

**Exam : AZ-203**

**Title : Developing Solutions for Microsoft Azure**

**Vendor : Microsoft**

**Version : V14.95**

**NO.1** You need to deploy a new version of the LabelMaker application.

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.

NOTE: Each correct selection is worth one point.

**Actions****Answer Area**

Restart the cluster.

Create an alias of the image with the a new build number.

Build a new application image by using msbuild.

Create an alias of the image with the fully qualified path to the registry.



Build a new application image by using dockerfile.

Download the image to your local computer.

Log in to the registry and push image.



**Answer:**

**Actions**

Restart the cluster.

Create an alias of the image with the a new build number.

Build a new application image by using msbuild.

Create an alias of the image with the fully qualified path to the registry.

Build a new application image by using dockerfile.

Download the image to your local computer.

Log in to the registry and push image.

**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.



Explanation

**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.

References:

<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>

Case Study:

Overview

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.

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 Structured Query Language (SQL).
- \* Changes to the Order data must reflect immediately across all partitions. All reads to the Order data must fetch the most recent writes.

##### Security

You have the following security requirements:

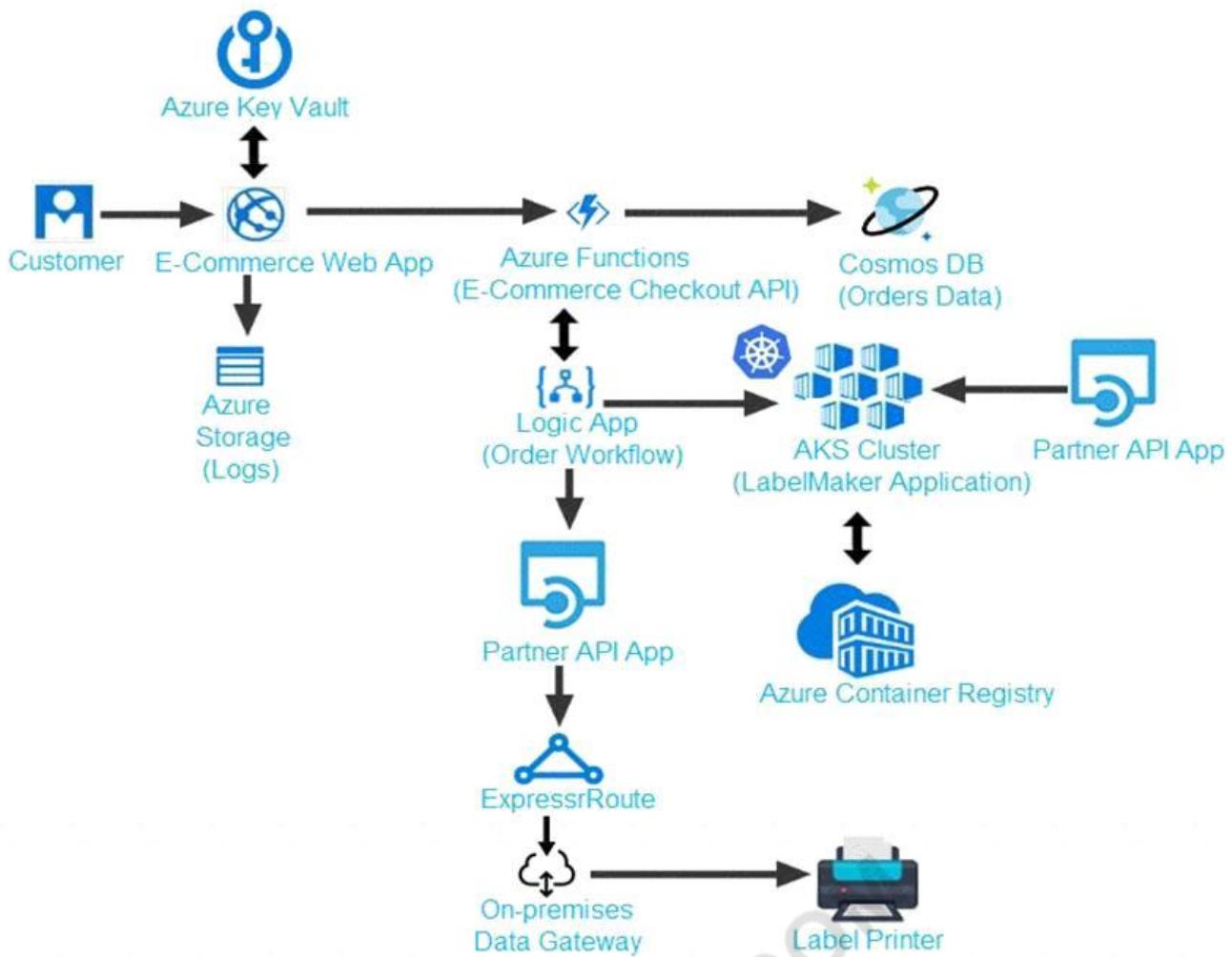
- \* Users of Coho Winery applications must be able to provide access to documents, resources, and applications to external partners.
- \* External partners must use their own credentials and authenticate with their organization's identity management solution.
- \* External partner logins must be audited 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.

##### 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 fall 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.

The JSON file contains a representation of the data for an order that includes a single item.

```
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 }
38 ],
39 "address" : {
40   "state" : "NY",
41   "country" : "Manhattan",
42   "city" : "NY"
43 }
44 }
```

**NO.2** 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 SequenceNumber property.
- B.** Assign the value of the hazard message SequenceNumber property to the DeliveryCount property.
- C.** Assign the value of the hazard message MessageId property to the DeliveryCount property.
- D.** Assign the value of the hazard message MessageId property to the CorrelationId property.
- E.** Assign the value of the hazard message SessionId property to the ReplyToSessionId property.

**Answer:** A,B

**NO.3** 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.

```
"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": {  
        "type": "▼"  
        5  
        10  
        60  
    }  
}
```

**Answer:**

```
"print_label": {  
    "type": "Http",  
    "inputs": {  
        "method" : "POST",  
        "uri": "https://www.cohowinery.com/printer/printlabel",  
        "retryPolicy": {  
            "type": "fixed",  
            "interval": "PT10S",  
            "count": 5  
        }  
    }  
}
```

Explanation

```

"print_label": {
  "type": "Http",
  "inputs": {
    "method": "POST",
    "uri": "https://www.cohowinery.com/printer/printlabel",
    "retryPolicy": {
      "type": "fixed", ▾
      "interval": "PT10S", ▾
      "count": 5
    }
  }
}

```

The image shows three dropdown menus from a configuration interface:

- Box 1 (retryPolicy.type):** A dropdown menu with options: default, none, fixed (selected), and exponential.
- Box 2 (retryPolicy.interval):** A dropdown menu with options: PT10S (selected), PT30S, PT60S, and PT1D.
- Box 3 (retryPolicy.count):** A dropdown menu with options: 5 (selected), 10, and 60.

Box 1: Fixed

To specify that the action or trigger waits the specified interval before sending the next request, set the

<retry-policy-type> to fixed.

Box 2: PT10S

Box 3: 5

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.

**NO.4** You need to ensure that you can deploy the LabelMaker application.

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

NOTE: Each correct selection is worth one point.

```

az group create - -name CohoWIneryLabelMaker - -location eastus
group
aks
acr

```

```

az create - -resource-group CohoWIneryLabelMaker - -name
group
aks
acr

```

```

LabelMakerCluster - -node-count 5 - -enable-addons monitoring
LabelMakerCluster
http_application_routing

```

**Answer:**

```

az group create - -name CohoWIneryLabelMaker - -location eastus
group
aks
acr

```

```

az create - -resource-group CohoWIneryLabelMaker - -name
group
aks
acr

```

```

LabelMakerCluster - -node-count 5 - -enable-addons monitoring
LabelMakerCluster
http_application_routing

```

**Explanation**

```

az group create - -name CohoWIneryLabelMaker - -location eastus
group
aks
acr

```

```

az create - -resource-group CohoWIneryLabelMaker - -name
group
aks
acr

```

```

LabelMakerCluster - -node-count 5 - -enable-addons monitoring
LabelMakerCluster
http_application_routing

```

**Box 1: group**

Create a resource group with the az group create command. An Azure resource group is a logical group in which Azure resources are deployed and managed.

The following example creates a resource group named myResourceGroup in the westeurope location.

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

**Box 2: CohoWIneryLabelMaker**

Use the resource group named, which is used in the second command.

**Box 3: aks**

The command az aks create, is used to create a new managed Kubernetes cluster.

**Box 4: monitoring****Scenario: 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.

**NO.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 need to meet the LabelMaker application security requirement.

Solution: Create a Microsoft Azure Active Directory service principal and assign it to the Azure Kubernetes Service (AKS) cluster.

Does the solution meet the goal?

**A.** Yes

**B.** No

**Answer:** A

**NO.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.

Determine whether the solution meets the stated goals.

You need to meet the LabelMaket application

Solution: Create a conditional access policy and assign it to the Azure Kubernetes service cluster.

Does the solution meet the goal?

**A.** Yes

**B.** No

**Answer:** B

Explanation

Scenario: The LabelMaker applications must be secured by using an AAD account that has full access to all namespaces of the Azure Kubernetes Service (AKS) cluster.

Before an Azure Active Directory account can be used with the AKS cluster, a role binding or cluster role binding needs to be created.

References:

<https://docs.microsoft.com/en-us/azure/aks/aad-integration>

**NO.7** You need to meet the LabelMaker security requirement.

What should you do?

- A.** Create a conditional access policy and assign it to the Azure Kubernetes Service cluster
- B.** Place the Azure Active Directory account into an Azure AD group. Create a ClusterRoleBinding and assign it to the group.
- C.** Create a Microsoft Azure Active Directory service principal and assign it to the Azure Kubernetes Service (AKS) duster.
- D.** Create a RoleBinding and assign it to the Azure AD account.

**Answer:** B

Explanation

Scenario: The LabelMaker applications must be secured by using an AAD account that has full access to all namespaces of the Azure Kubernetes Service (AKS) cluster.

Permissions can be granted within a namespace with a RoleBinding, or cluster-wide with a ClusterRoleBinding.

References:

<https://kubernetes.io/docs/reference/access-authn-authz/rbac/>

**NO.8** You need to access user claims in the e-commerce web app\* What should you do first?

- A.** Update the e-commerce web app to read the HTTP request header values.
- B.** Assign the Contributor RBAC role to the e-commerce web app by using the Resource Manager create role assignment API.
- C.** Write custom code to make a Microsoft Graph API call from the e-commerce web app.
- D.** Using the Azure CU enable Cross-origin resource sharing (CORS) from the e-commerce checkout API to the e-commerce web app

**Answer:** C

Explanation

If you want more information about the user, you'll need to use the Azure AD Graph API.

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

**NO.9** You need to implement the e-commerce checkout API.

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

NOTE: Each correct selection is worth one point.

- A.** In the Azure Function App, enable Manger Service Identity (MSI).
- B.** Set the function template's Mode property to Webhook and the Webhook type property to Generic JSON
- C.** Set the function template's Mode property to Webhook and the Webhook type property to GitHub.
- D.** Create an Azure Function using the HTTP POST function template.
- E.** In the Azure Function App, enable Cross-Origin Resource Sharing (CORS) with all origins permitted.
- F.** Create an Azure Function using the Generic webhook function template.

**Answer:** A B D

Explanation

Scenario: E-commerce application sign-ins must be secured by using Azure App Service authentication and Azure Active Directory (AAD).

References:

<https://docs.microsoft.com/en-us/azure/app-service/overview-managed-identity>

**NO.10** 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.

<b>Setting</b>	<b>Value</b>
Consistency Level	Strong Bounded-staleness Session Eventual
API	SQL MongoDB Graph Table

**Answer:**

<b>Setting</b>	<b>Value</b>
Consistency Level	Strong Bounded-staleness Session Eventual
API	SQL MongoDB Graph Table

**Explanation**

Setting	Value
Consistency Level	<p>Strong</p> <p>Bounded-staleness</p> <p>Session</p> <p>Eventual</p>
API	<p>SQL</p> <p>MongoDB</p> <p>Graph</p> <p>Table</p>

**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).

**NO.11** You need to meet the security requirements for external partners.

Which Azure Active Directory features should you use?

