/app-service/monitor-instances-health-check> <https://docs.microsoft.com/en-us/azure/app-service/troubleshoot-diagnostic-logs>

You develop a solution that uses Azure Virtual Machines (VMs).

The VMs contain code that must access resources in an Azure resource group. You grant the VM access to the resource group in Resource Manager.

You need to obtain an access token that uses the VM's system-assigned managed identity.

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

- A. From the code on the VM, call Azure Resource Manager using an access token.
- B. Use PowerShell on a remote machine to make a request to the local managed identity for Azure resources endpoint.
- C. Use PowerShell on the VM to make a request to the local managed identity for Azure resources endpoint.
- D. From the code on the VM, call Azure Resource Manager using a SAS token.
- E. From the code on the VM, generate a user delegation SAS token.

Correct Answer: BD

You are developing a road tollway tracking application that sends tracking events by using Azure Event Hubs using premium tier.

Each road must have a throttling policy uniquely assigned.

You need to configure the event hub to allow for per-road throttling.

What should you do?

- A. Use a unique consumer group for each road.
- B. Ensure each road stores events in a different partition.
- C. Ensure each road has a unique connection string.
- D. Use a unique application group for each road.

Correct Answer: B

You develop and deploy an ASP.NET Core application that connects to an Azure Database for MySQL instance.

Connections to the database appear to drop intermittently and the application code does not handle the connection failure.

You need to handle the transient connection errors in code by implementing retries.

What are three possible ways to achieve this goal? Each correct answer presents part of the solution.

NOTE: Each correct selection is worth one point.

- A. Close the database connection and immediately report an error.
- B. Disable connection pooling and configure a second Azure Database for MySQL instance.
- C. Wait five seconds before repeating the connection attempt to the database.
- D. Set a maximum number of connection attempts to 10 and report an error on subsequent connections.
- E. Increase connection repeat attempts exponentially up to 120 seconds.

Correct Answer: ACD

You are building a B2B web application that uses Azure B2B collaboration for authentication. Paying customers authenticate to Azure B2B using federation.

The application allows users to sign up for trial accounts using any email address.

When a user converts to a paying customer, the data associated with the trial should be kept, but the user must authenticate using federation.

You need to update the user in Azure Active Directory (Azure AD) when they convert to a paying customer.

Which Graph API parameter is used to change authentication from one-time passcodes to federation?

- A. resetRedemption
- B. Status
- C. userFlowType
- D. invitedUser

Correct Answer: B

HOTSPOT

You develop an image upload service that is exposed using Azure API Management. Images are analyzed after upload for automatic tagging.

Images over 500 KB are processed by a different backend that offers a lower tier of service that costs less money. The lower tier of service is denoted by a header named x-large-request. Images over 500 KB must never be processed by backends for smaller images and must always be charged the lower price.

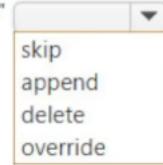
You need to implement API Management policies to ensure that images are processed correctly.

How should you complete the API Management inbound policy? To answer, select the appropriate options in the answer area.

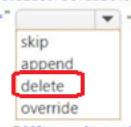
NOTE: Each correct selection is worth one point.

Answer Area

```
<inbound>
  <base/>
  <set-variable name="imageSize" value="@{context.Request.Headers["Content-Length"][[0]]}" />
  <choose>
    <when condition="@{int.Parse(context.Variables.GetValueOrDefault<string>("imageSize"))<512000}">
      <set-headername="x-large-request"exists-action=""
        <value>true</value>
      </set-header>
    </when>
    <otherwise>
      <|>
        set-body
        forward-request
        set-backend-service
        set-query-parameter
        base-url
        dimension
        vary-by-header
        publish-to-dapr
      <|>
        = "{{large-image-host}}"/>
    </otherwise>
  </choose>
</inbound>
```

**Answer Area**

```
<inbound>
  <base/>
  <set-variable name="imageSize" value="@{context.Request.Headers["Content-Length"][[0]]}" />
  <choose>
    <when condition="@{int.Parse(context.Variables.GetValueOrDefault<string>("imageSize"))<512000}">
      <set-headername="x-large-request"exists-action=""
        <value>true</value>
      </set-header>
    </when>
    <otherwise>
      <|>
        set-body
        base-url
        set-backend-service
        set-query-parameter
    <|>
        = "{{large-image-host}}"/>
    </otherwise>
  </choose>
</inbound>
```

**Correct Answer:**

HOTSPOT

You develop several Azure Functions app functions to process JSON documents from a third-party system. The third-party system publishes events to Azure Event Grid to include hundreds of event types, such as billing, inventory, and shipping updates.

Events must be sent to a single endpoint for the Azure Functions app to process. The events must be filtered by event type before processing. You must have authorization and authentication control to partition your tenants to receive the event data.

You need to configure Azure Event Grid.

Which configuration should you use? To answer, select the appropriate values in the answer area.

NOTE: Each correct selection is worth one point.

Answer Area**Requirement**

Third-party system endpoint to send events

Configuration Value

- system topic
- custom topic
- event domain
- event subscription

Azure Functions app endpoint to handle filtered events

- system topic
- custom topic
- event domain
- event subscription

Answer Area**Requirement**

Third-party system endpoint to send events

Configuration Value

- system topic
- custom topic
- event domain
- event subscription

Correct Answer:

Azure Functions app endpoint to handle filtered events

- system topic
- custom topic
- event domain
- event subscription

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-AzureRmServiceBusQueue  
  -ResourceGroupName fridge-rg  
  -NamespaceName fridge-ns  
  -Name fridge-q  
  -EnablePartitioning $False
```

C.

```
connectionString=$(az servicebus namespace authorization-rule keys list  
  --resource-group fridge-rg  
  --fridge-ns fridge-ns  
  --name RootManageSharedAccessKey  
  --query primaryConnectionString --output tsv)
```

D.

```
az servicebus namespace create  
  --resource-group fridge-rg  
  --name fridge-ns  
  --location fridge-loc
```

Correct Answer: B

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?

- New-AzureRmServiceBusNamespace
A. -ResourceGroupName fridge-rg
 -NamespaceName fridge-ns
 -Location fridge-loc
- New-AzureRmResourceGroup
B. -Name fridge-rg
 -Location fridge-loc
- New-AzureRmServiceBusQueue
C. -ResourceGroupName fridge-rg
 -NamespaceName fridge-ns
 -Name fridge-q
 -EnablePartitioning \$False
- D.
`connectionString=$(az servicebus namespace authorization-rule keys list
 --resource-group fridge-rg
 --namespace-name fridge-ns
 --name RootManageSharedAccessKey
 --query primaryConnectionString --output tsv)`

Correct Answer: C

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?

- ```
az group create
A. --name fridge-rg
 --location fridge-loc

New-AzureRmServiceBusNamespace
B. -ResourceGroupName fridge-rg
 -NamespaceName fridge-ns
 -Location fridge-loc

New-AzureRmServiceBusQueue
C. -NamespaceName fridge-ns
 -Name fridge-q
 -EnablePartitioning $False

New-AzureRmResourceGroup
D. -Name fridge-rg
 -Location fridge-loc
```

Correct Answer: C

DRAG DROP

You develop and deploy several APIs to Azure API Management.

You create the following policy fragment named APICounts:

```
<fragment>
 <emit-metric value="1" namespace="custom-metrics">
 <dimension name="User ID" />
 <dimension name="Operation ID" />
 <dimension name="API ID" />
 <dimension name="Client IP" value="@{context.RequestIpAddress)" />
 </emit-metric>
</fragment>
```

The policy fragment must be reused across various scopes and APIs. The policy fragment must be applied to all APIs and run when a calling system invokes any API.

You need to implement the policy fragment.

### XML elements

- name
- inbound
- outbound
- set-variable
- fragment-id
- include-fragment

### Answer Area

```
<policies>
 <[XML]>
 <[XML] [XML] = "APICounts" />
 <base />
 </[XML]>
 ...
 </policies>
```

### Answer Area

Correct Answer:

```
<policies>
 <[inbound]>
 <[include-fragment] [fragment-id] = "APICounts" />
 <base />
 </[inbound]>
 ...
 </policies>
```

**HOTSPOT**

You are developing a solution that uses several Azure Service Bus queues. You create an Azure Event Grid subscription for the Azure Service Bus namespace. You use Azure Functions as subscribers to process the messages.

You need to emit events to Azure Event Grid from the queues. You must use the principal of least privilege and minimize costs.

Which Azure Service Bus values should you use? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

**Answer Area**

<b>Configuration</b>	<b>Value</b>
Tier	<input type="button" value="Basic"/> <input type="button" value="Standard"/> <input type="button" value="Premium"/>
Access control (IAM) level	<input type="button" value="Contributor"/> <input type="button" value="Data Receiver"/> <input type="button" value="Data Sender"/> <input type="button" value="Data Owner"/>

**Answer Area**

<b>Configuration</b>	<b>Value</b>
Tier	<input type="button" value="Basic"/> <input type="button" value="Standard"/> <input checked="" type="button" value="Premium"/>
Access control (IAM) level	<input checked="" type="button" value="Contributor"/> <input type="button" value="Data Receiver"/> <input type="button" value="Data Sender"/> <input type="button" value="Data Owner"/>

**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 -

Wide World Importers is moving all their datacenters to Azure. The company has developed several applications and services to support supply chain operations and would like to leverage serverless computing where possible.

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 -

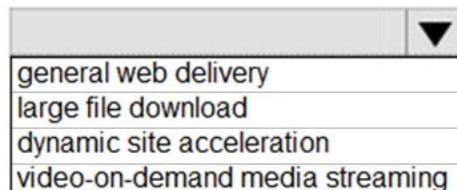
You need to configure Azure CDN for the Shipping web site.

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

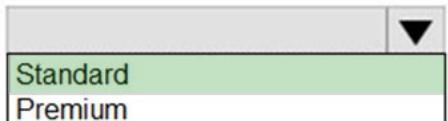
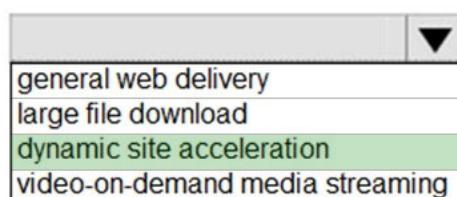
NOTE: Each correct selection is worth one point.

Hot Area:

## Answer Area

Option	Value
Tier	
Profile	
Optimization	

## Answer Area

Option	Value
Tier	
Correct Answer: Profile	
Optimization	

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>

## 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 -

Wide World Importers is moving all their datacenters to Azure. The company has developed several applications and services to support supply chain operations and would like to leverage serverless computing where possible.

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 -

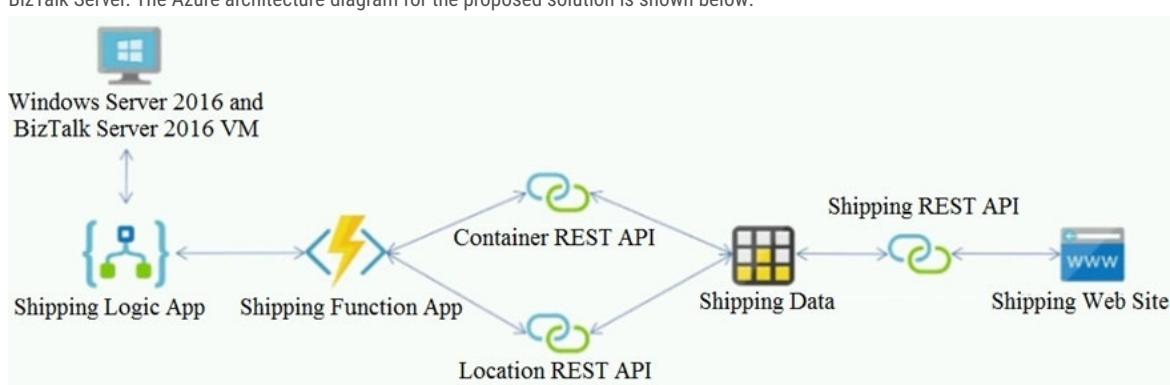
The application uses MongoDB JSON document storage database for all container and transport information.

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	<ul style="list-style-type: none"><li>Azure Site Recovery</li><li>Azure Backup</li><li>Azure Data Box</li><li>Azure Migrate</li></ul>
Performance	<ul style="list-style-type: none"><li>Azure Network Watcher</li><li>Azure Traffic Manager</li><li>ExpressRoute</li><li>Accelerated Networking</li></ul>

## Answer Area

Issue	Tool				
Backup and Restore Correct Answer:	<table border="1"><tr><td>Azure Site Recovery</td></tr><tr><td>Azure Backup</td></tr><tr><td>Azure Data Box</td></tr><tr><td>Azure Migrate</td></tr></table>	Azure Site Recovery	Azure Backup	Azure Data Box	Azure Migrate
Azure Site Recovery					
Azure Backup					
Azure Data Box					
Azure Migrate					
Performance	<table border="1"><tr><td>Azure Network Watcher</td></tr><tr><td>Azure Traffic Manager</td></tr><tr><td>ExpressRoute</td></tr><tr><td>Accelerated Networking</td></tr></table>	Azure Network Watcher	Azure Traffic Manager	ExpressRoute	Accelerated Networking
Azure Network Watcher					
Azure Traffic Manager					
ExpressRoute					
Accelerated Networking					

Box 1: Azure Backup -

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.

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/>

**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)
[redacted]
[redacted]
StringTokenProvider("storage", "msi")
[redacted]
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>

## 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
<pre>var copyOptions = new CopyOptions { }; var context = new Value = (source, destination) =&gt; Task.FromResult(true); context. Value = (source, destination) =&gt; Task.FromResult(true); await TransferManager.CopyAsync(blob, GetDRBlob(blob), isServiceCopy: Value , context: context, options:copyOptions);</pre>

**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.

Note: The TransferContext object comes in two forms: SingleTransferContext and DirectoryTransferContext. The former is for transferring a single file and the latter is for transferring a directory of files.

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>

**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.

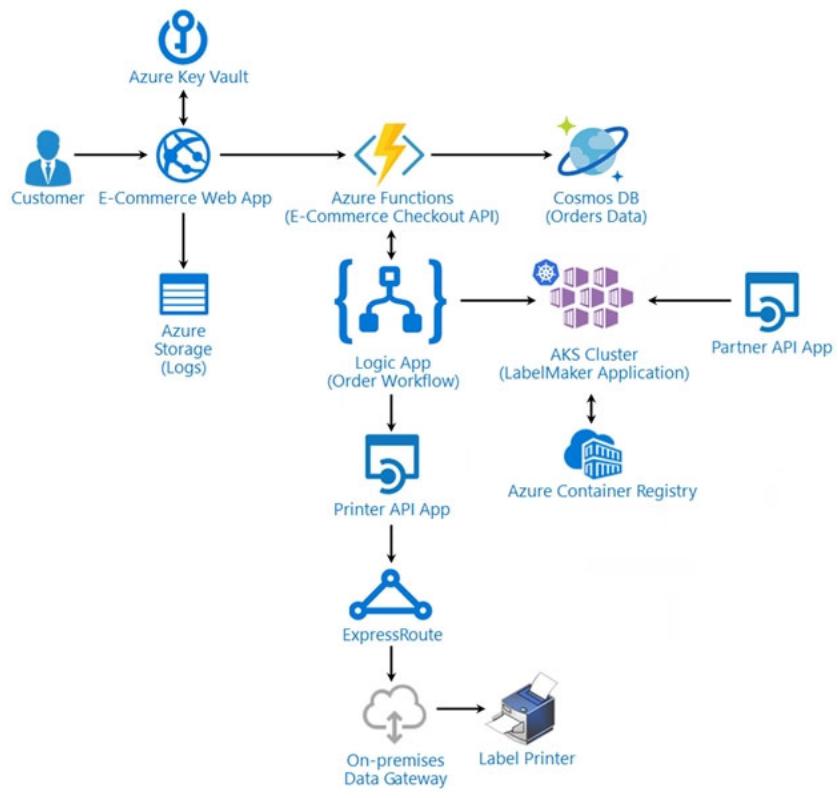
Requirements. LabelMaker app -

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	<div style="border: 1px solid black; padding: 5px; width: 150px;"><p>Strong</p><p>Bounded-staleness</p><p>Session</p><p>Eventual</p></div>
API	<div style="border: 1px solid black; padding: 5px; width: 150px;"><p>SQL</p><p>MongoDB</p><p>Graph</p><p>Table</p></div>

## Answer Area

Setting	Value				
Consistency Level Correct Answer:	<table border="1"><tr><td>Strong</td></tr><tr><td>Bounded-staleness</td></tr><tr><td>Session</td></tr><tr><td>Eventual</td></tr></table>	Strong	Bounded-staleness	Session	Eventual
Strong					
Bounded-staleness					
Session					
Eventual					
API	<table border="1"><tr><td>SQL</td></tr><tr><td>MongoDB</td></tr><tr><td>Graph</td></tr><tr><td>Table</td></tr></table>	SQL	MongoDB	Graph	Table
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).

## 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.

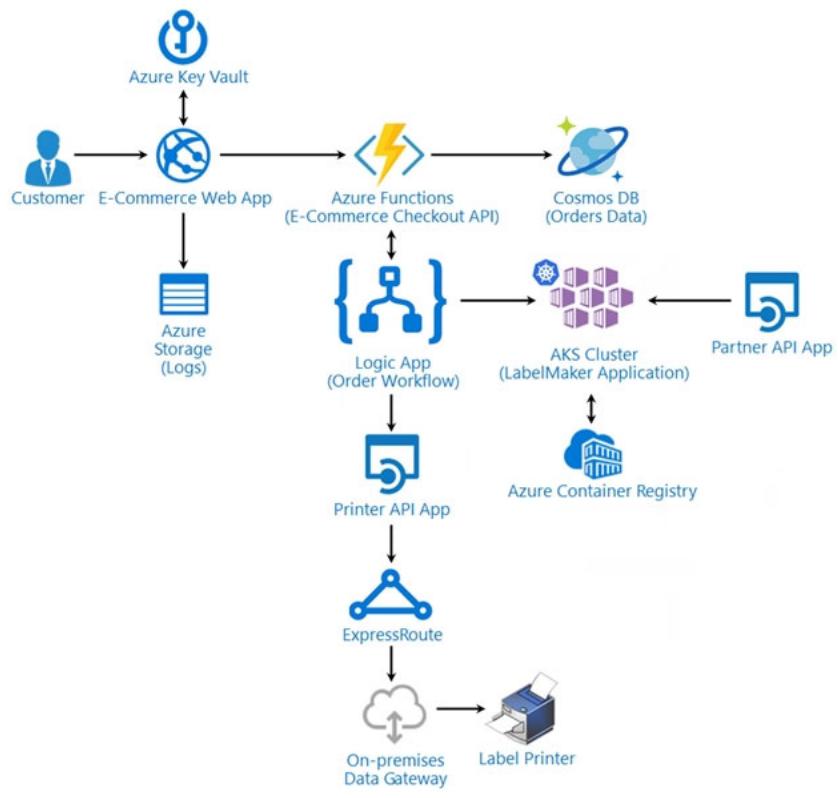
### Requirements. LabelMaker app -

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
 Lineltems li
JOIN
 li
 IN
 o.line_items
 li.line_items
 o.address
ORDER BY
 o.address.city
 li.address.city
 o.city
 li.city
 ASC
```

## Answer Area

Correct Answer:

```
SELECT li.id AS lineitemid, li.price
FROM Orders o
JOIN LineItems li
IN
li
o
ORDER BY
ASC
o.address.city
li.address.city
o.city
li.city
```

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 -

Box 4: o.city -

The city field is in Order, not in the 2s.

**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 -

VanArsdel, Ltd. is a global office supply company. The company is based in Canada and has retail store locations across the world. The company is developing several cloud-based solutions to support their stores, distributors, suppliers, and delivery services.

Current environment -

Corporate website -

The company provides a public website located at <http://www.vanarsdelltd.com>. The website consists of a React JavaScript user interface, HTML, CSS, image assets, and several APIs hosted in Azure Functions.

Retail Store Locations -

The company supports thousands of store locations globally. Store locations send data every hour to an Azure Blob storage account to support inventory, purchasing and delivery services. Each record includes a location identifier and sales transaction information.

Requirements -

The application components must meet the following requirements:

Corporate website -

Secure the website by using SSL.

Minimize costs for data storage and hosting.

Implement native GitHub workflows for continuous integration and continuous deployment (CI/CD).

Distribute the website content globally for local use.

Implement monitoring by using Application Insights and availability web tests including SSL certificate validity and custom header value verification.

The website must have 99.95 percent uptime.

Retail store locations -

Azure Functions must process data immediately when data is uploaded to Blob storage. Azure Functions must update Azure Cosmos DB by using native SQL language queries.

Audit store sale transaction information nightly to validate data, process sales financials, and reconcile inventory.

Delivery services -

Store service telemetry data in Azure Cosmos DB by using an Azure Function. Data must include an item id, the delivery vehicle license plate, vehicle package capacity, and current vehicle location coordinates.

Store delivery driver profile information in Azure Active Directory (Azure AD) by using an Azure Function called from the corporate website.

**Inventory services -**

The company has contracted a third-party to develop an API for inventory processing that requires access to a specific blob within the retail store storage account for three months to include read-only access to the data.

**Security -**

All Azure Functions must centralize management and distribution of configuration data for different environments and geographies, encrypted by using a company-provided RSA-HSM key.

Authentication and authorization must use Azure AD and services must use managed identities where possible.

**Issues -**

**Retail Store Locations -**

You must perform a point-in-time restoration of the retail store location data due to an unexpected and accidental deletion of data.

Azure Cosmos DB queries from the Azure Function exhibit high Request Unit (RU) usage and contain multiple, complex queries that exhibit high point read latency for large items as the function app is scaling.

**Question**

**HOTSPOT -**

You need to implement the Azure Function for delivery driver profile information.

Which configurations 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**

### **Configuration      Value**

Code library

- Microsoft Authentication Library (MSAL)
- Microsoft Azure Key Vault SDK
- Azure Identity library

API

- Microsoft Graph
- Azure Active Directory Graph
- Azure Key Vault

## **Answer Area**

### **Configuration      Value**

Code library

- Microsoft Authentication Library (MSAL)
- Microsoft Azure Key Vault SDK
- Azure Identity library

Correct Answer:

API

- Microsoft Graph
- Azure Active Directory Graph
- Azure Key Vault

Box 1: Azure Identity library -

Store delivery driver profile information in Azure Active Directory (Azure AD) by using an Azure Function called from the corporate website.

We recommend that you use a managed identity for applications deployed to Azure.

The preceding authentication scenarios are supported by the Azure Identity client library and integrated with Key Vault SDKs.

Note: What is Managed Service Identity?

Azure Key Vault avoids the need to store keys and secrets in application code or source control. However, in order to retrieve keys and secrets from Azure Key

Vault, you need to authorize a user or application with Azure Key Vault, which in its turn needs another credential. Managed Service Identity avoids the need of storing credentials for Azure Key Vault in application or environment settings by creating a Service Principal for each application or cloud service on which

Managed Service Identity is enabled. This Service Principal enables you to call a local MSI endpoint to get an access token from Azure AD using the credentials of the Service Principal. This token is then used to authenticate to an Azure Service, for example Azure Key Vault.

Box 2: Azure Key Vault -

Azure Key Vault allows you to securely access sensitive information from within your applications:

\* Keys, secrets, and certificates are protected without your having to write the code yourself, and you can easily use them from your applications.

Use Azure Key Vault to store only secrets for your application. Examples of secrets that should be stored in Key Vault include:

Client application secrets -

Connection strings -

Passwords -

Shared access keys -

SSH keys -

Reference:

<https://docs.microsoft.com/en-us/azure/key-vault/general/developers-guide> <https://integration.team/blog/retrieve-azure-key-vault-secrets-using-azure-functions-and-managed-service-identity>

## 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 -

VanArsdel, Ltd. is a global office supply company. The company is based in Canada and has retail store locations across the world. The company is developing several cloud-based solutions to support their stores, distributors, suppliers, and delivery services.

Current environment -

Corporate website -

The company provides a public website located at <http://www.vanarsdelltd.com>. The website consists of a React JavaScript user interface, HTML, CSS, image assets, and several APIs hosted in Azure Functions.

Retail Store Locations -

The company supports thousands of store locations globally. Store locations send data every hour to an Azure Blob storage account to support inventory, purchasing and delivery services. Each record includes a location identifier and sales transaction information.

Requirements -

The application components must meet the following requirements:

Corporate website -

Secure the website by using SSL.

Minimize costs for data storage and hosting.

Implement native GitHub workflows for continuous integration and continuous deployment (CI/CD).

Distribute the website content globally for local use.

Implement monitoring by using Application Insights and availability web tests including SSL certificate validity and custom header value verification.

The website must have 99.95 percent uptime.

Retail store locations -

Azure Functions must process data immediately when data is uploaded to Blob storage. Azure Functions must update Azure Cosmos DB by using native SQL language queries.

Audit store sale transaction information nightly to validate data, process sales financials, and reconcile inventory.

Delivery services -

Store service telemetry data in Azure Cosmos DB by using an Azure Function. Data must include an item id, the delivery vehicle license plate, vehicle package capacity, and current vehicle location coordinates.

Store delivery driver profile information in Azure Active Directory (Azure AD) by using an Azure Function called from the corporate website.

Inventory services -

The company has contracted a third-party to develop an API for inventory processing that requires access to a specific blob within the retail store

storage account for three months to include read-only access to the data.

Security -

All Azure Functions must centralize management and distribution of configuration data for different environments and geographies, encrypted by using a company-provided RSA-HSM key.

Authentication and authorization must use Azure AD and services must use managed identities where possible.

Issues -

Retail Store Locations -

You must perform a point-in-time restoration of the retail store location data due to an unexpected and accidental deletion of data.

Azure Cosmos DB queries from the Azure Function exhibit high Request Unit (RU) usage and contain multiple, complex queries that exhibit high point read latency for large items as the function app is scaling.

**Question**

You need to grant access to the retail store location data for the inventory service development effort.

What should you use?

- A. Azure AD access token
- B. Azure RBAC role
- C. Shared access signature (SAS) token
- D. Azure AD ID token
- E. Azure AD refresh token

**Correct Answer:** C

A shared access signature (SAS) provides secure delegated access to resources in your storage account. With a SAS, you have granular control over how a client can access your data. For example:

What resources the client may access.

What permissions they have to those resources.

How long the SAS is valid.

Note: Inventory services:

The company has contracted a third-party to develop an API for inventory processing that requires access to a specific blob within the retail store storage account for three months to include read-only access to the data.

Reference:

<https://docs.microsoft.com/en-us/azure/storage/common/storage-sas-overview>

## 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 -

VanArsdel, Ltd. is a global office supply company. The company is based in Canada and has retail store locations across the world. The company is developing several cloud-based solutions to support their stores, distributors, suppliers, and delivery services.