To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

**Answer Area**

<b>Requirement</b>	<b>Option</b>
Authentication	B2C B2B Self-service signup Organizational Units (OU)
Login Auditing	Access review Risky sign-ins report Identity Protection Privileged Identity Management

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

<b>Requirement</b>	<b>Option</b>
Authentication	B2C B2B Self-service signup Organizational Units (OU)
Login Auditing	Access review Risky sign-ins report Identity Protection Privileged Identity Management

**Explanation**

Requirement	Option
Authentication	<ul style="list-style-type: none"> <li>B2C</li> <li>B2B</li> <li>Self-service signup</li> <li>Organizational Units (OU)</li> </ul>
Login Auditing	<ul style="list-style-type: none"> <li>Access review</li> <li>Risky sign-ins report</li> <li>Identity Protection</li> <li>Privileged Identity Management</li> </ul>

#### Box 1: B2B

Scenario: External partners must use their own credentials and authenticate with their organization's identity management solution.

Azure Active Directory (Azure AD) business-to-business (B2B) collaboration lets you securely share your company's applications and services with guest users from any other organization, while maintaining control over your own corporate data. Work safely and securely with external partners, large or small, even if they don't have Azure AD or an IT department. A simple invitation and redemption process lets partners use their own credentials to access your company's resources. Developers can use Azure AD business-to-business APIs to customize the invitation process or write applications like self-service sign-up portals.

#### Box 2: Access Review

Scenario: External partner logins must be audited monthly for application use by a user account administrator to maintain company compliance.

Azure Active Directory (Azure AD) Access Reviews enable organizations to efficiently manage group memberships, access to enterprise applications, and role assignments.

Administrators can use Azure Active Directory (Azure AD) to create an access review for group members or users assigned to an application. Azure AD automatically sends reviewers an email that prompts them to review access.

#### References:

<https://docs.microsoft.com/en-us/azure/active-directory/b2b/what-is-b2b>

**NO.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. Determine whether the solution meets the stated goals.

You need to meet the LabelMaker application security requirement.

Solution: Place the Azure Active Directory account into an Azure AD group. Create a ClusterRoleBinding and assign it to the group.

Does the solution meet the goal?

- A. Yes
- B. No

**Answer:** A

Explanation

Scenario: The LabelMaker applications must be secured by using an AAD account that has full access to all namespaces of the Azure Kubernetes Service (AKS) cluster.

Permissions can be granted within a namespace with a RoleBinding, or cluster-wide with a ClusterRoleBinding.

References:

<https://kubernetes.io/docs/reference/access-authn-authz/rbac/>

**NO.13** You need to troubleshoot the order workflow.

What should you do? Each correct answer presents part of the solution.

NOTE: Each correct selection is worth one point.

- A. Review the run history.
- B. Review the trigger history.
- C. Review the API connections.
- D. Review the activity log.

**Answer:** B D

Explanation

Scenario: The order workflow fails to run upon initial deployment to Azure.

Deployment errors arise from conditions that occur during the deployment process. They appear in the activity log.

References:

<https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-audit>

**NO.14** You need to provision and deploy the order workflow.

Which three components should you include? Each correct answer presents part of the solution.

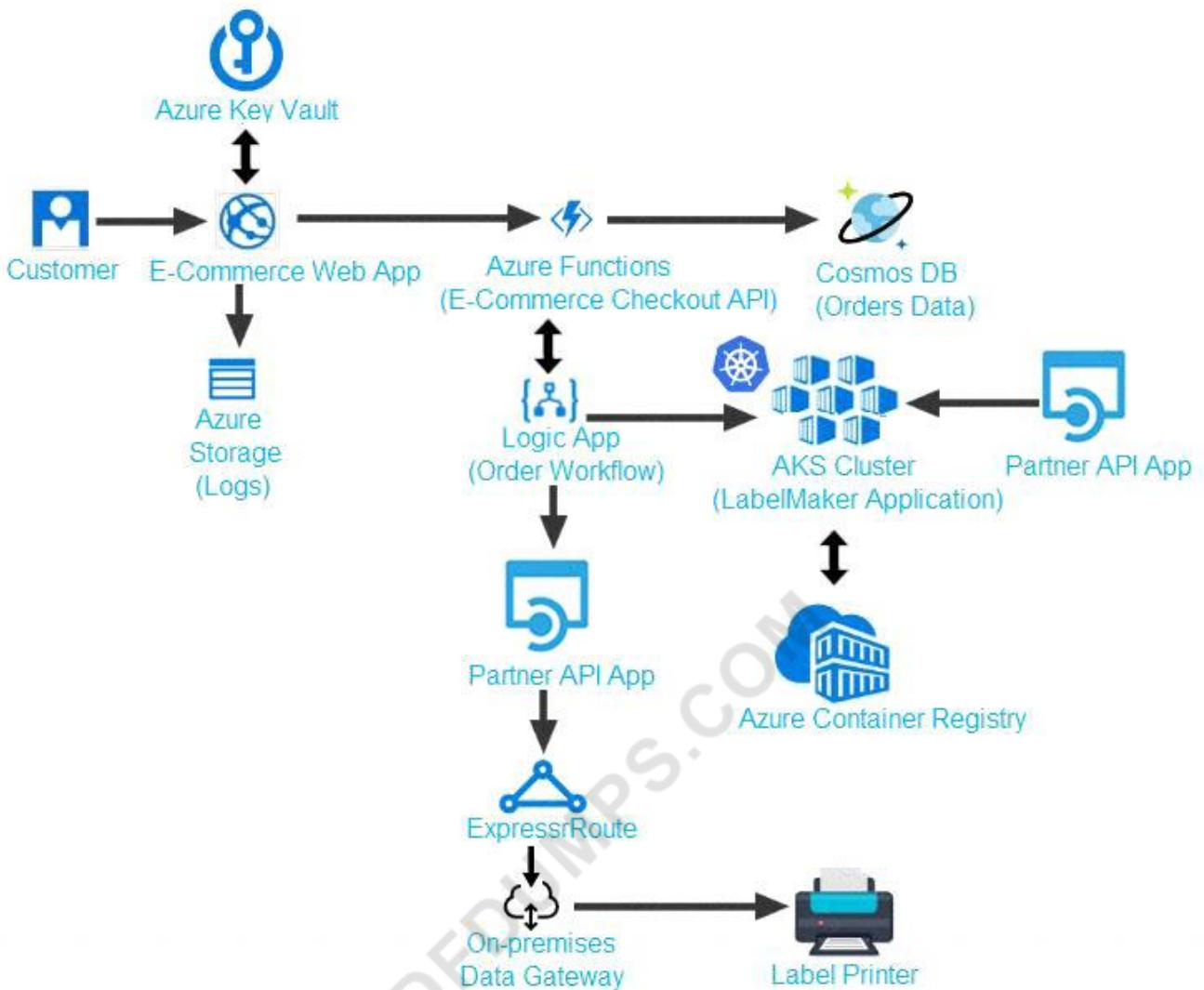
NOTE: Each correct selection is worth one point

- A. Connections
- B. On-premises Data Gateway
- C. Workflow definition
- D. Resources
- E. Functions

**Answer:** B C E

Explanation

Scenario: The order workflow fails to run upon initial deployment to Azure.



**NO.15** Note: In this section you will see one or more sets of questions with the same scenario and problem. Each question presents a unique solution to the problem, and you must determine whether the solution meets the stated goals. More than one solution might solve the problem. It is also possible that none of the solutions solve the problem.

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.

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 need to meet the LabelMaker application security requirement.

Solution: Create a RoleBinding and assign it to the Azure AD account.

Does the solution meet the goal?

- A. Yes
- B. No

**Answer:** B

Explanation

Scenario: The LabelMaker applications must be secured by using an AAD account that has full access to all namespaces of the Azure Kubernetes Service (AKS) cluster.

Permissions can be granted within a namespace with a RoleBinding, or cluster-wide with a ClusterRoleBinding.

References:

<https://kubernetes.io/docs/reference/access-authn-authz/rbac/>

**NO.16** You need to meet the security requirements for the E-Commerce Web App. Which two steps should you take?

Each correct answer presents part of the solution. NOTE: Each correct selection is worth one point.

- A. Create an Azure AD service principal.
- B. Enable Managed Service Identity (MSI) on the E-Commerce Web App.
- C. Add a policy to the Azure Key Vault to grant access to the E-Commerce Web App.
- D. Update the E-Commerce Web App with the service principal's client secret.

**Answer:** B C

Explanation

Scenario: E-commerce application sign-ins must be secured by using Azure App Service authentication and Azure Active Directory (AAD).

A managed identity from Azure Active Directory allows your app to easily access other AAD-protected resources such as Azure Key Vault. T References:

<https://docs.microsoft.com/en-us/azure/app-service/overview-managed-identity>

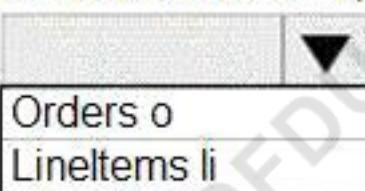
**NO.17** You need to retrieve all order line items sorted alphabetically by the city.

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

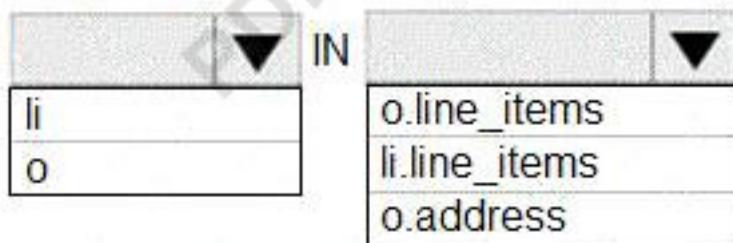
NOTE: Each correct selection is worth one point.

SELECT li.id AS lineitemid, li.price

FROM

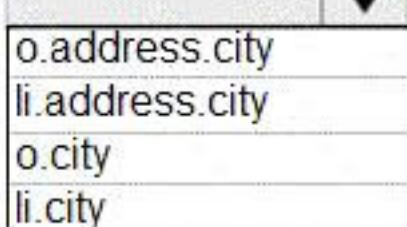


JOIN



ORDER BY

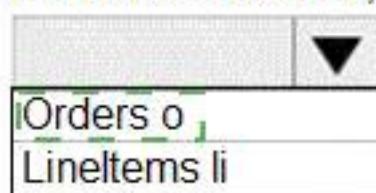
ASC



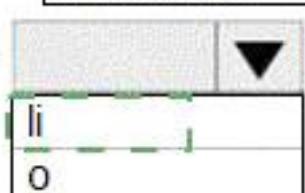
**Answer:**

SELECT li.id AS lineitemid, li.price

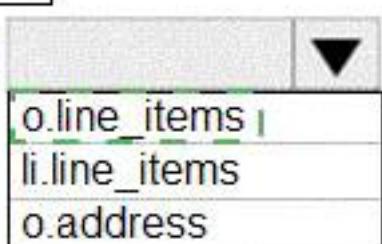
FROM



JOIN

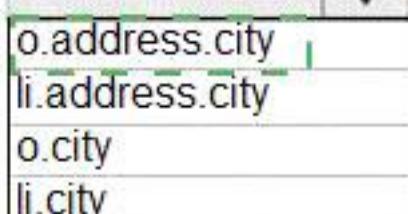


IN



ORDER BY

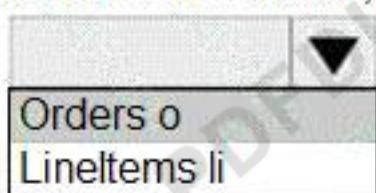
ASC



Explanation

SELECT li.id AS lineitemid, li.price

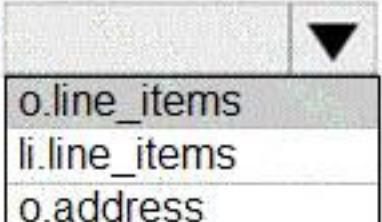
FROM



JOIN

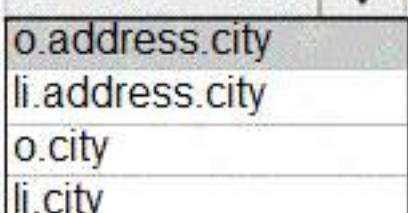


IN



ORDER BY

ASC



Scenario:

Order data is stored as nonrelational JSON and must be queried using Structured Query Language (SQL). The Order data is stored in a Cosmos database.

## Topic 1, Litware Inc

### Case Study:

#### Overview

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.

#### Overview

#### 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, 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.

### Requirements

#### Receipt processing

Concurrent processing of a receipt must be prevented.

#### Logging

Azure Application Insights is used for telemetry and logging in both the processor and the web application.