Current environment -

Corporate website -

The company provides a public website located at <http://www.vanarsdelltd.com>. The website consists of a React JavaScript user interface, HTML, CSS, image assets, and several APIs hosted in Azure Functions.

Retail Store Locations -

The company supports thousands of store locations globally. Store locations send data every hour to an Azure Blob storage account to support inventory, purchasing and delivery services. Each record includes a location identifier and sales transaction information.

Requirements -

The application components must meet the following requirements:

Corporate website -

Secure the website by using SSL.

Minimize costs for data storage and hosting.

Implement native GitHub workflows for continuous integration and continuous deployment (CI/CD).

Distribute the website content globally for local use.

Implement monitoring by using Application Insights and availability web tests including SSL certificate validity and custom header value verification.

The website must have 99.95 percent uptime.

Retail store locations -

Azure Functions must process data immediately when data is uploaded to Blob storage. Azure Functions must update Azure Cosmos DB by using native SQL language queries.

Audit store sale transaction information nightly to validate data, process sales financials, and reconcile inventory.

Delivery services -

Store service telemetry data in Azure Cosmos DB by using an Azure Function. Data must include an item id, the delivery vehicle license plate, vehicle package capacity, and current vehicle location coordinates.

Store delivery driver profile information in Azure Active Directory (Azure AD) by using an Azure Function called from the corporate website.

Inventory services -

The company has contracted a third-party to develop an API for inventory processing that requires access to a specific blob within the retail store

storage account for three months to include read-only access to the data.

Security -

All Azure Functions must centralize management and distribution of configuration data for different environments and geographies, encrypted by using a company-provided RSA-HSM key.

Authentication and authorization must use Azure AD and services must use managed identities where possible.

Issues -

Retail Store Locations -

You must perform a point-in-time restoration of the retail store location data due to an unexpected and accidental deletion of data.

Azure Cosmos DB queries from the Azure Function exhibit high Request Unit (RU) usage and contain multiple, complex queries that exhibit high point read latency for large items as the function app is scaling.

**Question**

HOTSPOT -

You need to reliably identify the delivery driver profile information.

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

NOTE: Each correct selection is worth one point.

Hot Area:

## Answer Area

Configuration	Value
JSON web token (JWT) type	<input type="checkbox"/> ID <input type="checkbox"/> Refresh <input type="checkbox"/> Access
Payload claim value	<input type="checkbox"/> oid <input type="checkbox"/> aud <input type="checkbox"/> idp

## Answer Area

### Configuration

JSON web token (JWT) type

Correct Answer:

	▼
ID	
Refresh	
Access	

### Payload claim value

	▼
oid	
aud	
idp	

Box 1: ID -

Scenario: Store delivery driver profile information in Azure Active Directory (Azure AD) by using an Azure Function called from the corporate website.

ID token - A JWT that contains claims that you can use to identify users in your application. This token is securely sent in HTTP requests for communication between two components of the same application or service. You can use the claims in an ID token as you see fit. They're commonly used to display account information or to make access control decisions in an application. ID tokens are signed, but they're not encrypted. When your application or API receives an ID token, it must validate the signature to prove that the token is authentic. Your application or API must also validate a few claims in the token to prove that it's valid.

Depending on the scenario requirements, the claims validated by an application can vary, but your application must perform some common claim validations in every scenario.

Box 2: Oid -

Oid - The immutable identifier for the "principal" of the request - the user or service principal whose identity has been verified. In ID tokens and app+user tokens, this is the object ID of the user. In app-only tokens, this is the object ID of the calling service principal. It can also be used to perform authorization checks safely and as a key in database tables. This ID uniquely identifies the principal across applications - two different applications signing in the same user will receive the same value in the oid claim.

Incorrect:

Aud - Identifies the intended recipient of the token. For Azure AD B2C, the audience is the application ID. Your application should validate this value and reject the token if it doesn't match. Audience is synonymous with resource.

Idp - Records the identity provider that authenticated the subject of the token. This value is identical to the value of the Issuer claim unless the user account not in the same tenant as the issuer - guests, for instance. If the claim isn't present, it means that the value of iss can be used instead. For personal accounts being used in an organizational context (for instance, a personal account invited to an Azure AD tenant), the idp claim may be 'live.com' or an STS URI containing the Microsoft account tenant.

Reference:

<https://docs.microsoft.com/en-us/azure/active-directory-b2c/tokens-overview> <https://docs.microsoft.com/en-us/azure/active-directory/develop/access-tokens>

## 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 -

VanArsdel, Ltd. is a global office supply company. The company is based in Canada and has retail store locations across the world. The company is developing several cloud-based solutions to support their stores, distributors, suppliers, and delivery services.

Current environment -

Corporate website -

The company provides a public website located at <http://www.vanarsdelltd.com>. The website consists of a React JavaScript user interface, HTML, CSS, image assets, and several APIs hosted in Azure Functions.

Retail Store Locations -

The company supports thousands of store locations globally. Store locations send data every hour to an Azure Blob storage account to support inventory, purchasing and delivery services. Each record includes a location identifier and sales transaction information.

Requirements -

The application components must meet the following requirements:

Corporate website -

Secure the website by using SSL.

Minimize costs for data storage and hosting.

Implement native GitHub workflows for continuous integration and continuous deployment (CI/CD).

Distribute the website content globally for local use.

Implement monitoring by using Application Insights and availability web tests including SSL certificate validity and custom header value verification.

The website must have 99.95 percent uptime.

Retail store locations -

Azure Functions must process data immediately when data is uploaded to Blob storage. Azure Functions must update Azure Cosmos DB by using native SQL language queries.

Audit store sale transaction information nightly to validate data, process sales financials, and reconcile inventory.

Delivery services -

Store service telemetry data in Azure Cosmos DB by using an Azure Function. Data must include an item id, the delivery vehicle license plate, vehicle package capacity, and current vehicle location coordinates.

Store delivery driver profile information in Azure Active Directory (Azure AD) by using an Azure Function called from the corporate website.

Inventory services -

The company has contracted a third-party to develop an API for inventory processing that requires access to a specific blob within the retail store

storage account for three months to include read-only access to the data.

Security -

All Azure Functions must centralize management and distribution of configuration data for different environments and geographies, encrypted by using a company-provided RSA-HSM key.

Authentication and authorization must use Azure AD and services must use managed identities where possible.

Issues -

Retail Store Locations -

You must perform a point-in-time restoration of the retail store location data due to an unexpected and accidental deletion of data.

Azure Cosmos DB queries from the Azure Function exhibit high Request Unit (RU) usage and contain multiple, complex queries that exhibit high point read latency for large items as the function app is scaling.

**Question**

You need to secure the Azure Functions to meet the security requirements.

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

NOTE: Each correct selection is worth one point.

- A. Store the RSA-HSM key in Azure Key Vault with soft-delete and purge-protection features enabled.
- B. Store the RSA-HSM key in Azure Blob storage with an immutability policy applied to the container.
- C. Create a free tier Azure App Configuration instance with a new Azure AD service principal.
- D. Create a standard tier Azure App Configuration instance with an assigned Azure AD managed identity.
- E. Store the RSA-HSM key in Azure Cosmos DB. Apply the built-in policies for customer-managed keys and allowed locations.

**Correct Answer:** AD

Scenario: All Azure Functions must centralize management and distribution of configuration data for different environments and geographies, encrypted by using a company-provided RSA-HSM key.

Microsoft Azure Key Vault is a cloud-hosted management service that allows users to encrypt keys and small secrets by using keys that are protected by hardware security modules (HSMs).

You need to create a managed identity for your application.

Reference:

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

**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	], "displayName": "ContentReviewer",
allowedMemberTypes	"id": "e1c2ade8-98f8-45fd-aa4a-6d24b512c22a",
allowedAccountTypes	"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>

## 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>

## 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
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.

Azure 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>

## 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

- volumeMounts
- 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>

## 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",
 "platf",
 "sid",
 "tenant_ctry",
 "sid",
 "upn",
 "email",
 "enfpolids"],
```

## Answer Area

Correct Answer:

```
"optionalClaims": [
 "urn:ietf:params:openid-connect:claim:acct",
 "urn:ietf:params:openid-connect:claim:platf",
 "urn:ietf:params:openid-connect:claim:sid", sid
 "urn:ietf:params:openid-connect:claim:tenant_ctry",
 "urn:ietf:params:openid-connect:claim:sid",
 "urn:ietf:params:openid-connect:claim:upn",
 "urn:ietf:params:openid-connect:claim:email", email
 "urn:ietf:params:openid-connect:claim:enfpolids"
],
```

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.

### 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 -

Wide World Importers is moving all their datacenters to Azure. The company has developed several applications and services to support supply chain operations and would like to leverage serverless computing where possible.

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 -

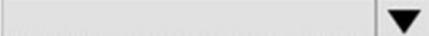
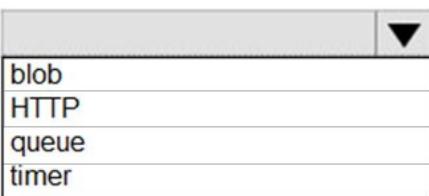
You need to secure the Shipping Function app.

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

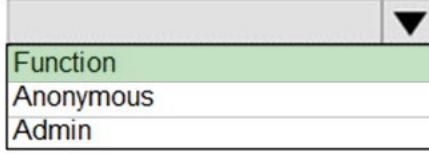
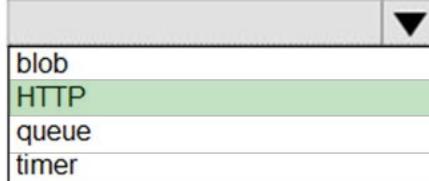
NOTE: Each correct selection is worth one point.

Hot Area:

## Answer Area

Setting	Value
Authorization level	 Function Anonymous Admin
User claims	 JSON Web Token (JWT) Shared Access Signature (SAS) token API Key
Trigger type	 blob HTTP queue timer

## 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>

## 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 -

Wide World Importers is moving all their datacenters to Azure. The company has developed several applications and services to support supply chain operations and would like to leverage serverless computing where possible.

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 -

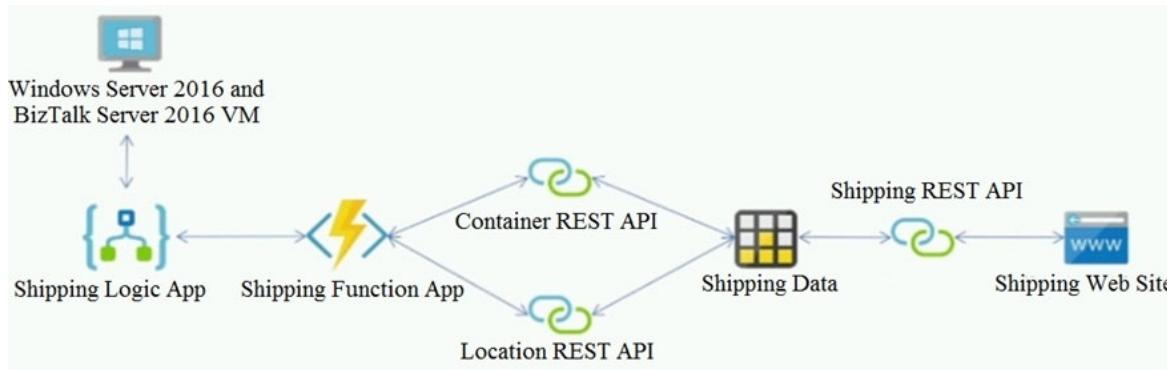
The application uses MongoDB JSON document storage database for all container and transport information.

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>

**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.cpandl.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.

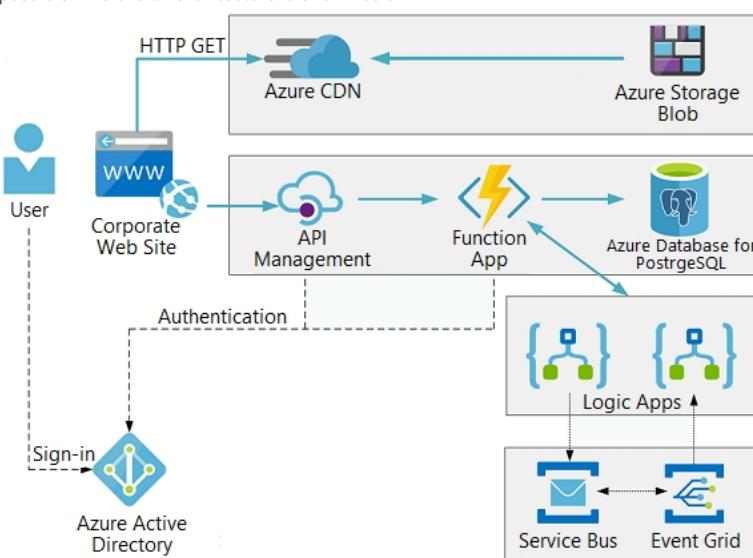
API Management and Azure Function App functions are used to process and store data in Azure Database for PostgreSQL. API Management is used to broker communications to the Azure Function app functions for Logic app integration. Logic apps are used to orchestrate the data processing while Service Bus 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,
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 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" value="cpandlkeyvault"/> ▼	.vault.azure.net/secrets/	<input type="text" value="cpandlkeyvault"/> ▼ /
	<input type="text" value="PostgreSQL Conn"/>		<input type="text" value="PostgreSQL Conn"/>
	<input type="text" value="80df3e46ffcd4f1cb187f79905e9a1e8"/>		<input type="text" value="80df3e46ffcd4f1cb187f79905e9a1e8"/>

Variable type to access Azure Key Vault secret values:

<input type="text" value="Environment"/> ▼
<input type="text" value="Session"/>
<input type="text" value="ViewState"/>
<input type="text" value="QueryString"/>

**Correct Answer:**

**Answer Area**

REST API Endpoint:

https://	cpandlkeyvault	.vault.azure.net/secrets/	cpandlkeyvault	/
	PostgreSQLConn		PostgreSQLConn	

Variable type to access Azure Key Vault secret values:

Environment
Session
ViewState
QueryString

Azure database connection string retrieve REST API vault.azure.net/secrets/

Box 1: cpandlkeyvault -

We specify the key vault, cpandlkeyvault.

Scenario: The database connection string is stored in Azure Key Vault with the following attributes:

Azure Key Vault name: cpandlkeyvault

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>

## 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.

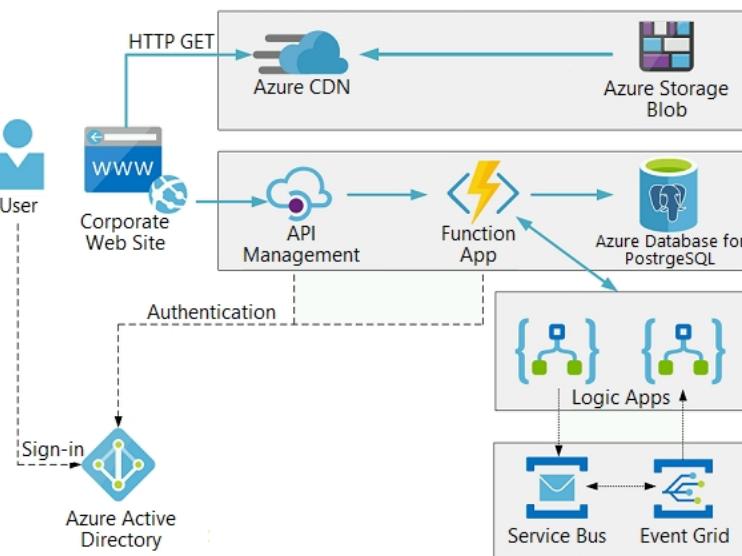
API Management and Azure Function App functions are used to process and store data in Azure Database for PostgreSQL. API Management is used to broker communications to the Azure Function app functions for Logic app integration. Logic apps are used to orchestrate the data processing while Service Bus 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(HttpRequest req)
RA14 {
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.
<b>Correct Answer:</b> 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.	

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>

## 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.

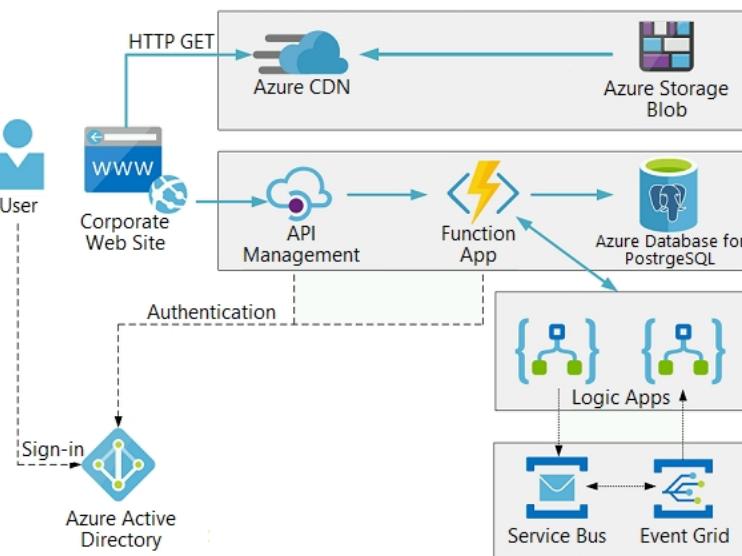
API Management and Azure Function App functions are used to process and store data in Azure Database for PostgreSQL. API Management is used to broker communications to the Azure Function app functions for Logic app integration. Logic apps are used to orchestrate the data processing while Service Bus 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 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 black; padding: 5px;"><input type="checkbox"/> Check HTTP header <input type="checkbox"/> Restrict caller IPs <input type="checkbox"/> Limit call rate by key <input checked="" type="checkbox"/> Validate JWT</div>
Policy section	<div style="border: 1px solid black; padding: 5px;"><input type="checkbox"/> Inbound <input type="checkbox"/> Outbound</div>

### Answer Area

Setting	Value
Policy	<div style="border: 1px solid black; padding: 5px;"><input type="checkbox"/> Check HTTP header <input type="checkbox"/> Restrict caller IPs <input type="checkbox"/> Limit call rate by key <input checked="" type="checkbox"/> Validate JWT</div>
Policy section	<div style="border: 1px solid black; padding: 5px;"><input type="checkbox"/> Inbound <input checked="" type="checkbox"/> Outbound</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>

## 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.

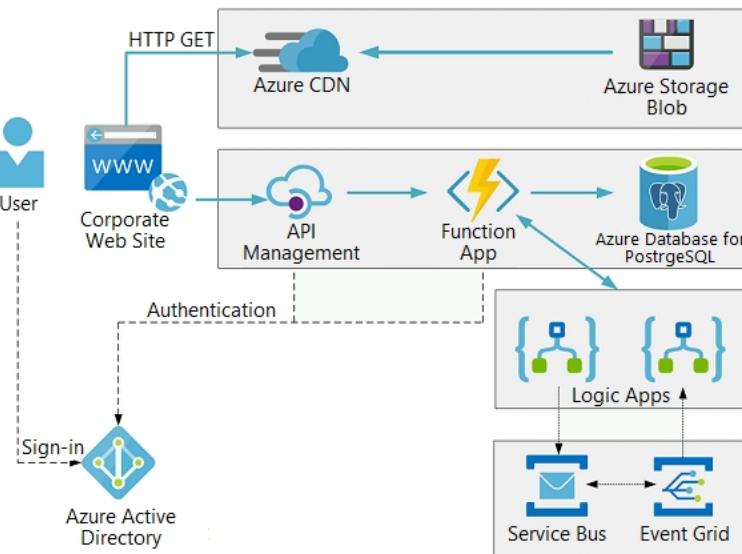
API Management and Azure Function App functions are used to process and store data in Azure Database for PostgreSQL. API Management is used to broker communications to the Azure Function app functions for Logic app integration. Logic apps are used to orchestrate the data processing while Service Bus 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>

## 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.

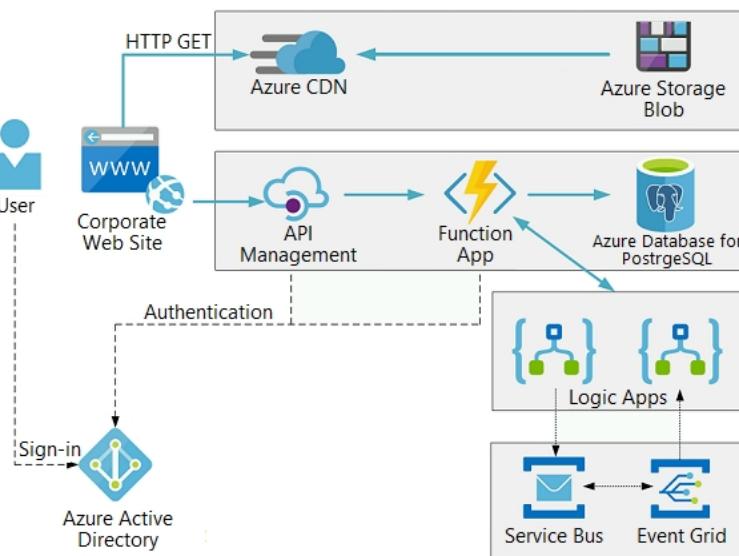
API Management and Azure Function App functions are used to process and store data in Azure Database for PostgreSQL. API Management is used to broker communications to the Azure Function app functions for Logic app integration. Logic apps are used to orchestrate the data processing while Service Bus 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(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	<input type="button" value="▼"/> anonymous function admin
managed identity	<input type="button" value="▼"/> system-assigned user-assigned

### Answer Area

Setting	Value
authentication level	<input type="button" value="▼"/> anonymous <b>function</b> admin
managed identity	<input type="button" value="▼"/> <b>system-assigned</b> user-assigned

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>

## 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.

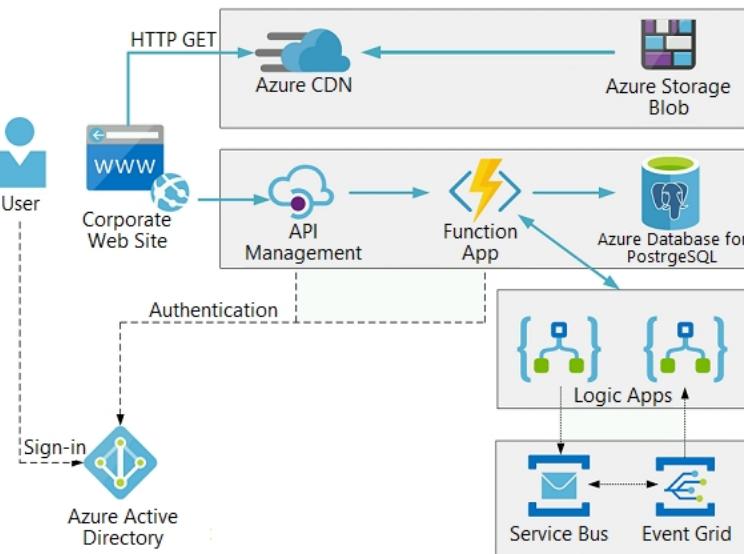
API Management and Azure Function App functions are used to process and store data in Azure Database for PostgreSQL. API Management is used to broker communications to the Azure Function app functions for Logic app integration. Logic apps are used to orchestrate the data processing while Service Bus 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(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 checked="" type="checkbox"/> Standard <input type="checkbox"/> Premium
RBAC role	<input type="checkbox"/> Owner <input type="checkbox"/> Contributor <input checked="" type="checkbox"/> Azure Service Bus Data Owner <input type="checkbox"/> Azure Service Bus Data Receiver

## Answer Area

Setting	Value				
Tier	<table border="1"><tr><td>Basic</td></tr><tr><td>Standard</td></tr><tr style="background-color: #90EE90;"><td>Premium</td></tr></table>	Basic	Standard	Premium	
Basic					
Standard					
Premium					
Correct Answer:					
RBAC role	<table border="1"><tr><td>Owner</td></tr><tr style="background-color: #90EE90;"><td>Contributor</td></tr><tr><td>Azure Service Bus Data Owner</td></tr><tr><td>Azure Service Bus Data Receiver</td></tr></table>	Owner	Contributor	Azure Service Bus Data Owner	Azure Service Bus Data Receiver
Owner					
Contributor					
Azure Service Bus Data Owner					
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>

**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

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>

## 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

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
- D. "PermissionsToKeys wrapkey, unwrapkey, get

### 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>

**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 -

VanArsdel, Ltd. is a global office supply company. The company is based in Canada and has retail store locations across the world. The company is developing several cloud-based solutions to support their stores, distributors, suppliers, and delivery services.

Current environment -

Corporate website -

The company provides a public website located at <http://www.vanarsdelltd.com>. The website consists of a React JavaScript user interface, HTML, CSS, image assets, and several APIs hosted in Azure Functions.

Retail Store Locations -

The company supports thousands of store locations globally. Store locations send data every hour to an Azure Blob storage account to support inventory, purchasing and delivery services. Each record includes a location identifier and sales transaction information.

Requirements -

The application components must meet the following requirements:

Corporate website -

Secure the website by using SSL.

Minimize costs for data storage and hosting.

Implement native GitHub workflows for continuous integration and continuous deployment (CI/CD).

Distribute the website content globally for local use.

Implement monitoring by using Application Insights and availability web tests including SSL certificate validity and custom header value verification.

The website must have 99.95 percent uptime.

Retail store locations -

Azure Functions must process data immediately when data is uploaded to Blob storage. Azure Functions must update Azure Cosmos DB by using native SQL language queries.

Audit store sale transaction information nightly to validate data, process sales financials, and reconcile inventory.

Delivery services -

Store service telemetry data in Azure Cosmos DB by using an Azure Function. Data must include an item id, the delivery vehicle license plate, vehicle package capacity, and current vehicle location coordinates.

Store delivery driver profile information in Azure Active Directory (Azure AD) by using an Azure Function called from the corporate website.

**Inventory services -**

The company has contracted a third-party to develop an API for inventory processing that requires access to a specific blob within the retail store storage account for three months to include read-only access to the data.

**Security -**

All Azure Functions must centralize management and distribution of configuration data for different environments and geographies, encrypted by using a company-provided RSA-HSM key.

Authentication and authorization must use Azure AD and services must use managed identities where possible.

**Issues -**

**Retail Store Locations -**

You must perform a point-in-time restoration of the retail store location data due to an unexpected and accidental deletion of data.

Azure Cosmos DB queries from the Azure Function exhibit high Request Unit (RU) usage and contain multiple, complex queries that exhibit high point read latency for large items as the function app is scaling.

**Question**

You need to reduce read latency for the retail store solution.

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. Create a new composite index for the store location data queries in Azure Cosmos DB. Modify the queries to support parameterized SQL and update the Azure Function app to call the new queries.
- B. Provision an Azure Cosmos DB dedicated gateway. Update the Azure Function app connection string to use the new dedicated gateway endpoint.
- C. Configure Azure Cosmos DB consistency to session consistency. Cache session tokens in a new Azure Redis cache instance after every write. Update reads to use the session token stored in Azure Redis.
- D. Provision an Azure Cosmos DB dedicated gateway. Update blob storage to use the new dedicated gateway endpoint.
- E. Configure Azure Cosmos DB consistency to strong consistency. Increase the RUs for the container supporting store location data.

**Correct Answer: BC**

Azure Cosmos DB queries from the Azure Function exhibit high Request Unit (RU) usage and contain multiple, complex queries that exhibit high point read latency for large items as the function app is scaling.

B: A dedicated gateway is server-side compute that is a front-end to your Azure Cosmos DB account. When you connect to the dedicated gateway, it both routes requests and caches data.

You can provision a dedicated gateway to improve performance at scale.

You must connect to Azure Cosmos DB using the dedicated gateway in order to use the integrated cache. The dedicated gateway has a different endpoint from the standard one provided with your Azure Cosmos DB account. When you connect to your dedicated gateway endpoint, your application sends a request to the dedicated gateway, which then routes the request to different backend nodes. If possible, the integrated cache will serve the result.

C: Azure Cache for Redis perfectly complements Azure database services such as Cosmos DB. It provides a cost-effective solution to scale read and write throughput of your data tier. Store and share database query results, session states, static contents, and more using a common cache-aside pattern.

Reference:

<https://docs.microsoft.com/en-us/azure/architecture/solution-ideas/articles/data-cache-with-redis-cache> <https://docs.microsoft.com/en-us/azure/cosmos-db/dedicated-gateway>

## 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 -

VanArsdel, Ltd. is a global office supply company. The company is based in Canada and has retail store locations across the world. The company is developing several cloud-based solutions to support their stores, distributors, suppliers, and delivery services.

Current environment -

Corporate website -

The company provides a public website located at <http://www.vanarsdelltd.com>. The website consists of a React JavaScript user interface, HTML, CSS, image assets, and several APIs hosted in Azure Functions.

Retail Store Locations -

The company supports thousands of store locations globally. Store locations send data every hour to an Azure Blob storage account to support inventory, purchasing and delivery services. Each record includes a location identifier and sales transaction information.

Requirements -

The application components must meet the following requirements:

Corporate website -

Secure the website by using SSL.

Minimize costs for data storage and hosting.

Implement native GitHub workflows for continuous integration and continuous deployment (CI/CD).

Distribute the website content globally for local use.

Implement monitoring by using Application Insights and availability web tests including SSL certificate validity and custom header value verification.

The website must have 99.95 percent uptime.

Retail store locations -

Azure Functions must process data immediately when data is uploaded to Blob storage. Azure Functions must update Azure Cosmos DB by using native SQL language queries.

Audit store sale transaction information nightly to validate data, process sales financials, and reconcile inventory.

Delivery services -

Store service telemetry data in Azure Cosmos DB by using an Azure Function. Data must include an item id, the delivery vehicle license plate, vehicle package capacity, and current vehicle location coordinates.

Store delivery driver profile information in Azure Active Directory (Azure AD) by using an Azure Function called from the corporate website.

Inventory services -

The company has contracted a third-party to develop an API for inventory processing that requires access to a specific blob within the retail store

storage account for three months to include read-only access to the data.

Security -

All Azure Functions must centralize management and distribution of configuration data for different environments and geographies, encrypted by using a company-provided RSA-HSM key.

Authentication and authorization must use Azure AD and services must use managed identities where possible.

Issues -

Retail Store Locations -

You must perform a point-in-time restoration of the retail store location data due to an unexpected and accidental deletion of data.

Azure Cosmos DB queries from the Azure Function exhibit high Request Unit (RU) usage and contain multiple, complex queries that exhibit high point read latency for large items as the function app is scaling.

**Question**

You need to audit the retail store sales transactions.

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. Update the retail store location data upload process to include blob index tags. Create an Azure Function to process the blob index tags and filter by store location.
- B. Process the change feed logs of the Azure Blob storage account by using an Azure Function. Specify a time range for the change feed data.
- C. Enable blob versioning for the storage account. Use an Azure Function to process a list of the blob versions per day.
- D. Process an Azure Storage blob inventory report by using an Azure Function. Create rule filters on the blob inventory report.
- E. Subscribe to blob storage events by using an Azure Function and Azure Event Grid. Filter the events by store location.

**Correct Answer: BE**

Scenario: Audit store sale transaction information nightly to validate data, process sales financials, and reconcile inventory.

"Process the change feed logs of the Azure Blob storage account by using an Azure Function. Specify a time range for the change feed data":

Change feed support is well-suited for scenarios that process data based on objects that have changed. For example, applications can:

Store, audit, and analyze changes to your objects, over any period of time, for security, compliance or intelligence for enterprise data management.

"Subscribe to blob storage events by using an Azure Function and Azure Event Grid. Filter the events by store location": Azure Storage events allow applications to react to events, such as the creation and deletion of blobs. It does so without the need for complicated code or expensive and inefficient polling services. The best part is you only pay for what you use.

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.

Incorrect Answers:

"Enable blob versioning for the storage account. Use an Azure Function to process a list of the blob versions per day": You can enable Blob storage versioning to automatically maintain previous versions of an object. When blob versioning is enabled, you can access earlier versions of a blob to recover your data if it is modified or deleted.

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>

**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 `n alert `g `E` -scopes `E` -condition "avg Percentage CPU > 8"
- B. az monitor metrics alert create `n alert `g `E` -scopes `E` -condition "avg Percentage CPU > 800"
- C. az monitor metrics alert create `n alert `g `E` -scopes `E` -condition "CPU Usage > 800"
- D. az monitor metrics alert create `n alert `g `E` -scopes `E` -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/azure/monitor/metrics/alert>

## 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>

**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.cpandl.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.

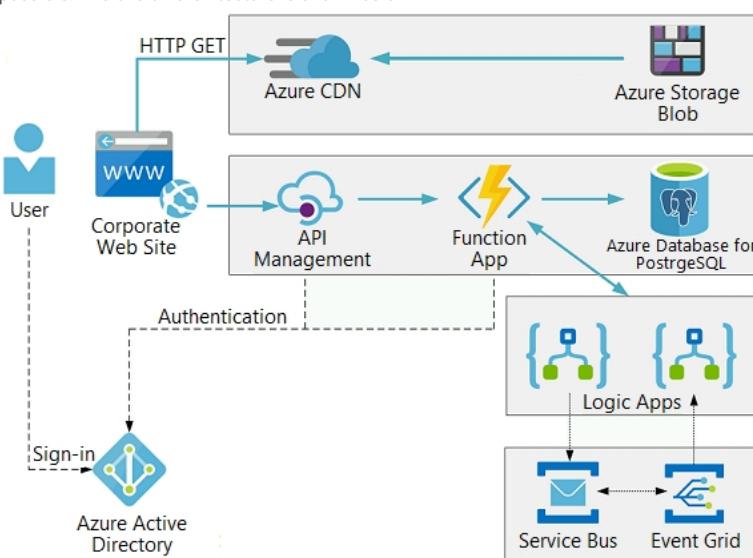
API Management and Azure Function App functions are used to process and store data in Azure Database for PostgreSQL. API Management is used to broker communications to the Azure Function app functions for Logic app integration. Logic apps are used to orchestrate the data processing while Service Bus 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,
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>

## 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.

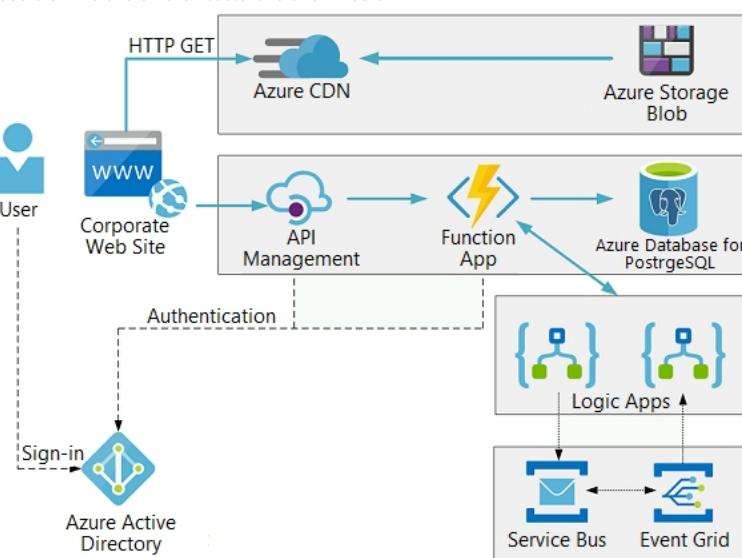
API Management and Azure Function App functions are used to process and store data in Azure Database for PostgreSQL. API Management is used to broker communications to the Azure Function app functions for Logic app integration. Logic apps are used to orchestrate the data processing while Service Bus 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 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

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 Storage Diagnostic settings (preview) interface. The left sidebar lists monitoring options: Files, Table service, Queue service, Monitoring (classic), and Diagnostic settings (preview). The 'Diagnostic settings (preview)' link is highlighted with a red box. The main content area displays 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>

**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.cpandl.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.

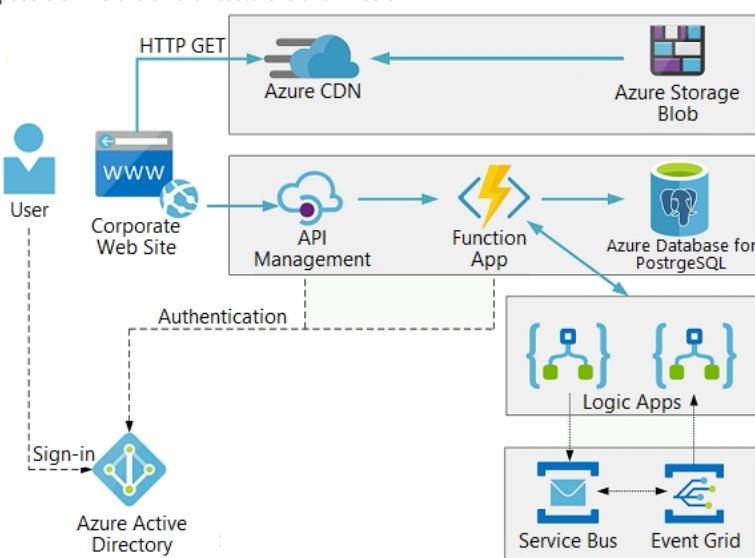
API Management and Azure Function App functions are used to process and store data in Azure Database for PostgreSQL. API Management is used to broker communications to the Azure Function app functions for Logic app integration. Logic apps are used to orchestrate the data processing while Service Bus 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,
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>

**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=VXbGWce53l48Mt8wuotr0GPmyJ/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
```