The processor also has Trace Writer logging enabled. Application Insights must always contain all log messages.

#### 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

Users' 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 Least Privilege Principal.

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

#### 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().CreateIfNotExistAsync();
PC28             return cloudBlobClient.GetRootContainerReference();
PC29         }
PC30         private static async Task<StorageCredentials> GetCredentials()
PC31         {
PC32             . . .
PC33         }
PC34         private static async Task<List<IListFileItem>> ListFiles()
PC35         {
PC36             . . .
PC37         }
PC38         private KeyVaultClient _keyVaultClient = new KeyVaultClient(" . . .");
PC39     }

```

#### Database.cs

```

DB01 public class Database
DB02 {
DB03     private string ConnectionString =
DB04
DB05     public async Task<object> LoadUserDetails(string userId)
DB06     {
DB07
DB08         return await policy.ExecuteAsync (async () =>
DB09         {
DB10             using (var connection = new SqlConnection (ConnectionString))
DB11             {
DB12                 await connection.OpenAsync ();
DB13                 using (var command = new SqlCommand(" ", connection))
DB14                     using (var reader = command.ExecuteReader())
DB15                         {
DB16                             -
DB17                         }
DB18                     }
DB19                 });
DB20             }
DB21         }

```

### ReceiptUploader.cs

```

RU01 public class ReceiptUploader
RU02 {
RU03     public async Task UploadFile(string file, byte[ ] binary)
RU04     {
RU05         var httpClient = new HttpClient();
RU06         var response = await httpClient.PutAsync( "...", new ByteArrayContent(binary));
RU07         while (ShouldRetry (response))
RU08         {
RU09             response = await httpClient.PutAsync ( "...", new ByteArrayContent(binary));
RU10         }
RU11     }
RU12     private bool ShouldRetry (HttpResponseMessage 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

```

**NO.18** You need to construct the link to the summary report for the email that is sent to users. What should you do?

- A. Create a SharedAccessBlobPolicy and add it to the containers SharedAccessPolicies. Call GetSharedAccessSignature on the blob and use the resulting link.
- B. Create a SharedAccessBlobPolicy and set the expiry time to two weeks from today. Call GetSharedAccessSignature on the blob and use the resulting link.
- C. Create a SharedAccessAccountPolicy and call GetsharedAccessSignature on storage account and use the resulting link.
- D. Create a SharedAccessBlobPolicy and set the expiry time to two weeks from today. Call GetSharedAccessSignature on the container and use the resulting link.

**Answer:** D

Explanation

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.

Create a stored access policy to manage signatures on a container's resources, and then generate the shared access signature on the container, setting the constraints directly on the signature.

Code example: Add a method that generates the shared access signature for the container and returns the signature URI.

```

static string GetContainerSasUri(CloudBlobContainer container)
{
    //Set the expiry time and permissions for the container.
    //In this case no start time is specified, so the shared access signature becomes valid immediately.
    SharedAccessBlobPolicy sasConstraints = new SharedAccessBlobPolicy();
    sasConstraints.SharedAccessExpiryTime = DateTimeOffset.UtcNow.AddHours(24);
    sasConstraints.Permissions = SharedAccessBlobPermissions.List |
        SharedAccessBlobPermissions.Write;
    //Generate the shared access signature on the container, setting the constraints directly on the

```

signature.

```
string sasContainerToken = container.GetSharedAccessSignature(sasConstraints);
//Return the URI string for the container, including the SAS token.
return container.Uri + sasContainerToken;
}
```

References:

<https://docs.microsoft.com/en-us/azure/storage/blobs/storage-dotnet-shared-access-signature-part-2>

**NO.19** You need to ensure the security policies are met. What code do you add at line CS07?

- A.** -PermissionsToKeys wrapkey, unwrapkey, get
- B.** -PermissionsToKeys create, encrypt, decrypt
- C.** -PermissionsToCertificates wrapkey, unwrapkey, get
- D.** -PermissionsToCertificates create, encrypt, decrypt

**Answer:** D

Explanation

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-AzureRmKeyVaultAccessPolicy 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 References:

<https://docs.microsoft.com/en-us/powershell/module/azurerm.keyvault/set-azurermkeyvaultaccesspolicy>

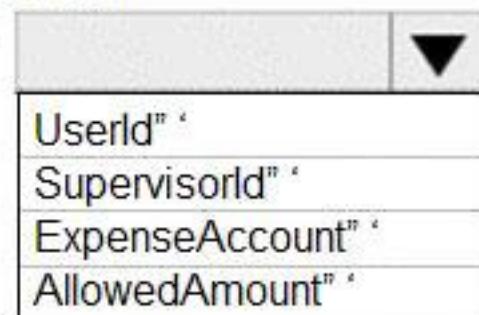
**NO.20** You need to ensure that security requirements are met.

How should you complete the code segment?

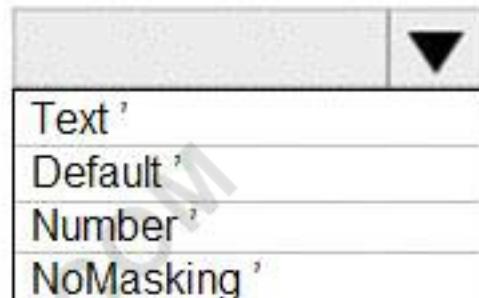
To answer, select the appropriate option in the answer area.

NOTE: Each correct selection is worth one point.

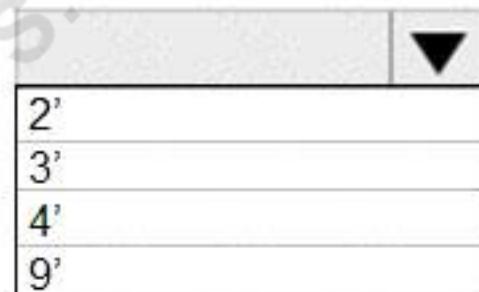
```
New-AzureRmSqlDatabaseDataMaskingRule  
    -ServerName          "datastore"  
    -DatabaseName        "expense"  
    -ResourceGroupName   "production"  
    -SchemaName          "dbo"  
    -TableName            "Users"  
    -ColumnName
```



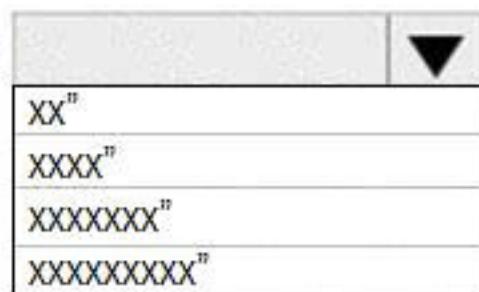
```
-MaskingFunction
```



```
-SuffixSize
```



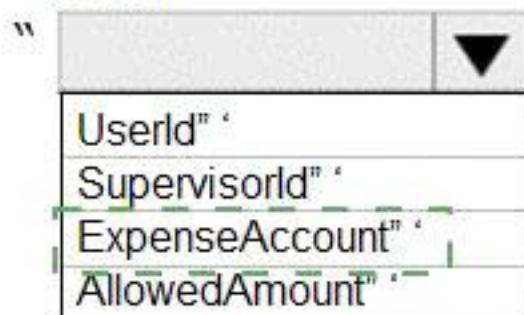
```
-ReplacementString
```



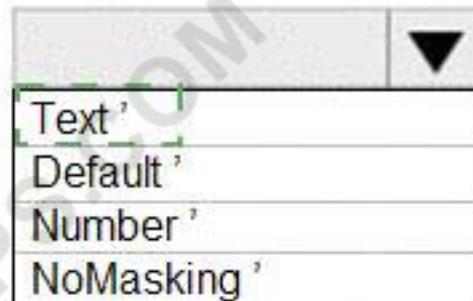
**Answer:**

```
New-AzureRmSqlDatabaseDataMaskingRule `
```

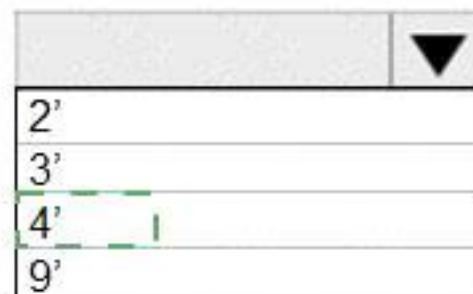
- ServerName "datastore"
- DatabaseName "expense"
- ResourceGroupName "production"
- SchemaName "dbo"
- TableName "Users"
- ColumnName



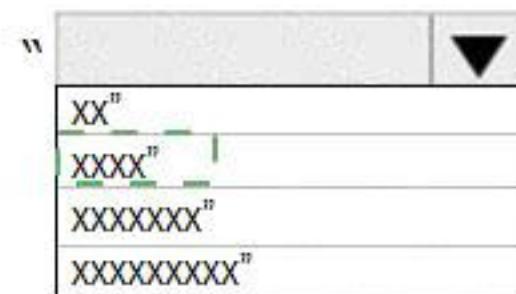
-MaskingFunction



-SuffixSize



-ReplacementString

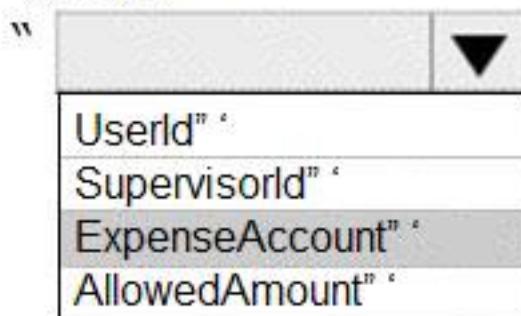


Explanation

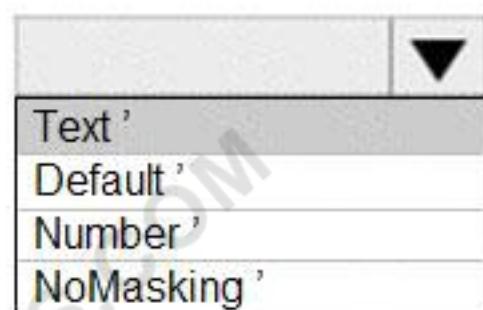
```

-DatabaseName      "expense" '
-ResourceGroupName "production" '
-SchemaName        "dbo" '
-TableName          "Users" '
-ColumnName

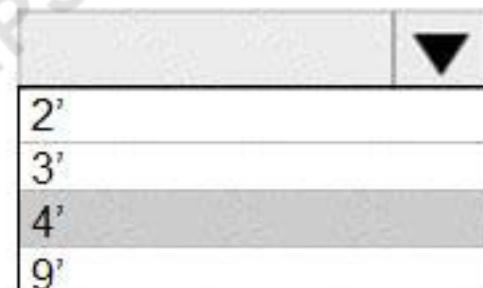
```



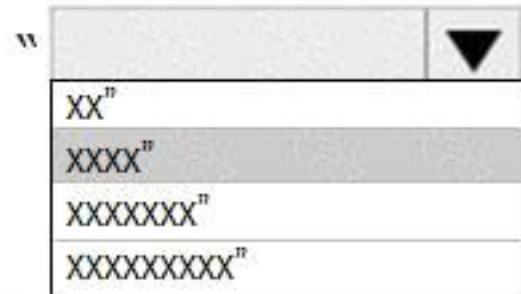
-MaskingFunction



-SuffixSize



-ReplacementString



#### Box 1: ExpenseAccount

Scenario: 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.

#### Box 2: Text

If MaskingFunction has a value of Number or Text, you can specify the NumberFrom and NumberTo parameters, for number masking, or the PrefixSize, ReplacementString, and SuffixSize for text masking.

Box 3: 4

-SuffixSize specifies the number of characters at the end of the text that are not masked. Specify this parameter only if you specify a value of Text for the MaskingFunction parameter.

Scenario: Format is 1234-1234-1235

Box 4: xxxx"

Scenario: Format is 1234-1234-1235

References:

<https://docs.microsoft.com/en-us/powershell/module/azurerm.sql/new-azurermsqldatabasedatamaskingrule?view>

**NO.21** You need to resolve the capacity issue.

What should you do?

- A.** Move the Azure Function to a dedicated App Service Plan.
- B.** Convert the trigger on the Azure Function to a File Trigger.
- C.** Ensure that the consumption plan is configured correctly to allowSw for scaling.
- D.** Update the loop starting on line PC09 to process items in parallel.

**Answer:** D

Explanation

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    }

```

References:

<https://stackoverflow.com/questions/37576685/using-async-await-with-a-foreach-loop>

**NO.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 question, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

You need to ensure that the SecurityPin security requirements are met.