## 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 EnsureLogging(@event["subject"].ToString());
EG17 }
EG18 }
EG19 return null;
EG20 }
EG21 private void EnsureLogging(string resource)
EG22 {
EG23 . . .
EG24 }
EG25 private async Task SendToAnomalyDetectionService(string uri)
EG26 {
EG27 var content = GetLogData(uri);
EG28 var scoreRequest = new
EG29 {
EG30 Inputs = new Dictionary<string, List<Dictionary<string, string>>()
EG31 {
EG32 {
EG33 "input1",
EG34 new List<Dictionary<string, string>>()
EG35 {
EG36 new Dictionary<string, string>()
EG37 {
EG38 {
EG39 "logcontent", content
EG40 }
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>

## 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 EnsureLogging(@event["subject"].ToString());
EG17 }
EG18 }
EG19 return null;
EG20 }
EG21 private void EnsureLogging(string resource)
EG22 {
EG23 . . .
EG24 }
EG25 private async Task SendToAnomalyDetectionService(string uri)
EG26 {
EG27 var content = GetLogData(uri);
EG28 var scoreRequest = new
EG29 {
EG30 Inputs = new Dictionary<string, List<Dictionary<string, string>>()
EG31 {
EG32 {
EG33 "input1",
EG34 new List<Dictionary<string, string>>()
EG35 {
EG36 new Dictionary<string, string>()
EG37 {
EG38 {
EG39 "logcontent", content
EG40 }
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>

## 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 EnsureLogging(@event["subject"].ToString());
EG17 }
EG18 }
EG19 return null;
EG20 }
EG21 private void EnsureLogging(string resource)
EG22 {
EG23 . . .
EG24 }
EG25 private async Task SendToAnomalyDetectionService(string uri)
EG26 {
EG27 var content = GetLogData(uri);
EG28 var scoreRequest = new
EG29 {
EG30 Inputs = new Dictionary<string, List<Dictionary<string, string>>()
EG31 {
EG32 {
EG33 "input1",
EG34 new List<Dictionary<string, string>>()
EG35 {
EG36 new Dictionary<string, string>()
EG37 {
EG38 {
EG39 "logcontent", content
EG40 }
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	Answer Area
Process	public class IncludeEventId : <input type="text"/> code segment
Initialize	{
telemetry.Sequence	public void <input type="text"/> code segment ( <itelemetry td="" telemetry)<=""></itelemetry>
ITelemetryProcessor	{
ITelemetryInitializer	<input type="text"/> code segment .Properties["EventId"] =
telemetry.Context	<input type="text"/> code segment ;
EventGridController.EventId.Value	}
((EventTelemetry)telemetry).Properties["EventId"]	}

### Correct Answer:

Code segments	Answer Area
Process	public class IncludeEventId : <input type="text"/> ITelemetryInitializer
Initialize	{
telemetry.Sequence	public void <input type="text"/> Initialize ( <itelemetry td="" telemetry)<=""></itelemetry>
ITelemetryProcessor	{
ITelemetryInitializer	<input type="text"/> telemetry.Context .Properties["EventId"] =
telemetry.Context	<input type="text"/> ((EventTelemetry)telemetry).Properties["EventId"] ;
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>

**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 GetDRBlockBlob(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

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>

## 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 GetDRBBlob(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

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>

## 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 GetDRBBlob(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

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:

<https://docs.microsoft.com/en-us/azure/azure-monitor/app/sampling>

**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 -

VanArsdel, Ltd. is a global office supply company. The company is based in Canada and has retail store locations across the world. The company is developing several cloud-based solutions to support their stores, distributors, suppliers, and delivery services.

Current environment -

Corporate website -

The company provides a public website located at <http://www.vanarsdelltd.com>. The website consists of a React JavaScript user interface, HTML, CSS, image assets, and several APIs hosted in Azure Functions.

Retail Store Locations -

The company supports thousands of store locations globally. Store locations send data every hour to an Azure Blob storage account to support inventory, purchasing and delivery services. Each record includes a location identifier and sales transaction information.

Requirements -

The application components must meet the following requirements:

Corporate website -

Secure the website by using SSL.

Minimize costs for data storage and hosting.

Implement native GitHub workflows for continuous integration and continuous deployment (CI/CD).

Distribute the website content globally for local use.

Implement monitoring by using Application Insights and availability web tests including SSL certificate validity and custom header value verification.

The website must have 99.95 percent uptime.

Retail store locations -

Azure Functions must process data immediately when data is uploaded to Blob storage. Azure Functions must update Azure Cosmos DB by using native SQL language queries.

Audit store sale transaction information nightly to validate data, process sales financials, and reconcile inventory.

Delivery services -

Store service telemetry data in Azure Cosmos DB by using an Azure Function. Data must include an item id, the delivery vehicle license plate, vehicle package capacity, and current vehicle location coordinates.

Store delivery driver profile information in Azure Active Directory (Azure AD) by using an Azure Function called from the corporate website.

**Inventory services -**

The company has contracted a third-party to develop an API for inventory processing that requires access to a specific blob within the retail store storage account for three months to include read-only access to the data.

**Security -**

All Azure Functions must centralize management and distribution of configuration data for different environments and geographies, encrypted by using a company-provided RSA-HSM key.

Authentication and authorization must use Azure AD and services must use managed identities where possible.

**Issues -**

**Retail Store Locations -**

You must perform a point-in-time restoration of the retail store location data due to an unexpected and accidental deletion of data.

Azure Cosmos DB queries from the Azure Function exhibit high Request Unit (RU) usage and contain multiple, complex queries that exhibit high point read latency for large items as the function app is scaling.

**Question**

**HOTSPOT -**

You need to implement event routing for retail store location data.

Which configurations 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**

**Event data      Configuration**

**Source**

Azure Blob Storage
Azure Event Grid
Azure Service Bus
Azure Event Hub

**Receiver**

Azure Event Grid
Azure Event Hub
Azure Service Bus
Azure Blob Storage

**Handler**

Azure Function App
Azure Logic App
Azure Event Grid
Azure Blob Storage

## Answer Area

Event data	Configuration
Source	<div style="border: 1px solid #ccc; padding: 5px; width: fit-content;"><div style="background-color: #80e0a0; color: white; padding: 2px;">Azure Blob Storage</div>Azure Event Grid Azure Service Bus Azure Event Hub</div>
Correct Answer: Receiver	<div style="border: 1px solid #ccc; padding: 5px; width: fit-content;"><div style="background-color: #80e0a0; color: white; padding: 2px;">Azure Event Grid</div>Azure Event Hub Azure Service Bus Azure Blob Storage</div>
Handler	<div style="border: 1px solid #ccc; padding: 5px; width: fit-content;"><div style="background-color: #80e0a0; color: white; padding: 2px;">Azure Function App</div>Azure Logic App Azure Event Grid Azure Blob Storage</div>

Box 1: Azure Blob Storage -

Azure event publishers and event handlers are at the core of the Event Grid routing-service. Event Grid listens to Azure event publishers, such as Blob Storage, then reacts by routing specific events to Azure event handlers, such as WebHooks. You can easily control this entire process at a granular level through event subscriptions and event filters.

Box 2: Azure Event Grid -

Azure Event Grid is a highly scalable event-routing service that listens for specific system events, then reacts to them according to your precise specifications. In the past, event handling has relied largely on polling – a high latency, low-efficiency approach that can prove prohibitively expensive at scale.

Box 3: Azure Logic App -

Event Grid's supported event handlers currently include Event Hubs, WebHooks, Logic Apps, Azure Functions, Azure Automation and Microsoft Flow.

Reference:

<https://www.appliedi.net/blog/using-azure-event-grid-for-highly-scalable-event-routing>

**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.

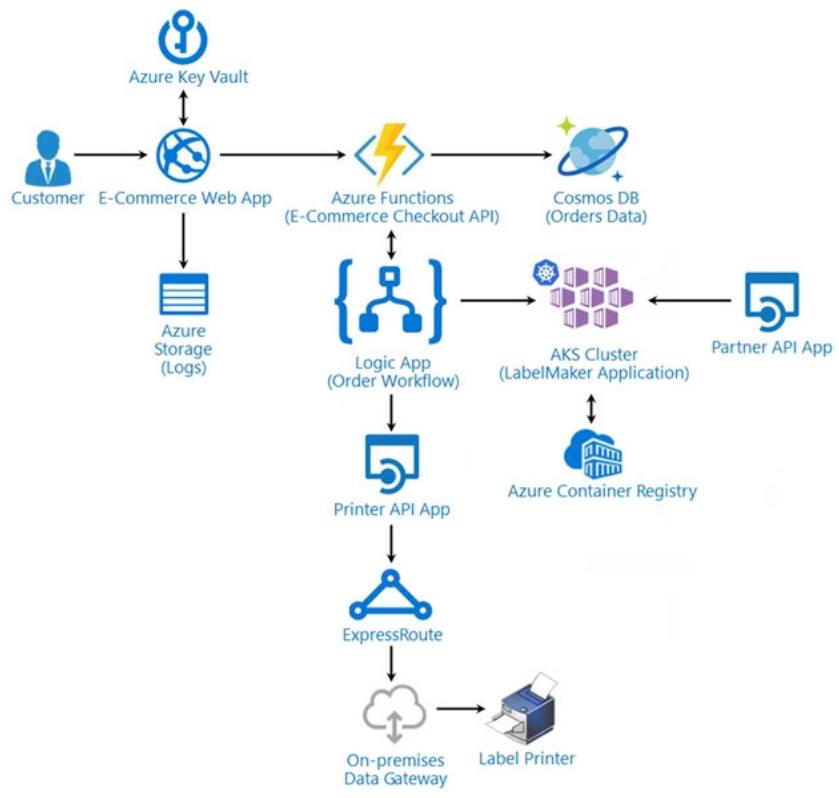
Requirements. LabelMaker app -

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]
30 }
31]
32 }
```

### 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   
 

Search to filter items by identifier

Status	Start time	Identifier	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>

## 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.

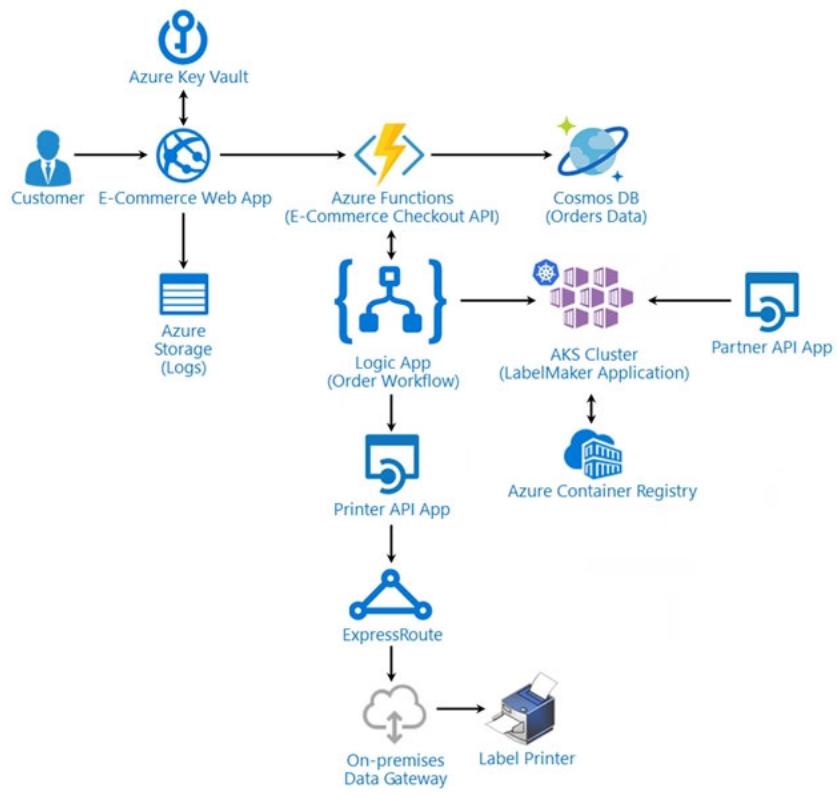
### Requirements. LabelMaker app -

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]
30 }
31]
32 }
```

### 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": "",

| |
|-------------|
| default |
| none |
| fixed |
| exponential |

 "interval": "",

| |
|-------|
| PT10S |
| PT30S |
| PT60S |
| PT1D |

 "count": ""