Solution: Configure the web application to connect to the database using the WebAppIdentity security principal. Using the Azure Portal, add Data Masking to the SecurityPin column and exclude the WebAppIdentity service principal.

Does the solution meet the goal?

- A.** Yes
- B.** No

**Answer: A**

**NO.23** You need to ensure receipt processing occurs correctly.

What should you do?

- A. Use blob metadata to prevent concurrency problems.
- B. Use blob SnapshotTime to prevent concurrency problems.
- C. Use blob leases to prevent concurrency problems.
- D. Use blob properties to prevent concurrency problems.

**Answer: B**

Explanation

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.

References:

<https://docs.microsoft.com/en-us/rest/api/storageservices/creating-a-snapshot-of-a-blob>

**NO.24** You need to ensure that security requirements are met.

What value should be used for the ConnectionString field on line DB03 in the Database class? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

**"Data Source=datasource.database.windows.net;Initial Catalog=expense;**

Integrated Security = SSPI	;
Trusted_Connection = False	
Network Library = DBNSSOCN	
MultipleActiveResultSets = True	

;	;"
Encrypt = True	
Integrated Security = True	
Failover Partner = False	
Named Pipes = True	

**Answer:**

```
"Data Source=datastore.database.windows.net;Initial Catalog=expense;
```

Integrated Security = SSPI	;
Trusted_Connection = False	
Network Library = DBNSSOCN	
MultipleActiveResultSets = True	
Encrypt = True	;
Integrated Security = True	
Failover Partner = False	
Named Pipes = True	

### Explanation

```
"Data Source=datastore.database.windows.net;Initial Catalog=expense;
```

Integrated Security = SSPI	;
Trusted_Connection = False	
Network Library = DBNSSOCN	
MultipleActiveResultSets = True	
Encrypt = True	;
Integrated Security = True	
Failover Partner = False	
Named Pipes = True	

#### Box 1: Integrated Security=SSPI

Integrated security: For all data source types, connect using the current user account.

For SqlClient you can use Integrated Security=true; or Integrated Security=SSPI; Scenario: All access to Azure Storage and Azure SQL database must use the application's Managed Service Identity (MSI)

Box 2: Encrypt = True Scenario: All data must be protected in transit.

#### References:

<https://docs.microsoft.com/en-us/dotnet/framework/data/adonet/connection-string-syntax>

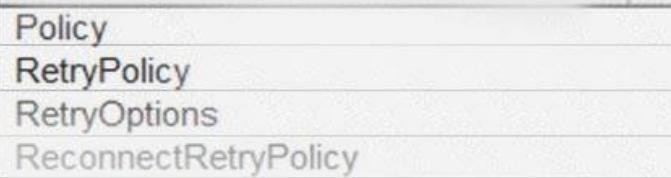
**NO.25** You need to configure retries in the LoadUserDetails function in the Database class without impacting user experience.

What code should you insert on line DB07?

To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

```
var policy=
```



```
.Handle<Exception>()
```

```
.Retry(3);
.CircuitBreaker(3, TimeSpan.fromMilliseconds(100));
.WaitAndRetryAsync(3, i => TimeSpan.FromMilliseconds(100));
.WaitAndRetryAsync(3,i => TimeSpan.FromMilliseconds(100" Math.Pow(2,i-1)));
```

### Answer:

```
var policy=
```

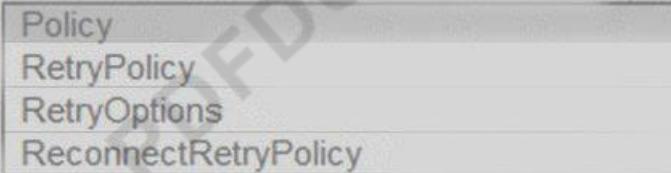


```
.Handle<Exception>()
```

```
.Retry(3);
.CircuitBreaker(3, TimeSpan.fromMilliseconds(100));
.WaitAndRetryAsync(3, i => TimeSpan.FromMilliseconds(100));
.WaitAndRetryAsync(3,i => TimeSpan.FromMilliseconds(100" Math.Pow(2,i-1)));
```

### Explanation

```
var policy=
```



```
.Handle<Exception>()
```

```
.Retry(3);
.CircuitBreaker(3, TimeSpan.fromMilliseconds(100));
.WaitAndRetryAsync(3, i => TimeSpan.FromMilliseconds(100));
.WaitAndRetryAsync(3,i => TimeSpan.FromMilliseconds(100" Math.Pow(2,i-1)));
```

Box 1: Policy

RetryPolicy retry = Policy

```
Handle<HttpRequestException>()
Retry(3);
```

The above example will create a retry policy which will retry up to three times if an action fails with an exception handled by the Policy.

Box 2: `WaitAndRetryAsync(3,i => TimeSpan.FromMilliseconds(100 * Math.Pow(2,i-1)))`; A common retry strategy is exponential backoff: this allows for retries to be made initially quickly, but then at progressively longer intervals, to avoid hitting a subsystem with repeated frequent calls if the subsystem may be struggling.

Example:

Policy

```
Handle<SomeExceptionType>()
WaitAndRetry(3, retryAttempt =>
TimeSpan.FromSeconds(Math.Pow(2, retryAttempt))
);
```

References:

<https://github.com/App-vNext/Polly/wiki/Retry>

**NO.26** Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.

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

You need to ensure that the SecurityPin security requirements are met.

Solution: Enable Always Encrypted for the SecurityPin column using a certificate based on a trusted certificate authority. Update the Getting Started document with instructions to ensure that the certificate is installed on user machines.

Does the solution meet the goal?

A. Yes

B. No

**Answer:** B

Explanation

Enable Always Encrypted is correct, but only the WebAppIdentity service principal should be given access to the certificate.

Scenario: Users' 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.

**NO.27** You need to ensure that security policies are met.

What code should you add at line PC26?

To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

```
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));
```

### Answer:

```
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));
```

### Explanation

## 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; References:  
<https://github.com/Azure/azure-storage-net/blob/master/Samples/GettingStarted/EncryptionSamples/KeyRotatio>

**NO.28** Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.

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

You need to ensure that the SecurityPin security requirements are met.

Solution: Using the Azure Portal, add Data Masking to the SecurityPin column, and exclude the dbo user. Add a SQL security policy with a filter predicate based on the user identity.

Does the solution meet the goal?

A. Yes

B. No

**Answer:** B

Explanation

Instead of DataMasking, enable Always Encrypted for the SecurityPin column.

Scenario: Users' 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.

**NO.29** 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.

Values	Answer Area
true	var copyOptions = new CopyOptions {};
false	var context= new [ ](source,destination)=>Task.FromResult(true);
SingleTransferContext	context. [ ](source, destination) => Task.FromResult(true);
DirectoryTransferContext	await TransferManager.CopyAsync(blob, GetDRBlob(blob), isServiceCopy:[ ]
ShouldTransferCallbackAsync	, context:context, options: copyOptions); copyOptions, context);
ShouldOverwriteCallbackAsync	

### Answer:

Values	Answer Area
true	var copyOptions = new CopyOptions {};
false	var context= new DirectoryTransferContext [ ](source,destination)=>Task.FromResult(true);
SingleTransferContext	context. ShouldTransferCallbackAsync [ ](source, destination) => Task.FromResult(true);
DirectoryTransferContext	await TransferManager.CopyAsync(blob, GetDRBlob(blob), isServiceCopy:[ ]
ShouldTransferCallbackAsync	, context:context, options: copyOptions); copyOptions, context);
ShouldOverwriteCallbackAsync	

### Explanation

```
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); copyOptions, context);
```

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 References:

<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.directorytransfercon>

**NO.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 need to ensure that the SecurityPin security requirements are met.

Solution: Enable Always Encrypted for the SecurityPin column using a certificate contained in Azure Key Vault and grant the WebAppIdentity service principal access to the certificate.

Does the solution meet the goal?

**A.** Yes

**B.** No

**Answer:** A

Explanation

Scenario: Users' 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.

**NO.31** You need to add code at line PC32 in Processing.es 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.

#### Code segments

```
MSITokenProvider("...", null)
tp.GetAccessTokenAsync("...")
AzureServiceTokenProvider()
StringTokenProvider("storage", "msi")
tp.GetAuthenticationHeaderAsync(CancellationToken.None)
```

#### Answer Area

```
var tp = new _____
var t = new TokenCredential(await _____
return new StorageCredential_____(t);
```

**Answer:**

#### Code segments

```
MSITokenProvider("...", null)
tp.GetAccessTokenAsync("...")
AzureServiceTokenProvider()
StringTokenProvider("storage", "msi")
tp.GetAuthenticationHeaderAsync(CancellationToken.None)
```

#### Answer Area

```
var tp = AzureServiceTokenProvider()
var t = new TokenCredential(await tp.GetAccessTokenAsync("..."))
return new StorageCredential_____(t);
```

Explanation

```

var tp=new AzureServiceTokenProvider()
var t=new TokenCredential(await tp.GetAccessTokenAsync("..."))
return new StorageCredentials(t);

```

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/");
}

```

References:

<https://joonasw.net/view/azure-ad-authentication-with-azure-storage-and-managed-service-identity>

**NO.32** You need to resolve the log capacity issue.

What should you do?

- A. Implement Application Insights Sampling.
- B. Change the minimum log level in the host.json file for the function.
- C. Create an Application Insights Telemetry Filter.
- D. Set a LogCategoryFilter during startup.

**Answer:** A

Explanation

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.

References:

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

---

Topic 2, Proseware, Inc

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 Policy Lib 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 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  }

```

**LoginEvents.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 }
```

**NO.33** You need to meet the scaling requirement for Policy Service.

What should you store in Azure Redis Cache?

- A. HttpContext.Items
- B. ViewState
- C. Session state
- D. TempData

**Answer:** C

Explanation

Azure Cache for Redis provides a session state provider that you can use to store your session state in-memory with Azure Cache for Redis instead of a SQL Server database.

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:

Ensure that scaling actions do not disrupt application usage.

References:

<https://docs.microsoft.com/en-us/azure/azure-cache-for-redis/cache-aspnet-session-state-provider>

**NO.34** Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution. After you answer a question in this section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

You need to ensure that authentication events are triggered and processed according to the policy.

Solution: Ensure that signout events have a subject prefix. Create an Azure Event Grid subscription that uses the subjectBeginsWith filter.

- A. Yes
- B. No

**Answer:** B

**NO.35** 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.

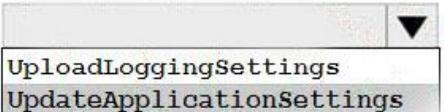
```
var client = new WebSiteManagementClient(. . .);
var id = ParseResourceId(resource);
var appSettings = new StringDictionary(name: "properties",
    properties: new Dictionary<string, string> {
        {"DIAGNOSTICS_AZUREBLOBCONTAINERSASURL", BlobStoreAccountsSAS("logs") },
        {"DIAGNOSTICS_AZUREBLOBRETENTIONDAYS", "15" }
    });
client.WebApps.UploadLoggingSettings(id.resourceGroup,
    id.name, appSettings);
```

**Answer:**

```
var client = new WebSiteManagementClient(. . .);
var id = ParseResourceId(resource);
var appSettings = new StringDictionary(name: "properties",
    properties: new Dictionary<string, string> {
        {"DIAGNOSTICS_AZUREBLOBCONTAINERSASURL", BlobStoreAccountsSAS("logdrop") },
        {"DIAGNOSTICS_AZUREBLOBRETENTIONDAYS", "15" }
    });
client.WebApps.UploadLoggingSettings(id.resourceGroup,
    id.name, appSettings);
```

**Explanation**

```

var client = new WebSiteManagementClient(. . .);
var id = ParseResourceId(resource);
var appSettings = new StringDictionary(name: "properties",
    properties: new Dictionary<string, string> {
        {"DIAGNOSTICS_AZUREBLOBCONTAINERSASURL", BlobStoreAccountsSAS(""
            
        {"DIAGNOSTICS_AZUREBLOBRETENTIONDAYS", ""
            
        });
        client.WebApps. 
        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.