| |
|----|
| 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
 }
 }
}
```

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/>

**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 -

Wide World Importers is moving all their datacenters to Azure. The company has developed several applications and services to support supply chain operations and would like to leverage serverless computing where possible.

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 -

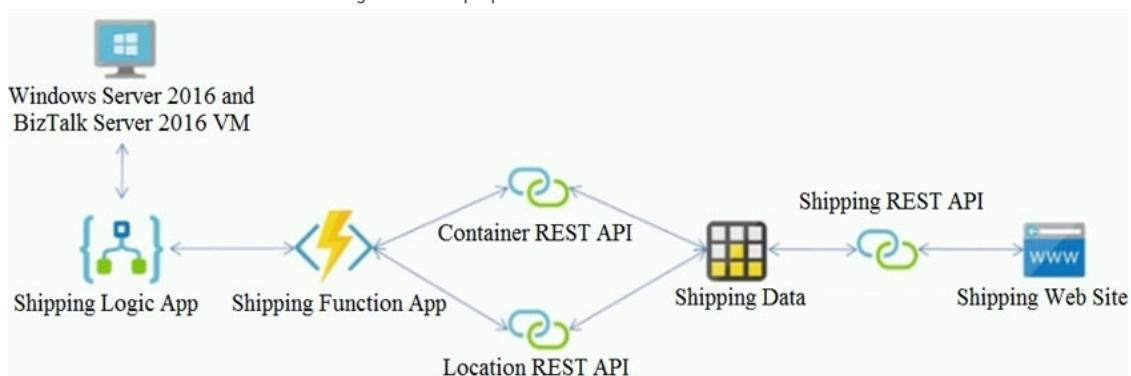
The application uses MongoDB JSON document storage database for all container and transport information.

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.

Correct Answer:

### Actions

### Answer Area

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>

## 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 -

Wide World Importers is moving all their datacenters to Azure. The company has developed several applications and services to support supply chain operations and would like to leverage serverless computing where possible.

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 -

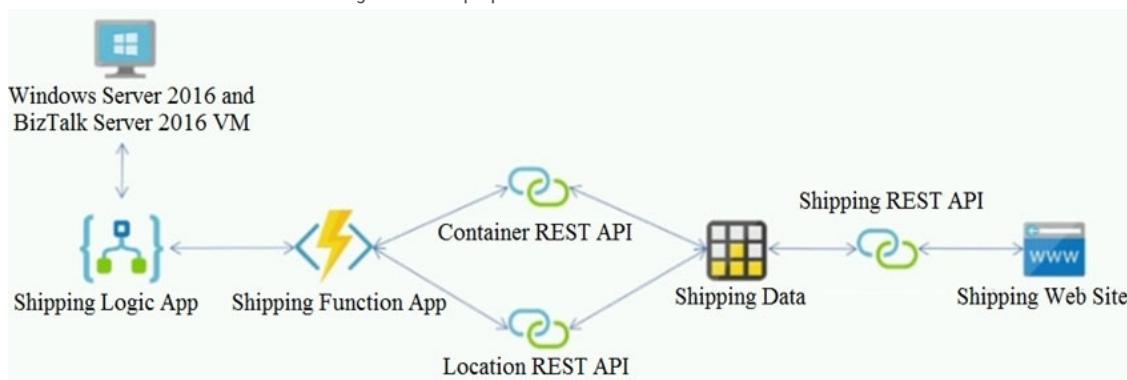
The application uses MongoDB JSON document storage database for all container and transport information.

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>

**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.

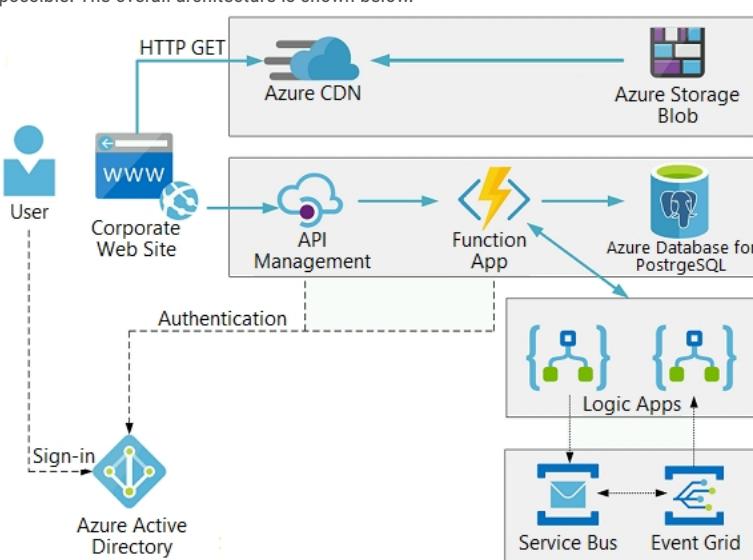
API Management and Azure Function App functions are used to process and store data in Azure Database for PostgreSQL. API Management is used to broker communications to the Azure Function app functions for Logic app integration. Logic apps are used to orchestrate the data processing while Service Bus 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,
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 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	<input type="button" value="▼"/>	<input type="button" value="▼"/>	create --source-resource-id \$topicid --name \$name --
	<input type="button" value="eventgrid"/>	<input type="button" value="event-subscription"/>	
	<input type="button" value="servicebus"/>	<input type="button" value="topic"/>	
		<input type="button" value="queue"/>	
endpoint-type	<input type="button" value="▼"/>	<input type="button" value="▼"/>	--endpoint \$endpoint
	<input type="button" value="webhook"/>	<input type="button" value="eventhub"/>	
	<input type="button" value="servicebusqueue"/>		

Correct Answer:

### Answer Area

az	<input type="button" value="▼"/>	<input type="button" value="▼"/>	create --source-resource-id \$topicid --name \$name --
	<input type="button" value="eventgrid"/>	<input type="button" value="event-subscription"/>	
	<input type="button" value="servicebus"/>	<input type="button" value="topic"/>	
		<input type="button" value="queue"/>	
endpoint-type	<input type="button" value="▼"/>	<input type="button" value="▼"/>	--endpoint \$endpoint
	<input type="button" value="webhook"/>	<input type="button" value="eventhub"/>	
	<input type="button" value="servicebusqueue"/>		

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)

## 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.

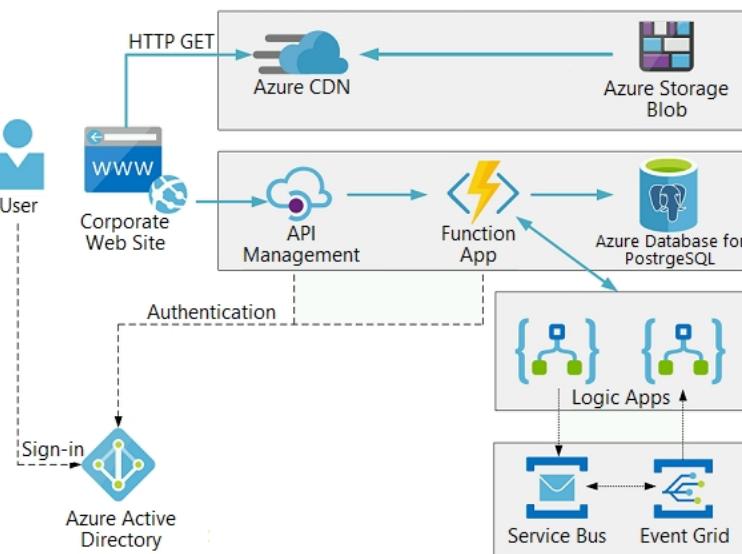
API Management and Azure Function App functions are used to process and store data in Azure Database for PostgreSQL. API Management is used to broker communications to the Azure Function app functions for Logic app integration. Logic apps are used to orchestrate the data processing while Service Bus 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

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>

**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>

## 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

Correct Answer:

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.

Box 2: eventType -

eventType is one of the registered event types for this event source.

Box 3: dataVersion -

dataVersion is the schema version of the data object. The publisher defines the schema version.

Scenario: 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.

The following example shows the properties that are used by all event publishers:

```
[
{
 "topic": string,
 "subject": string,
 "id": string,
 "eventType": string,
 "eventTime": string,
 "data": {
 object-unique-to-each-publisher
 },
 "dataVersion": string,
 "metadataVersion": string
}
]
```

Reference:

<https://docs.microsoft.com/en-us/azure/event-grid/event-schema>

## 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(" ")},
 {"DIAGNOSTICS_AZUREBLOBRETENTIONINDAYS", " "}

| |
|----|
| 15 |
| 30 |

 });
client.WebApps. (

| |
|--------------------------|
| UploadLoggingSettings |
| UpdateApplicationSetting |

 id.resourceGroup,
 id.name, appSettings);
```

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(" ")},
 {"DIAGNOSTICS_AZUREBLOBRETENTIONINDAYS", " "}

| |
|----|
| 15 |
| 30 |

 });
client.WebApps. (

| |
|--------------------------|
| UploadLoggingSettings |
| UpdateApplicationSetting |

 id.resourceGroup,
 id.name, appSettings);
```

Box 1: logdrop -

All log files should be saved to a container named logdrop.

Box 2: 15 -

Logs must remain in the container for 15 days.

Box 3: UpdateApplicationSettings

All Azure App Service Web Apps must write logs to Azure Blob storage.

Reference:

<https://blog.hompus.nl/2017/05/29/adding-application-logging-blob-to-a-azure-web-app-service-using-powershell/>

**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 BlobStoreAccountsSAS(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 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>

**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.

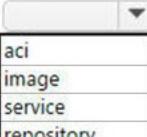
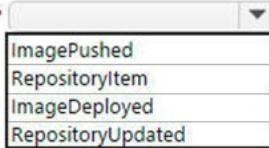
How should you complete the code segment? To answer, select the appropriate values in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

**Answer Area**

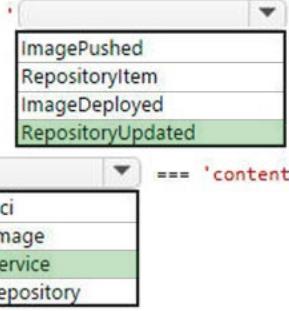
```
var event = getEvent();
if (event.eventType === 'ImagePushed'
 && event.data.target === 'contentanalysisservice'
 && event.data.topic.contains('contosoimages'))
{
 startValidationTesting();
}
```



## Answer Area

```
var event = getEvent();
if (event.eventType === 'RepositoryUpdated' && event.data.target.service === 'contentanalysiservice' && event.data.repository.contains('contosoimages')) {
 startValidationTesting();
}
```

Correct Answer:



The code snippet is a JavaScript function. It starts by getting an event. Then it checks if the event type is 'RepositoryUpdated'. If so, it checks if the target service is 'contentanalysiservice'. If both conditions are met, it checks if the repository contains 'contosoimages'. Finally, it calls a function named 'startValidationTesting()'. The code is surrounded by curly braces {}, indicating it's part of a larger function or class definition.

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>

## 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:

Consumption plan, Premium plan, and Dedicated (App Service) plan.

Incorrect Answers:

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>

**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.

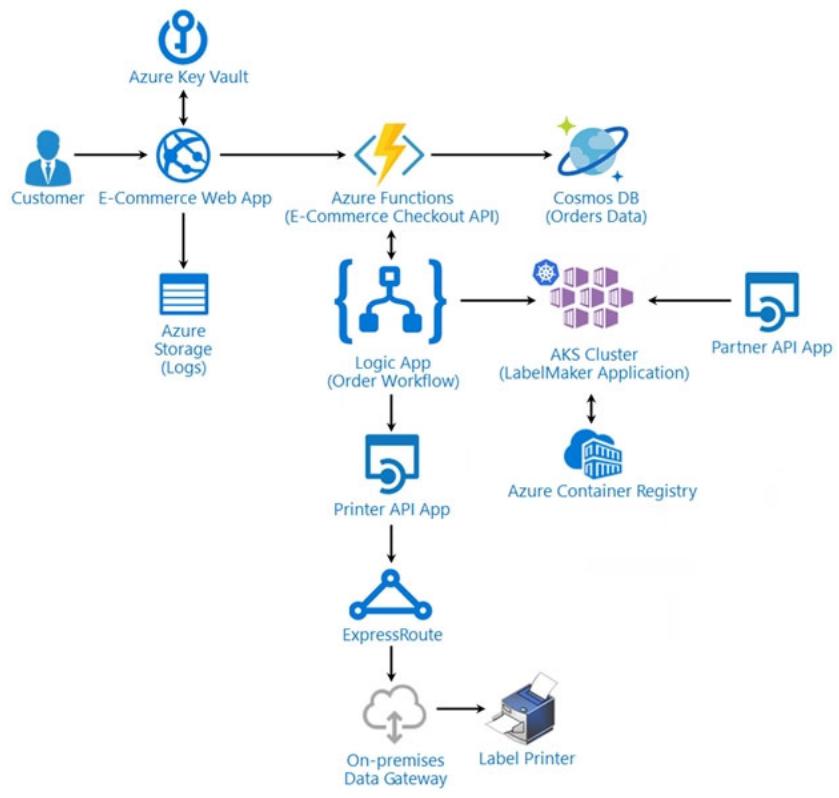
Requirements. LabelMaker app -

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've 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-word <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 form 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>

## 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.

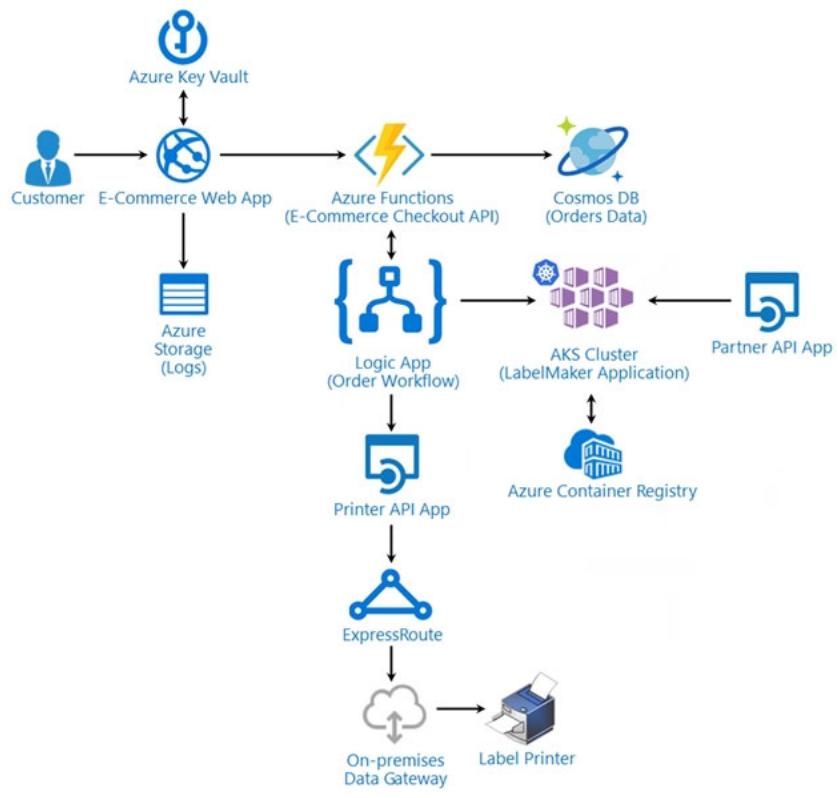
### Requirements. LabelMaker app -

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]
```

### 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 Service passes user claims to the app by using special request headers.

Reference:

<https://levelup.gitconnected.com/four-alternative-methods-to-get-user-identity-and-claims-in-a-net-azure-functions-app-df98c40424bb>

**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 -

VanArsdel, Ltd. is a global office supply company. The company is based in Canada and has retail store locations across the world. The company is developing several cloud-based solutions to support their stores, distributors, suppliers, and delivery services.

Current environment -

Corporate website -

The company provides a public website located at <http://www.vanarsdelltd.com>. The website consists of a React JavaScript user interface, HTML, CSS, image assets, and several APIs hosted in Azure Functions.

Retail Store Locations -

The company supports thousands of store locations globally. Store locations send data every hour to an Azure Blob storage account to support inventory, purchasing and delivery services. Each record includes a location identifier and sales transaction information.

Requirements -

The application components must meet the following requirements:

Corporate website -

Secure the website by using SSL.

Minimize costs for data storage and hosting.

Implement native GitHub workflows for continuous integration and continuous deployment (CI/CD).

Distribute the website content globally for local use.

Implement monitoring by using Application Insights and availability web tests including SSL certificate validity and custom header value verification.

The website must have 99.95 percent uptime.

Retail store locations -

Azure Functions must process data immediately when data is uploaded to Blob storage. Azure Functions must update Azure Cosmos DB by using native SQL language queries.

Audit store sale transaction information nightly to validate data, process sales financials, and reconcile inventory.

Delivery services -

Store service telemetry data in Azure Cosmos DB by using an Azure Function. Data must include an item id, the delivery vehicle license plate, vehicle package capacity, and current vehicle location coordinates.

Store delivery driver profile information in Azure Active Directory (Azure AD) by using an Azure Function called from the corporate website.

**Inventory services -**

The company has contracted a third-party to develop an API for inventory processing that requires access to a specific blob within the retail store storage account for three months to include read-only access to the data.