References:

<https://blog.hompus.nl/2017/05/29/adding-application-logging-blob-to-a-azure-web-app-service-using-powershe>

**NO.36** You need to insert code at line LE03 of LoginEvent.cs to ensure that all authentication events are procesed 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.

```
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:**

```
public string { get; set; }
```

id
eventType
dataVersion
metadataVersion

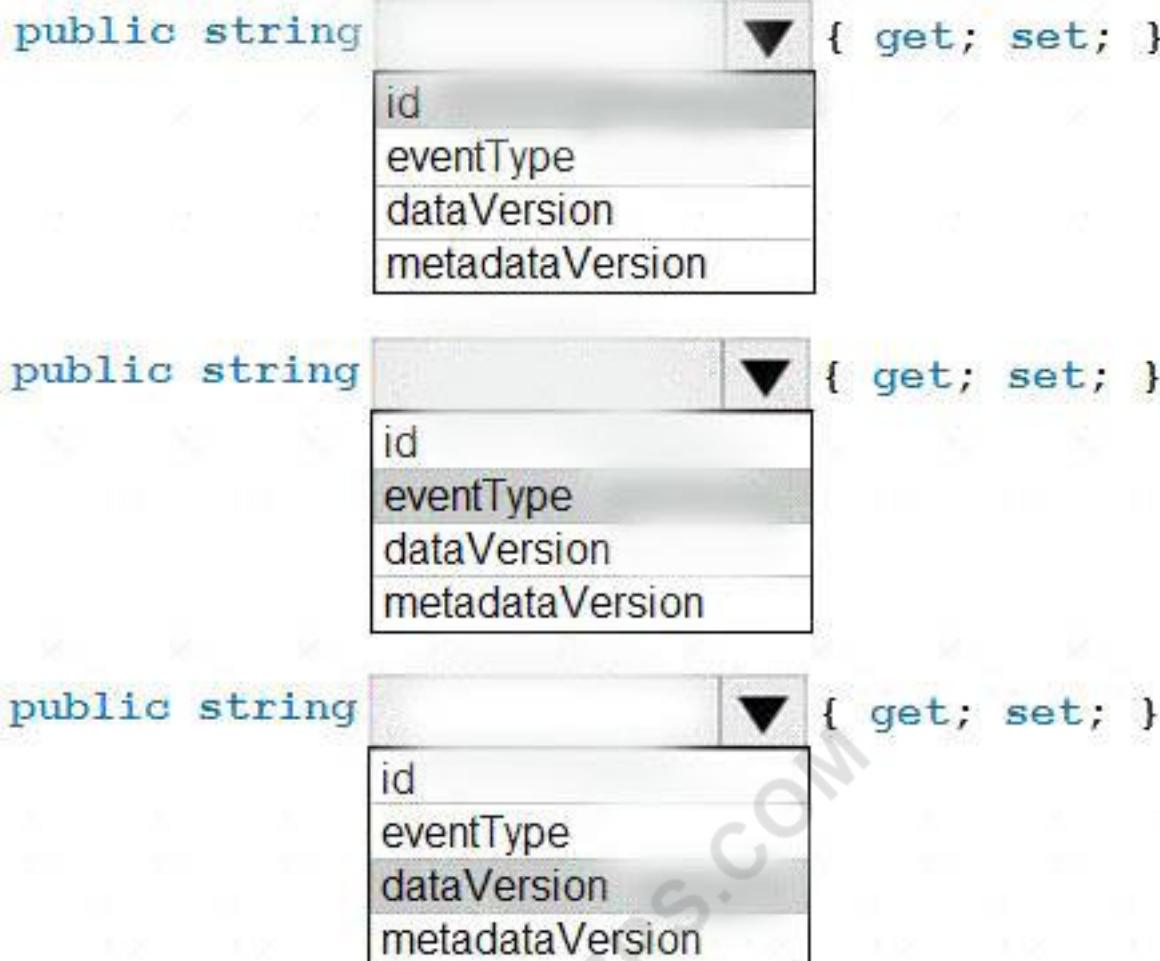
```
public string { get; set; }
```

id
eventType
dataVersion
metadataVersion

```
public string { get; set; }
```

id
eventType
dataVersion
metadataVersion

Explanation



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
}
]

```

References:

<https://docs.microsoft.com/en-us/azure/event-grid/event-schema>

### NO.37 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. Ensure that the Azure Function is using an App Service plan.
- B. Set Always On to false
- C. Ensure that the Azure Function is set to use a consumption plan.
- D. Set Always On to true.

**Answer:** A D

Explanation

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.

References:

<https://github.com/Azure/Azure-Functions/wiki/Enable-Always-On-when-running-on-dedicated-App-Service-Plan>

**NO.38** Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution. After you answer a question in this section, you will NOT be able to return to it As a result these questions will not appear in the review screen.

You need to ensure that authentication events are triggered and processed according to the policy.

Solution: Create a new Azure Event Grid topic and add a subscription for the events.

Does the solution meet the goal?

- A. Yes
- B. No

**Answer:** B

Explanation

Use a separate Azure Event Grid topics and subscriptions for sign-in and sign-out events.

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.

**NO.39** Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution. After you answer a question in this section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

You need to ensure that authentication events are triggered and processed according to the policy.

Solution: Create a new Azure Event Grid topic and add a subscription for the events.

Does the solution meet the goal?

**A.** Yes

**B.** No

**Answer:** B

Explanation

Instead ensure that signout events have a subject prefix. Create an Azure Event Grid subscription that uses the subjectBeginsWith filter.

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.

**NO.40** 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.

Code segments	Answer Area
topic	if (
status	@event[ "data" ] [ "status" ].ToString() == "Succeeded"
eventType	&&
Succeeded	@event[ "data" ] [ "operationName" ].ToString() == "Microsoft.Web/sites/write"
operationName	)
resourceProvider	

**Answer:**

Code segments	Answer Area
topic	if (
status	@event[ "data" ] [ "status" ].ToString() == "Succeeded"
eventType	&&
Succeeded	@event[ "data" ] [ "operationName" ].ToString() == "Microsoft.Web/sites/write"
operationName	)
resourceProvider	

Explanation

Code segments	Answer Area
topic	if ( @event[ "data" ] [ "status" ].ToString() == "Succeeded" )
eventType	&& @event[ "data" ] [ "operationName" ].ToString() == "Microsoft.Web/sites/write"
resourceProvider	)

Box 1: Status

Box 2: Succeeded

Box 3: operationName

Scenario: 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.

#### NO.41 You need to resolve the Policy Loss issue.

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. Add an Azure Event Hub. Send the policy to the event hub. Configure the Policy service to read actions from the event hub.
- B. Add an Azure Service Bus queue. Send the policy to the queue. Configure the Policy service to read actions from the queue.
- C. Add an Azure Queue storage queue. Send the policy to the queue. Configure the Policy service to read actions from the queue.
- D. Add an Azure Service Bus topic. Send the policy to the topic. Configure the Policy service to read actions from the topic.

**Answer:** B D

**NO.42** Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution.

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

You need to ensure that authentication events are triggered and processed according to the policy.

Solution: Create a new Azure Event Grid subscription for all authentication that delivers messages to an Azure Event Hub. Use the subscription to process signout events.

Does the solution meet the goal?

- A. Yes
- B. No

**Answer:** B

Explanation

Use a separate Azure Event Grid topics and subscriptions for sign-in and sign-out events.

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.

**NO.43** 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 metric
- B.** an Application Insights dependency
- C.** an Application Insights trace
- D.** an Application Insights even

**Answer:** A

Explanation

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.

References:

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

**NO.44** Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution. After you answer a question in this section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

You need to ensure that authentication events are triggered and processed according to the policy.

Solution: Create separate Azure Event Grid topics and subscriptions for sign-in and sign-out events.

Does the solution meet the goal?

- A.** Yes
- B.** No

**Answer:** B

Explanation

Instead ensure that signout events have a subject prefix. Create an Azure Event Grid subscription that uses the subjectBeginsWith filter.

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.

**NO.45** 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.

**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);
    }
}
```

**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);
    }
}
```

**Explanation**

```

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.

References:

<https://docs.microsoft.com/en-us/azure/azure-monitor/app/api-filtering-sampling>

**NO.46** 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 or scroll to view content.

NOTE: Each correct selection is worth one point.

**Code segment**      **Answer Area**

All	{
WebHook	"name": "newlogs",
EventHub	"properties" : {
subjectEndsWith	"topic" : "/subscriptions/ . . ./providers/Microsoft.EventGrid/topics/. . .",
Microsoft.Storage	"destination": {
subjectBeginsWith	"endpointType" : " [ ]",
Microsoft.Storage BlobCreated	"filter": {
	" [ ]": "/blobServices/default/containers/logdrop/",
	"includeEventTypes": [" [ ]" ]},
	},
	"labels": [],
	"eventDeliverySchema": "EventGridSchema"
	}
	}

**Answer:**
**Code segment**      **Answer Area**

All	{
WebHook	"name": "newlogs",
EventHub	"properties" : {
subjectEndsWith	"topic" : "/subscriptions/ . . ./providers/Microsoft.EventGrid/topics/. . .",
Microsoft.Storage	"destination": {
subjectBeginsWith	"endpointType" : " [ WebHook ]",
Microsoft.Storage BlobCreated	"filter": {
	" [ subjectBeginsWith ]": "/blobServices/default/containers/logdrop/",
	"includeEventTypes": [" [ Microsoft.Storage.BlobCreated ]" ]},
	},
	"labels": [],
	"eventDeliverySchema": "EventGridSchema"
	}
	}

**Explanation**

```
{
    "name": "newlogs",
    "properties" : {
        "topic" : "/subscriptions/ . . ./providers/Microsoft.EventGrid/topics/. . .",
        "destination": {
            "endpointType" : " [ WebHook ]",
            "filter": {
                " [ subjectBeginsWith ]": "/blobServices/default/containers/logdrop/",
                "includeEventTypes": [" [ Microsoft.Storage.BlobCreated ]" ],
            },
            "labels": [],
            "eventDeliverySchema": "EventGridSchema"
        }
    }
}
```

**Box 1:WebHook**

Scenario: If an anomaly is detected, an Azure Function that emails administrators is called by using an HTTP WebHook.

endpointType: The type of endpoint for the subscription (webhook/HTTP, Event Hub, or queue).

Box 2: SubjectBeginsWith

Box 3: Microsoft.Storage.BlobCreated

Scenario: Log Policy

All Azure App Service Web Apps must write logs to Azure Blob storage. All log files should be saved to a container named logdrop. Logs must remain in the container for 15 days.

Example subscription schema

```
{
  "properties": {
    "destination": {
      "endpointType": "webhook",
      "properties": {
        "endpointUrl": "https://example.azurewebsites.net/api/HttpTriggerCSharp1?code=VXbGWce53I48Mt8wuotr0GPmyJ/nDT4hgd
      }
    },
    "filter": {
      "includedEventTypes": [ "Microsoft.Storage.BlobCreated", "Microsoft.Storage.BlobDeleted" ],
      "subjectBeginsWith": "blobServices/default/containers/mycontainer/log",
      "subjectEndsWith": ".jpg",
      "isSubjectCaseSensitive": "true"
    }
  }
}
```

References:

<https://docs.microsoft.com/en-us/azure/event-grid/subscription-creation-schema>

**NO.47** 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.

#### Code segments

- Process
- Initialize
- telemetry.Sequence
- ITelemetryProcessor
- ITelemetryInitializer
- telemetry.Context
- EventGridController.EventId.Value
- ((EventTelemetry)telemetry).Properties["EventId"]

#### Answer Area

```
public class IncludeEventId :  code
{
  public void  code segment
    (ITelemetry telemetry)
  {
     code segment
     code segment
  }
}
```

**Answer:**

**Code segments**

```
[Process]
[Initialize]
[telemetry.Sequence]
[ITelemetryProcessor]
[ITelemetryInitializer]
[telemetry.Context]
[EventGridController.EventId.Value]
[((EventTelemetry)telemetry.Properties["EventId"]]
```

**Answer Area**

```
public class IncludeEventId : ITelemetryInitializer
{
    public void Initialize(ITelemetry telemetry)
    {
        telemetry.Context.Properties["EventId"] =
            ((EventTelemetry)telemetry.Properties["EventId"]);
    }
}
```

**Explanation****Answer Area**