**Security -**

All Azure Functions must centralize management and distribution of configuration data for different environments and geographies, encrypted by using a company-provided RSA-HSM key.

Authentication and authorization must use Azure AD and services must use managed identities where possible.

**Issues -**

**Retail Store Locations -**

You must perform a point-in-time restoration of the retail store location data due to an unexpected and accidental deletion of data.

Azure Cosmos DB queries from the Azure Function exhibit high Request Unit (RU) usage and contain multiple, complex queries that exhibit high point read latency for large items as the function app is scaling.

**Question**

**HOTSPOT -**

You need to implement the retail store location Azure Function.

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

NOTE: Each correct selection is worth one point.

Hot Area:

## **Answer Area**

<b>Configuration</b>	<b>Value</b>
Binding	<input type="button" value="▼"/>
	<input type="button" value="Blob storage"/> <input type="button" value="Azure Cosmos DB"/> <input type="button" value="Event Grid"/> <input type="button" value="HTTP"/>
Binding Direction	<input type="button" value="▼"/>
	<input type="button" value="Input"/> <input type="button" value="Output"/>
Trigger	<input type="button" value="▼"/>
	<input type="button" value="Blob storage"/> <input type="button" value="Azure Cosmos DB"/> <input type="button" value="Event Grid"/> <input type="button" value="HTTP"/>

## Answer Area

Configuration	Value
Binding	<div style="border: 1px solid black; padding: 5px; width: fit-content;"><div style="border-bottom: 1px solid black; padding-bottom: 2px;">Blob storage</div><div>Azure Cosmos DB</div><div>Event Grid</div><div style="background-color: #669966; color: white; padding: 2px;">HTTP</div></div>
Correct Answer: Binding Direction	<div style="border: 1px solid black; padding: 5px; width: fit-content;"><div style="border-bottom: 1px solid black; padding-bottom: 2px;">Input</div><div style="background-color: #669966; color: white; padding: 2px;">Output</div></div>
Trigger	<div style="border: 1px solid black; padding: 5px; width: fit-content;"><div style="border-bottom: 1px solid black; padding-bottom: 2px;">Blob storage</div><div>Azure Cosmos DB</div><div>Event Grid</div><div style="background-color: #669966; color: white; padding: 2px;">HTTP</div></div>

Scenario: Retail store locations: Azure Functions must process data immediately when data is uploaded to Blob storage.

Box 1: HTTP -

Binding configuration example: [https://<storage\\_account\\_name>.blob.core.windows.net](https://<storage_account_name>.blob.core.windows.net)

Box 2: Input -

Read blob storage data in a function: Input binding

Box 3: Blob storage -

The Blob storage trigger starts a function when a new or updated blob is detected.

Azure Functions integrates with Azure Storage via triggers and bindings. Integrating with Blob storage allows you to build functions that react to changes in blob data as well as read and write values.

Reference:

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

## 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 -

VanArsdel, Ltd. is a global office supply company. The company is based in Canada and has retail store locations across the world. The company is developing several cloud-based solutions to support their stores, distributors, suppliers, and delivery services.

Current environment -

Corporate website -

The company provides a public website located at <http://www.vanarsdelltd.com>. The website consists of a React JavaScript user interface, HTML, CSS, image assets, and several APIs hosted in Azure Functions.

Retail Store Locations -

The company supports thousands of store locations globally. Store locations send data every hour to an Azure Blob storage account to support inventory, purchasing and delivery services. Each record includes a location identifier and sales transaction information.

Requirements -

The application components must meet the following requirements:

Corporate website -

Secure the website by using SSL.

Minimize costs for data storage and hosting.

Implement native GitHub workflows for continuous integration and continuous deployment (CI/CD).

Distribute the website content globally for local use.

Implement monitoring by using Application Insights and availability web tests including SSL certificate validity and custom header value verification.

The website must have 99.95 percent uptime.

Retail store locations -

Azure Functions must process data immediately when data is uploaded to Blob storage. Azure Functions must update Azure Cosmos DB by using native SQL language queries.

Audit store sale transaction information nightly to validate data, process sales financials, and reconcile inventory.

Delivery services -

Store service telemetry data in Azure Cosmos DB by using an Azure Function. Data must include an item id, the delivery vehicle license plate, vehicle package capacity, and current vehicle location coordinates.

Store delivery driver profile information in Azure Active Directory (Azure AD) by using an Azure Function called from the corporate website.

Inventory services -

The company has contracted a third-party to develop an API for inventory processing that requires access to a specific blob within the retail store

storage account for three months to include read-only access to the data.

Security -

All Azure Functions must centralize management and distribution of configuration data for different environments and geographies, encrypted by using a company-provided RSA-HSM key.

Authentication and authorization must use Azure AD and services must use managed identities where possible.

Issues -

Retail Store Locations -

You must perform a point-in-time restoration of the retail store location data due to an unexpected and accidental deletion of data.

Azure Cosmos DB queries from the Azure Function exhibit high Request Unit (RU) usage and contain multiple, complex queries that exhibit high point read latency for large items as the function app is scaling.

**Question**

HOTSPOT -

You need to implement the corporate website.

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

NOTE: Each correct selection is worth one point.

Hot Area:

## Answer Area

### Azure Configuration

Plan

Free
Standard
Premium
Isolated

Service

App Service Web App
App Service Static Web App
Azure Function App
Azure Blob Storage

## Answer Area

Azure	Configuration				
Plan	<table border="1"> <tr><td>Free</td></tr> <tr><td>Standard</td></tr> <tr><td>Premium</td></tr> <tr><td>Isolated</td></tr> </table>	Free	Standard	Premium	Isolated
Free					
Standard					
Premium					
Isolated					
Correct Answer:	Standard				
Service	<table border="1"> <tr><td>App Service Web App</td></tr> <tr><td>App Service Static Web App</td></tr> <tr><td>Azure Function App</td></tr> <tr><td>Azure Blob Storage</td></tr> </table>	App Service Web App	App Service Static Web App	Azure Function App	Azure Blob Storage
App Service Web App					
App Service Static Web App					
Azure Function App					
Azure Blob Storage					
Box 1: Standard -					

Below is a high-level comparison of the features as per the pricing tier for the App Service Plan.

SKUs	FREE	SHARED	BASIC	STANDARD	PREMIUM	ISOLATED*
<b>Limits</b>						
Apps	10	100	Unlimited	Unlimited	Unlimited	Unlimited
Disk space	1 GB	1 GB	10 GB	50 GB	250 GB	
Max instances			Up to 3	Up to 10	Up to 20	
SLA			99.95%	99.95%	99.95%	
<b>App Deployment</b>						
Continuous Deployment	Available	Available	Available	Available	Available	Available
Deployment Slots			Available	Available	Available	
<b>Development Tools</b>						
Clone App				Available	Available	
Site Extensions	Available	Available	Available	Available	Available	Available
Testing in Production			Available	Available	Available	

Note: Corporate website -

The company provides a public website located at <http://www.vanarsdeltd.com>. The website consists of a React JavaScript user interface, HTML, CSS, image assets, and several APIs hosted in Azure Functions.

Corporate website requirements:

- ⇒ Secure the website by using SSL.
- ⇒ Minimize costs for data storage and hosting.
- ⇒ Implement native GitHub workflows for continuous integration and continuous deployment (CI/CD).
- ⇒ Distribute the website content globally for local use.
- ⇒ Implement monitoring by using Application Insights and availability web tests including SSL certificate validity and custom header value verification.
- ⇒ The website must have 99.95 percent uptime.

Box 2: App Service Web App -

A Web App is a web application that is hosted in an App Service. The App Service is the managed service in Azure that enables you to deploy a web application and make it available to your customers on the Internet in a very short amount of time.

Incorrect:

A Static Web Application is any web application that can be delivered directly to an end user's browser without any server-side alteration of the HTML, CSS, or

JavaScript content.

Reference:

<https://azure-training.com/2018/12/27/understanding-app-services-app-service-plan-and-ase/> <https://docs.microsoft.com/en-us/azure/app-service/overview>

**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 -

VanArsdel, Ltd. is a global office supply company. The company is based in Canada and has retail store locations across the world. The company is developing several cloud-based solutions to support their stores, distributors, suppliers, and delivery services.

Current environment -

Corporate website -

The company provides a public website located at <http://www.vanarsdelltd.com>. The website consists of a React JavaScript user interface, HTML, CSS, image assets, and several APIs hosted in Azure Functions.

Retail Store Locations -

The company supports thousands of store locations globally. Store locations send data every hour to an Azure Blob storage account to support inventory, purchasing and delivery services. Each record includes a location identifier and sales transaction information.

Requirements -

The application components must meet the following requirements:

Corporate website -

Secure the website by using SSL.

Minimize costs for data storage and hosting.

Implement native GitHub workflows for continuous integration and continuous deployment (CI/CD).

Distribute the website content globally for local use.

Implement monitoring by using Application Insights and availability web tests including SSL certificate validity and custom header value verification.

The website must have 99.95 percent uptime.

Retail store locations -

Azure Functions must process data immediately when data is uploaded to Blob storage. Azure Functions must update Azure Cosmos DB by using native SQL language queries.

Audit store sale transaction information nightly to validate data, process sales financials, and reconcile inventory.

Delivery services -

Store service telemetry data in Azure Cosmos DB by using an Azure Function. Data must include an item id, the delivery vehicle license plate, vehicle package capacity, and current vehicle location coordinates.

Store delivery driver profile information in Azure Active Directory (Azure AD) by using an Azure Function called from the corporate website.

**Inventory services -**

The company has contracted a third-party to develop an API for inventory processing that requires access to a specific blob within the retail store storage account for three months to include read-only access to the data.

**Security -**

All Azure Functions must centralize management and distribution of configuration data for different environments and geographies, encrypted by using a company-provided RSA-HSM key.

Authentication and authorization must use Azure AD and services must use managed identities where possible.

**Issues -**

**Retail Store Locations -**

You must perform a point-in-time restoration of the retail store location data due to an unexpected and accidental deletion of data.

Azure Cosmos DB queries from the Azure Function exhibit high Request Unit (RU) usage and contain multiple, complex queries that exhibit high point read latency for large items as the function app is scaling.

**Question**

You need to implement a solution to resolve the retail store location data issue.

Which three Azure Blob features should you enable? Each correct answer presents part of the solution.

NOTE: Each correct selection is worth one point.

- A. Soft delete
- B. Change feed
- C. Snapshots
- D. Versioning
- E. Object replication
- F. Immutability

**Correct Answer:** ABD

Scenario: You must perform a point-in-time restoration of the retail store location data due to an unexpected and accidental deletion of data.

Before you enable and configure point-in-time restore, enable its prerequisites for the storage account: soft delete, change feed, and blob versioning.

Reference:

<https://docs.microsoft.com/en-us/azure/storage/blobs/point-in-time-restore-manage>

**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>

## 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}")]
 [CosmosDBTrigger("content", "userContent")]
 [Table("content", "userContent", "{name}")]

 [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)
 {
 ...
 }
}
```

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.

The attribute's constructor takes the path to the blob and a FileAccess parameter indicating read or write, as shown in the following example:

```
[FunctionName("ResizeImage")]
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>

## 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/>

**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.cpandl.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.

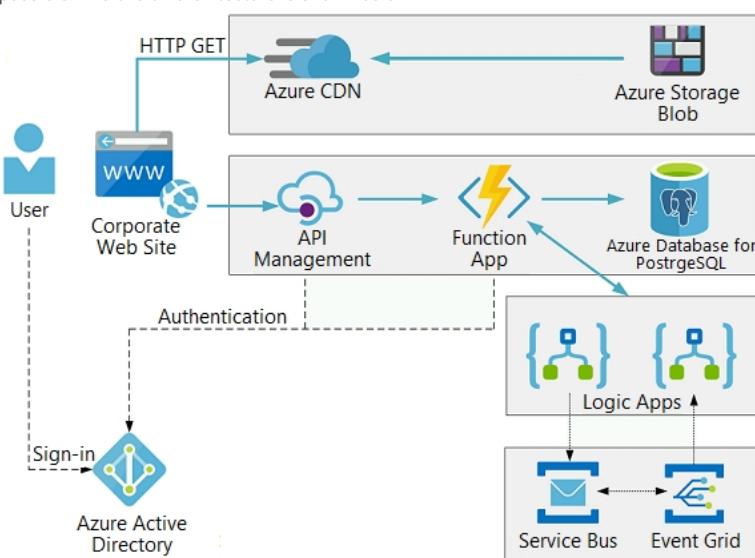
API Management and Azure Function App functions are used to process and store data in Azure Database for PostgreSQL. API Management is used to broker communications to the Azure Function app functions for Logic app integration. Logic apps are used to orchestrate the data processing while Service Bus 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 the Account Kind, Replication, and Access 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
	<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) ⓘ	<input type="radio"/> Cool <input type="radio"/> 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

[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

Account Kind: StorageV2 (general-purpose v2)

Scenario: Azure Storage blob will be used (refer to the exhibit). Data storage costs must be minimized.

General-purpose v2 accounts: Basic storage account type for blobs, files, queues, and tables. Recommended for most scenarios using Azure Storage.

Incorrect Answers:

⇒ BlockBlobStorage 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.

Incorrect Answers:

Geo-zone-redundant storage (GZRS), but it would be more costly.

Access tier: Cool -

Data storage costs must be minimized.

Note: Azure storage offers different access tiers, which allow you to store blob object data in the most cost-effective manner. The available access tiers include:

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.

Reference:

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