```
public class IncludeEventId: ITelemetryInitializer
{
    public void Initialize
        (ITelemetry telemetry)
    {
        Telemetry.Context.Properties["EventId"] =
            [(EventTelemetry)telemetry.Properties["EventId"]]
    }
}
```

---

**Topic 3, Chatbot****Background**

Best for You Organics Company is a global restaurant franchise that has multiple locations. The company wants to enhance user experiences and vendor integrations. The company plans to implement automated mobile ordering and delivery services.

Best For You Organics hosts an Azure web app at the URL <https://www.bestforyouorganics.com>. Users can use the web app to browse restaurant locations, menu items, nutritional information, and company information. The company developed and deployed a cross-platform mobile app.

**Requirements**

You must develop a chatbot by using the Bot Builder SDK and Language Understanding Intelligence Service (LUIS). The chatbot must allow users to order food for pickup or delivery.

The chatbot must meet the following requirements:

- \* Ensure that chatbot endpoint can be accessed only by the Bot Framework connector.
- \* Use natural language processing and speech recognition so that users can interact with the chatbot by using text and voice. Processing must be server-based.
- \* Alert users about promotions at local restaurants.
- \* Enable users to place an order for delivery or pickup by using their voice.
- \* Greet the user upon sign-in by displaying a graphical interface that contains action buttons.
- \* The chatbot greeting interface must match the formatting of the following example:

# Welcome to the Restaurant!



**John Doe**

Sun, Aug 26, 2018

Welcome to Best For You Organics Company! How can we help you today?

**Specials:** Chicken Marsala

## Order Pickup

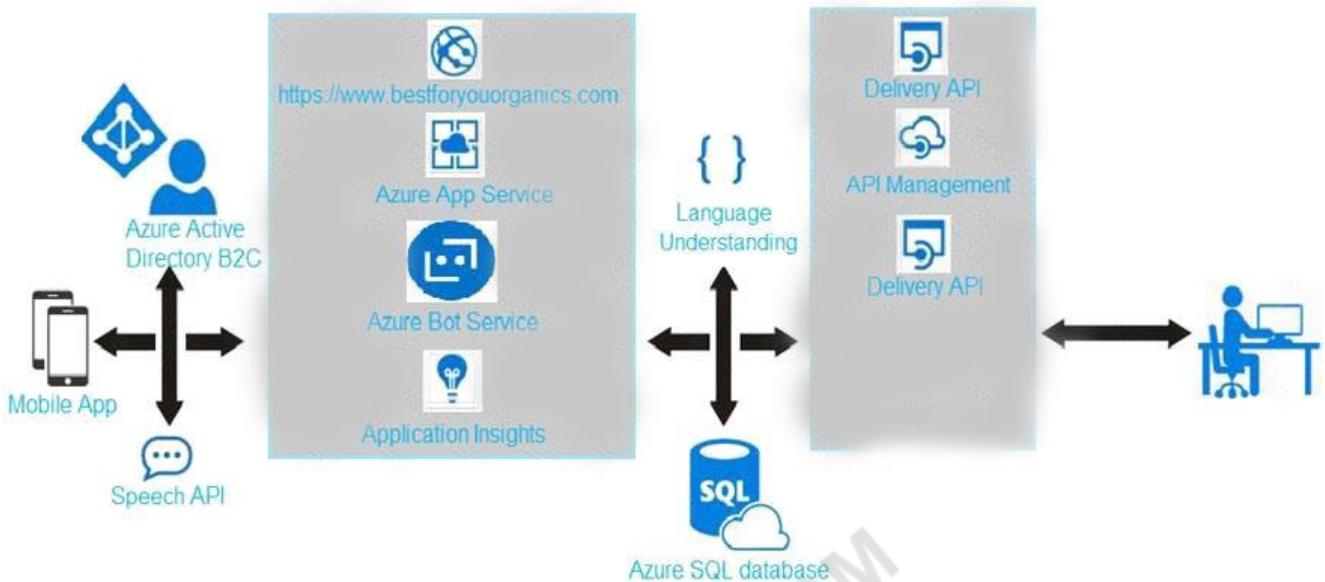
### Vendor API

Vendors receive and provide updates for the restaurant inventory and delivery services by using Azure API Management hosted APIs. Each vendor uses their own subscription to access each of the APIs.

APIs must meet the following conditions:

- \* API usage must not exceed 5,000 calls and 50,000 kilobytes of bandwidth per hour per vendor.
- \* If a vendor is nearing the number of calls or bandwidth limit, the API must trigger email notifications to the vendor.
- \* APIs must prevent API usage spikes on a per-subscription basis by limiting the call rate to 100 calls per minute.
- \* The Inventory API must be written by using ASP.NET Core and Node.js.
- \* The API must be updated to provide an interface to Azure SQL Database. Database objects must be managed by using code.
- \* The Delivery API must be protected by using the OAuth 2.0 protocol with Azure Active Directory (Azure AD) when called from the Azure web app. You register the Delivery API and web app in Azure AD. You enable OAuth 2.0 in the web app.
- \* The delivery API must update the Products table, the Vendor transactions table, and the Billing table in a single transaction.

The Best For You Organics Company architecture team has created the following diagram depicting the expected deployments into Azure:



## Delivery API

The Delivery API intermittently throws the following exception:

```
"System.Data.Entity.Core.EntityCommandExecutionException: An error occurred while executing the command definition. See the inner exception for details. --->System.Data.SqlClient.SqlException: A transport-level error has occurred when receiving results from the server. (provider: Session Provider, error: 19 - Physical connection is not usable)"
```

## Chatbot greeting

The chatbot's greeting does not show the user's name. You need to debug the chatbot locally.

## Language processing

Users report that the bot fails to understand when a customer attempts to order dishes that use Italian names.

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.

Startup.cs

```

SU01  namespace DeliveryApi
SU02  {
SU03    public class Startup
SU04    {
SU05      public Startup(IConfiguration configuration)
SU06      {
SU07        Configuration = configuration ;
SU08      }
SU09      public IConfiguration Configuration {get ;}
SU10      public void ConfigureServices(IServiceCollection services)
SU11    {
SU12      services.AddDbContext<RestaurantsContext> (opt =>
SU13        opt.UseSqlServer (Configuration.GetSection ("ConnectionStrings") ["RestaurantDatabase"]),
SU14        sqlServerOptionsAction: sqlOptions =>
SU15        {
SU16          . .
SU17        }));
SU18      services.AddMvc()
SU19      .SetCompatibilityVersion(CompatibilityVersion.Version_2_1) ;
SU20    }
SU21    public void Configure (IApplicationBuilder app)
SU22    {
SU23      app.UseMvc() ;
SU24    }
SU25  }
SU26 }

```

**NO.48** You need to debug the user greeting issue. What should you use?

- A.** Bot Framework Channel Inspector
- B.** Bot Connector service
- C.** Azure Compute Emulator
- D.** Azure Application Insights
- E.** Bot Framework Emulator

**Answer:** E

Explanation

Scenario: The chatbot's greeting does not show the user's name. You need to debug the chatbot locally.

Debug your bot using an integrated development environment (IDE) such as Visual Studio or Visual Studio Code and the Bot Framework Emulator. You can use these methods to debug any bot locally.

References:

<https://docs.microsoft.com/en-us/azure/bot-service/bot-service-debug-bot?view=azure-bot-service-4.0>

**NO.49** You need to implement the purchase requirement.

What should you do?

- A.** Use the Bot Framework REST API attachment operations to send the user's voice and the Speech Service API to recognize intents.
- B.** Use the Direct line REST API to send the user's voice and the Speech Service API to recognize intents.
- C.** Use the Speech Service API to send the user's voice and the Bot Framework REST API conversation operations to recognize intents.

- D.** Use the Bot Framework REST API conversation operations to send the user's voice and the Speech Service API to recognize intents.

**Answer:** D

Explanation

Scenario: Enable users to place an order for delivery or pickup by using their voice.

You must develop a chatbot by using the Bot Builder SDK and Language Understanding Intelligence Service (LUIS). The chatbot must allow users to order food for pickup or delivery.

The Bot Framework REST APIs enable you to build bots that exchange messages with channels configured in the Bot Framework Portal, store and retrieve state data, and connect your own client applications to your bots.

All Bot Framework services use industry-standard REST and JSON over HTTPS.

The Speech Service API is used to recognize intents.

References:

<https://docs.microsoft.com/en-us/azure/bot-service/rest-api/bot-framework-rest-connector-concepts?view=azure>

<https://docs.microsoft.com/en-us/azure/cognitive-services/speech-service/how-to-recognize-intents-from-speech->

**NO.50** 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 need to meet the vendor notification requirement.

Solution: Configure notifications in the Azure API Management instance.

Does the solution meet the goal?

**A.** Yes

**B.** No

**Answer:** B

Explanation

Use a custom outbound Azure API Management policy.

Scenario:

If a vendor is nearing the number of calls or bandwidth limit, the API must trigger email notifications to the vendor.

(API usage must not exceed 5,000 calls and 50,000 kilobytes of bandwidth per hour per vendor.)

References:

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

**NO.51** Note: In this section you will see one or more sets of questions with the same scenario and problem. Each question presents a unique solution to the problem, and you must determine whether the solution meets the stated goals. More than one solution might solve the problem. It is also possible that none of the solutions solve the problem.

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.

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 need to meet the vendor notification requirement.

Solution: Update the Delivery API to send emails by using a Microsoft Office 365 SMTP server.

Does the solution meet the goal?

A. Yes

B. No

**Answer:** B

Explanation

Use a custom outbound Azure API Management policy.

Scenario:

If a vendor is nearing the number of calls or bandwidth limit, the API must trigger email notifications to the vendor.

(API usage must not exceed 5,000 calls and 50,000 kilobytes of bandwidth per hour per vendor.)

References:

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

**NO.52** 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 need to meet the vendor notification requirement.

Solution: Create and apply a custom outbound Azure API Management policy.

Does the solution meet the goal?

A. Yes

B. No

**Answer:** A

Explanation

Scenario:

If a vendor is nearing the number of calls or bandwidth limit, the API must trigger email notifications to the vendor.

(API usage must not exceed 5,000 calls and 50,000 kilobytes of bandwidth per hour per vendor.) In Azure API Management (APIM), policies are a powerful capability of the system that allow the publisher to change the behavior of the API through configuration. Policies are a collection of Statements that are executed sequentially on the request or response of an API. Popular Statements include format conversion from XML to JSON and call rate limiting to restrict the amount of incoming calls from a developer. Many more policies are available out of the box.

References:

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

**NO.53** You need to update the chatbot to greet the user when they sign in.

Which two rich card formats can you use? Each correct answer presents a complete solution.

NOTE: Each correct selection is worth one point

A. Thumbnail

B. Adaptive

C. Sign-in

D. Hero

E. Animation

**Answer:** A C

Explanation

Scenario: The chatbot greeting interface must match the formatting of the following example:

## Welcome to the Restaurant!



**John Doe**

Sun, Aug 26, 2018

Welcome to Best For You Organics Company! How can we help you today?

**Specials:** Chicken Marsala

### Order Pickup

### Order Delivery

A message exchange between user and bot can contain one or more rich cards rendered as a list or carousel.

The Attachments property of the Activity object contains an array of Attachment objects that represent the rich cards and media attachments within the message.

The Bot Framework currently supports eight types of rich cards:

- \* Thumbnail Card. A card that typically contains a single thumbnail image, one or more buttons, and text.
- \* SignIn Card. A card that enables a bot to request that a user sign-in. It typically contains text and one or more buttons that the user can click to initiate the sign-in process.

**NO.54** 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 need to meet the vendor notification requirement.

Solution: Update the Delivery API to send emails by using a cloud-based email service.

Does the solution meet the goal?

**A.** Yes

**B.** No

**Answer:** B

**NO.55** You need to update the Inventory API.

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

NOTE: Each correct selection is worth one point.

# Development Tool

## Technology

ADO.NET
Entity Framework
Entity Framework Core
WCF Data Services

## Workflow

Model first
Database first
Code first

**Answer:**

# Development Tool

## Technology

ADO.NET
Entity Framework
Entity Framework Core
WCF Data Services

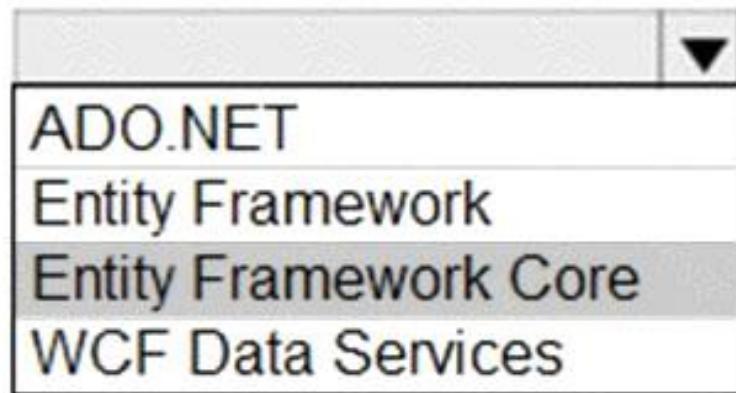
## Workflow

Model first
Database first
Code first

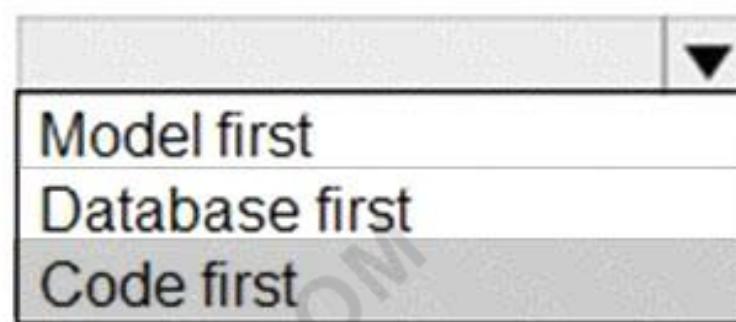
Explanation

# Development Tool

## Technology



## Workflow



Scenario: The Inventory API must be written by using ASP.NET Core and Node.js.

Box 1: Entity Framework Core

Box 2: Code first

References:

<https://docs.microsoft.com/en-us/aspnet/mvc/overview/getting-started/getting-started-with-ef-using-mvc/creating>

**NO.56** You need to resolve the delivery API error. What should you do?

- A. Implement simple retry by using the `EnableRetryOnFailure` feature of Entity Framework.
- B. Implement exponential backoff by using the `EnableRetryOnFailure` feature of Entity Framework.
- C. Implement the Circuit Breaker pattern by using the `EnableRetryOnFailure` feature of Entity Framework.
- D. Invoke a custom execution strategy in Entity Framework.

**Answer:** A

Explanation

Scenario: The Delivery API intermittently throws the following exception:

```
"System.Data.Entity.Core.EntityCommandExecutionException: An error occurred while executing the
command definition. See the inner exception for details. --->System.Data.SqlClient.SqlException:
A transport-level error has occurred when receiving results from the server. (provider: Session
Provider, error: 19 - Physical connection is not usable)"
```

A useful method to get rid of this error is to use RETRY LOGIC of Entity Framework 1.1.0  
`services.AddDbContext<DbContext>(options => options.UseSqlServer('yourconnectionstring',
...sqlServerOptionsAction: sqlOptions =>`

```
...{  
.....sqlOptions.EnableRetryOnFailure(  
.....maxRetryCount: 5,  
.....maxRetryDelay: TimeSpan.FromSeconds(30),  
.....errorNumbersToAdd: new List<int>() { 19 });  
...});
```

In Retry logic, error 19 is not included. So you have to pass the error code 19 to set retry logic for error code

19.

References:

<https://stackoverflow.com/questions/47558062/error-19-physical-connection-error/47559967>

---

## Topic 4, Mix QuestionsCase Study

### 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.

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.

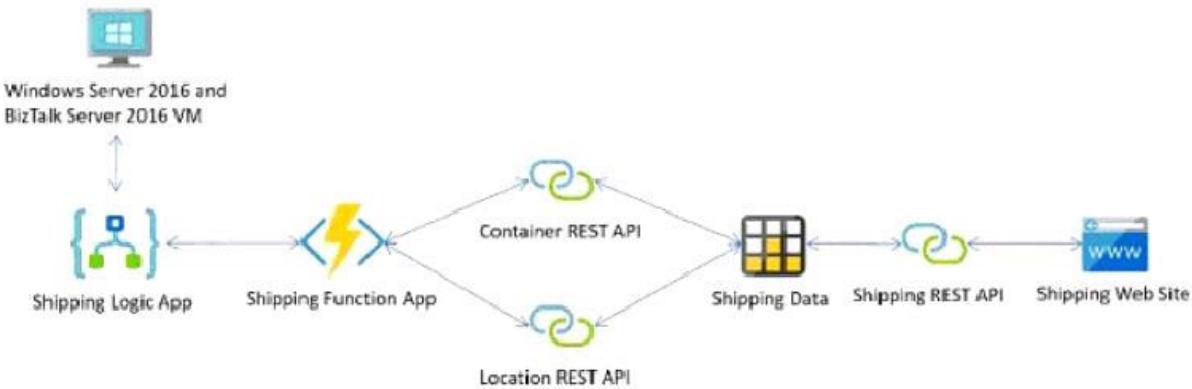
The VM supports the following REST API calls:

- \* 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 storage database for all container and transport information Shipping Website The site displays containers tracking information and containers content. The site is located at

<http://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:



### 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 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 m-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.
```

**NO.57** You have an application that provides weather forecasting data to external partners. You use Azure API Management to publish APIs.

You must change the behavior of the API to meet the following requirements:

- \* Support alternative input parameters.
- \* Remove formatting text from responses.
- \* Provide additional context to back-end services.

Which types of policies should you implement? To answer, drag the policy types to the correct scenarios. Each policy type may be used once, more than once, or not at all. You may need to drag the split bar between panes or scroll to view content. NOTE: Each correct selection is worth one point.

Policy types	Answer Area	Requirement	Policy type
Inbound		Rewrite the request URL to match to the format expected by the web service.	policy type
Outbound		Remove formatting text from responses.	policy type
Backend		Forward the user ID that is associated with the subscription key for the original request to the back-end service.	policy type

### Answer:

Policy types	Answer Area	Requirement	Policy type
Inbound		Rewrite the request URL to match to the format expected by the web service.	Outbound
Outbound		Remove formatting text from responses.	Inbound
Backend		Forward the user ID that is associated with the subscription key for the original request to the back-end service.	Backend

### Explanation

### Requirement

Rewrite the request URL to match to the format expected by the web service.

Remove formatting text from responses.

Forward the user ID that is associated with the subscription key for the original request to the back-end service

### Policy type

Outbound

Inbound

Backend

**NO.58** 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 question, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

Margie's Travel is an international travel and bookings management service. The company is expanding into restaurant bookings. You are tasked with implementing Azure Search for the restaurants listed in their solution.

You create the index in Azure Search.

You need to import the restaurant data into the Azure Search service by using the Azure Search .NET SDK.

Solution:

1. Create a `SearchServiceClient` object to connect to the search index.

2. Create a DataContainer that contains the documents which must be added.
3. Create a DataSource instance and set its Container property to the DataContainer.
4. Set the DataSources property of the SearchServiceClient.

Does the solution meet the goal?

**A.** Yes

**B.** No

**Answer:** B

Explanation

Use the following method:

- \* - Create a SearchIndexClient object to connect to the search index
- \* - Create an IndexBatch that contains the documents which must be added.
- \* - Call the Documents.Index method of the SearchIndexClient and pass the IndexBatch.

References:

<https://docs.microsoft.com/en-us/azure/search/search-howto-dotnet-sdk>

**NO.59** You are developing a software solution for an autonomous transportation system. The solution uses large data sets and Azure Batch processing to simulate navigation sets for entire fleets of vehicles.

You need to create compute nodes for the solution on Azure Batch.

What should you do?

- A.** In Python, implement the class: TaskAddParameter
- B.** In Python, implement the class: JobAddParameter
- C.** In the Azure portal, create a Batch account
- D.** In a .NET method, call the method: BatchClient.PoolOperations.CreateJob

**Answer:** D

Explanation

A Batch job is a logical grouping of one or more tasks. A job includes settings common to the tasks, such as priority and the pool to run tasks on. The app uses the BatchClient.JobOperations.CreateJob method to create a job on your pool.

Note:

Step 1: Create a pool of compute nodes. When you create a pool, you specify the number of compute nodes for the pool, their size, and the operating system. When each task in your job runs, it's assigned to execute on one of the nodes in your pool.

Step 2 : Create a job. A job manages a collection of tasks. You associate each job to a specific pool where that job's tasks will run.

Step 3: Add tasks to the job. Each task runs the application or script that you uploaded to process the data files it downloads from your Storage account. As each task completes, it can upload its output to Azure Storage.

**NO.60** 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.

### Answer Area

```
az provider register -n Microsoft.KeyVault  
resourcegroup="myResourceGroup"  
az group create --name $resourcegroup --location westus  
keyvault_name=myvaultname$RANDOM  
az vm create --name $vm_name --resource-group $resourcegroup --location westus --enable-vm-disk-encryption True  
az vm create --name $vm_name --resource-group $resourcegroup --location westus --key-vault $keyvault_name  
az vm create --name $vm_name --resource-group $resourcegroup --location westus --image Canonical:UbuntuServer:16.04-LTS:latest  
az vm enable-vm-encryption --vm-name $vm_name --key-vault $keyvault_name --key-name $key_name  
--volume-type all
```

**Answer:**

**Answer Area**

```
az provider register -n Microsoft.KeyVault
resourcegroup="myResourceGroup"
az group create --name $resourcegroup --location westus
keyvault_name=myvaultname$RANDOM
az vm create \
    --name myvm \
    --resource-group $resourcegroup \
    --location westus \
    --image Canonical:UbuntuServer:16.04-LTS:latest \
    --size Standard_DS1_v2 \
    --admin-username myuser \
    --admin-password $(az keyvault secret show --name mysecret --vault-name $keyvault_name --query value --output tsv) \
    --vnet-name myvnet \
    --subnet mysubnet \
    --public-ip-address myip \
    --os-type Linux \
    --storage-sku Standard_LRS \
    --os-disk-size-gb 20 \
    --data-disk-count 1 \
    --data-disk-size-gb 10 \
    --data-disk-lun 1 \
    --data-disk-vnet-name myvnet \
    --data-disk-subnet mysubnet \
    --data-disk-access-level ReadWrite \
    --data-disk-name mydata \
    --data-disk-encryption-key-vault $keyvault_name \
    --data-disk-encryption-key-name mykey \
    --data-disk-encryption-type Software \
    --data-disk-encryption-volume-type Os
az vm encryption enable \
    --vm myvm \
    --resource-group $resourcegroup \
    --key-vault $keyvault_name \
    --key-name mykey \
    --volume-type Os
```

## Explanation

```
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 create\

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.

References:

<https://docs.microsoft.com/bs-latn-ba/azure/virtual-machines/linux/encrypt-disks>

**NO.61** 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

**Answer:** D

Explanation

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

```

**NO.62** You are developing a .NET Core model-view controller (MVC) application hosted on Azure for a health care system that allows providers access to their information.

You develop the following code:

```

services.AddAuthorization (options =>
{
    options.AddPolicy("ProviderPartner", policy =>
    {
        policy.AddAuthenticationSchemes("Cookie, Bearer");
        policy.RequireAuthenticatedUser();
        policy.RequireRole("ProviderAdmin", "SysAdmin");
        policy.RequireClaim("editor", "partner");
    });
}

```

You define a role named SysAdmin.

You need to ensure that the application meets the following authorization requirements:

- \* Allow the ProviderAdmin and SysAdmin roles access to the Partner controller regardless of whether the user holds an editor claim of partner.
- \* Limit access to the Manage action of the controller to users with an editor claim of partner who are also members of the SysAdmin role.

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.

**Code Segments**

```
[Authorize (Policy = "ProviderEditor")]
[Authorize(Role = "SysAdmin")]
```

```
[Authorize(Role = "ProviderAdmin")]
[Authorize(Role = "SysAdmin")]
```

```
[Authorize(Role = "SysAdmin", "ProviderAdmin")]
```

```
[Authorize(Policy = "ProviderEditor", Role= "SysAdmin")]
```

**Answer Area**

```
public class PartnerController : Controller
{
    .
    .

    public ActionResult Manage()
    {
        .
        .
    }
}
```

**Answer:****Code Segments**

```
[Authorize (Policy = "ProviderEditor")]
[Authorize(Role = "SysAdmin")]
```

```
[Authorize(Role = "ProviderAdmin")]
[Authorize(Role = "SysAdmin")]
```

```
[Authorize(Role = "SysAdmin", "ProviderAdmin")]
```

```
[Authorize(Policy = "ProviderEditor", Role= "SysAdmin")]
```

**Answer Area**

```
[Authorize(Role = "ProviderAdmin")]
[Authorize(Role = "SysAdmin")]

public class PartnerController : Controller
{
    .
    .

    [Authorize(Policy = "ProviderEditor", Role= "SysAdmin")]

    public ActionResult Manage()
    {
        .
        .
    }
}
```

**Explanation**

```
[Authorize(Role = "ProviderAdmin")]
[Authorize(Role = "SysAdmin")]

public class PartnerController : Controller
{
    .
    .
}
```

```
[Authorize(Policy = "ProviderEditor", Role= "SysAdmin")]
```

```
Public ActionResult Manage()
{
    .
    .
}
```

**Box 1:**

Allow the ProviderAdmin and SysAdmin roles access to the Partner controller regardless of whether the user holds an editor claim of partner.

**Box 2:**

Limit access to the Manage action of the controller to users with an editor claim of partner who are also members of the SysAdmin role.

**NO.63** You store customer information in an 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. (Line numbers are included for reference only.)

```

1 CloudTableClient tableClient = account.CreateCloudTableClient();
2 CloudTable table = tableClient.GetTableReference("people");
3 TableQuery<CustomerEntity> query = new TableQuery<CustomerEntity>()
4 .Where(TableQuery.CombineFilters(
5 TableQuery.GenerateFilterCondition("PartitionKey", QueryComparisons.Equal, "Smith"),
6 TableOperators.And, TableQuery.
    GenerateFilterCondition("Email", QueryComparisons.Equal, "ssmith@contoso.com"))
7 );
8 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.

Yes	No
-----	----

The code returns every Record where the surname equals **Smith**.



The table endpoint <https://<mytableendpoint>/People>  
(PartitionKey='Smith',RowKey='Steve') returns the same results as the code.



**Answer:**

Yes	No
-----	----

The code returns every Record where the surname equals **Smith**.



The table endpoint <https://<mytableendpoint>/People>  
(PartitionKey='Smith',RowKey='Steve') returns the same results as the code.



Explanation

Yes	No
-----	----

The code returns every Record where the surname equals **Smith**.



The table endpoint <https://<mytableendpoint>/People>  
(PartitionKey='Smith',RowKey='Steve') returns the same results as the code.



Box 1: No

Only the second row is returned thank to the Email filter condition.

Box 2: Yes

This also returns the second row.

**NO.64** You are developing a ticket reservation system for an airline.

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

- \* Ensure at least 99/99% availability and provide low latency.
  - \* Accept reservations even in network outages or other unforeseen failures
  - \* 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.

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

`consistencyLevel=`

Strong
Eventual
ConsistentPrefix
BoundedStaleness

```
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:**

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

`consistencyLevel=`

Strong
Eventual
ConsistentPrefix
BoundedStaleness

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

```
--enable-virtual-network true \
--enable-automatic-failover true \
--kind 'GlovalDocumentDB' \
--kind 'MongoDB' \
```

```
- -resource-group $resourceGroupName \
- -max interval 5 \
```

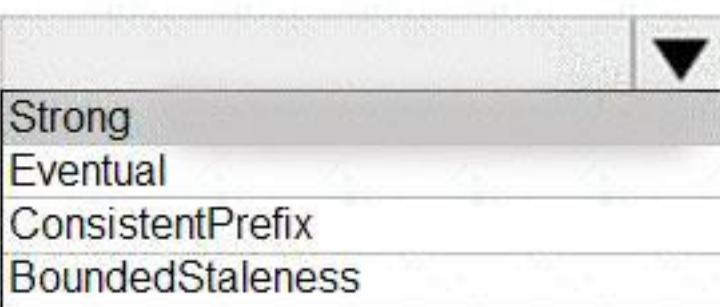
```
--locations 'southcentralus' \
--locations 'eastus' \
--locations 'southcentralus=0 eastus=1 westus=2' \
--locations 'southcentralus=0'
```

`- -default-consistency-level = $consistencyLevel`

Explanation

```
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 'GlovalDocumentDB' \
--kind 'MongoDB' \
  --resource-group $resourceGroupName \
  --max interval 5 \
```

```
--locations 'southcentralus'
--locations 'eastus'
--locations 'southcentralus=0 eastus=1 westus=2'
--locations 'southcentralus=0'
```

```
- --default-consistency-level = $consistencyLevel
```

Box 1: 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.

" Process reservations in the exact sequence as reservation are submitted to minimize overbooking or selling the same seat to multiple travelers." Box 2: Enable-automatic-failover 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.

" Accept reservations even in network outages or other unforeseen failures" Box 3: southcentralus  
" You provision a resource group named airlineResourceGroup in the Azure South-Central US region."

References:

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

<https://github.com/MicrosoftDocs/azure-docs/blob/master/articles/cosmos-db/high-availability.md>

**NO.65** 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

**Answer:** B

**NO.66** 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. `New-AzureRmServiceBusNamespace`
  - ResourceGroupName `fridge-rg`
  - NamespaceName `fridge-ns`
  - Location `fridge-loc`
- B. `az servicebus namespace create`
  - -resource-group `fridge-rg`
  - -name `fridge-ns`
  - -location `fridge-loc`
- C. `New-AzureRmResourceGroup`
  - Name `fridge-rg`
  - Location `fridge-loc`
- D. `New-AzureRmServiceBusQueue`
  - ResourceGroupName `fridge-rg`
  - NamespaceName `fridge-ns`
  - Name `fridge-q`
  - EnablePartitioning `$False`

- A. Option A
- B. Option B
- C. Option C
- D. Option D

**Answer:** B

Explanation

# Create a Service Bus messaging namespace with a unique name. Example:

namespaceName=myNameSpace\$RANDOM

az servicebus namespace create --resource-group \$resourceGroupName --name \$namespaceName  
--location eastus References:

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

**NO.67** 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.

Features	Requirement	Feature
Users	Which pages visited by users most often correlate to a product purchase?	
Funnels	How does load time of the product display page affect a user's decision to purchase a product?	
Impact	Which events most influence a user's decision to continue to use the application?	
Retention	Are there places in the application that users often perform repetitive actions?	
User Flows		

**Answer:**

Features	Requirement	Feature
Users	Which pages visited by users most often correlate to a product purchase?	Users
Funnels	How does load time of the product display page affect a user's decision to purchase a product?	Impact
Impact	Which events most influence a user's decision to continue to use the application?	Retention
Retention	Are there places in the application that users often perform repetitive actions?	User Flows
User Flows		

Explanation

Requirement	Feature
Which pages visited by users most often correlate to a product purchase?	Users
How does load time of the product display page affect a user's decision to purchase a product?	Impact
Which events most influence a user's decision to continue to use the application?	Retention
Are there places in the application that users often perform repetitive actions?	User Flows

Box1: 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?

**NO.68** 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.

```
void ClearCachedTeams()
```

```
{
```

IDatabase cache = Connection.GetDatabase();	▼
---	---

ICache cache = Connection.GetDatabase();	▼
--	---

cache.KeyDelete("teams");	▼
---------------------------	---

cache.StringSet("teams", "");	▼
-------------------------------	---

cache.ValueDelete("teams");	▼
-----------------------------	---

cache.StringGet("teams", "");	▼
-------------------------------	---

viewBag.nsg += Team data removed from cache.

```
}
```

**Answer:**

```

void ClearCachedTeams()
{
    IDatabase cache = Connection.GetDatabase();
    ICache cache = Connection.GetDatabase();

    cache.KeyDelete("teams");
    cache.StringSet("teams", "");
    cache.ValueDelete("teams");
    cache.StringGet("teams", "");

    viewBag.nsg += Team data removed from cache.
}

```

## Explanation

```

void ClearCachedTeams()
{
    IDatabase cache = Connection.GetDatabase();
    ICache cache = Connection.GetDatabase();

    cache.KeyDelete("teams");
    cache.StringSet("teams", "");
    cache.ValueDelete("teams");
    cache.StringGet("teams", "");

    viewBag.nsg += 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));`

## References:

<https://azure.microsoft.com/sv-se/blog/lap-around-azure-redis-cache-preview/>

**NO.69** You are deploying an Azure Kubernetes Services (AKS) cluster that will use multiple containers.

You need to create the cluster and verify that the services for the containers are configured correctly and available.

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

**Command segments**

az aks get-credentials

az appservice plan create

az aks create

az group create

kubectl apply

**Answer Area**

**Answer:**

**Command segments**

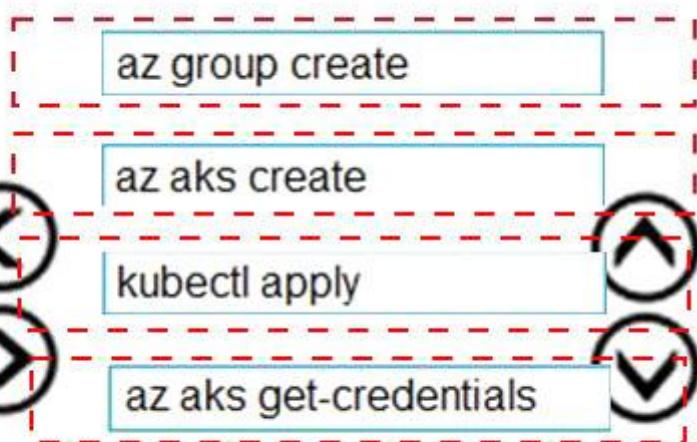
az aks get-credentials

az appservice plan create

az aks create

az group create

kubectl apply

**Answer Area**

Explanation

**az group create****az aks create****kubectl apply****az aks get-credentials**

Step 1: az group create

Create a resource group with the az group create command. An Azure resource group is a logical group in which Azure resources are deployed and managed.

Example: The following example creates a resource group named myAKSCluster in the eastus location.

```
az group create --name myAKSCluster --location eastus
```

Step 2 : az aks create

Use the az aks create command to create an AKS cluster.

Step 3: kubectl apply

To deploy your application, use the kubectl apply command. This command parses the manifest file and creates the defined Kubernetes objects.

Step 4: az aks get-credentials

Configure it with the credentials for the new AKS cluster. Example:

```
az aks get-credentials --name aks-cluster --resource-group aks-resource-group
```

[References:  
https://docs.bitnami.com/azure/get-started-aks/](https://docs.bitnami.com/azure/get-started-aks/)

### NO.70 You are developing Azure WebJobs.

You need to recommend a WebJob type for each scenario.

Which WebJob type should you recommend? To answer, drag the appropriate WebJob types to the correct scenarios. Each WebJob 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.

WebJob types	Scenario	WebJob type
Triggered	Run on all instances that the web app runs on. Optionally restrict the WebJob to a single instance.	
Continuous	Run on a single instance that Azure select for load balancing.	
	Supports remote debugging	

**Answer:**

WebJob types	Scenario	WebJob type
Triggered	Run on all instances that the web app runs on. Optionally restrict the WebJob to a single instance.	Continuous
Continuous	Run on a single instance that Azure select for load balancing.	Triggered
	Supports remote debugging	Continuous

Explanation

Scenario	WebJob type
Run on all instances that the web app runs on. Optionally restrict the WebJob to a single instance.	Continuous
Run on a single instance that Azure select for load balancing.	Triggered
Supports remote debugging	Continuous

Box 1: Continuous

Continuous runs on all instances that the web app runs on. You can optionally restrict the WebJob to a single instance.

Box 2: Triggered

Triggered runs on a single instance that Azure selects for load balancing.

Box 3: Continuous

Continuous supports remote debugging.

Note:

The following table describes the differences between continuous and triggered WebJobs.

Continuous	Triggered
Starts immediately when the WebJob is created. To keep the job from ending, the program or script typically does its work inside an endless loop. If the job does end, you can restart it.	Starts only when triggered manually or on a schedule.
Runs on all instances that the web app runs on. You can optionally restrict the WebJob to a single instance.	Runs on a single instance that Azure selects for load balancing.
Supports remote debugging.	Doesn't support remote debugging.

References:

<https://docs.microsoft.com/en-us/azure/app-service/web-sites-create-web-jobs>

**NO.71** 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.** Certificate Authentication
- B.** Basic Authentication
- C.** OAuth Client Credential Grant
- D.** Digest Authentication

**Answer:** A C

**NO.72** 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 complete the source code of the subscription client

What should you do?

- A.** await subscriptionClient.CloseAsync();
- B.** await subscriptionClient.AddRuleAsync(new RuleDescription(RuleDescription.DefaultRuleName, new TrueFilter()));
- C.** subscriptionClient.RegisterMessageHandler(ProcessMessagesAsync, messageHandlerOptions);
- D.** subscriptionClient = new SubscriptionClient(ServiceBusConnectionString, TopicName, SubscriptionName);

**Answer:** C

Explanation

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);

References:

<https://www.c-sharpcorner.com/article/azure-service-bus-topic-and-subscription-pub-sub/>

**NO.73** 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 an App Service that uses the Shared service tier. Configure the App Service plan to automatically scale when the CPU load is high.
- B.** Deploy the website to a virtual machine. Configure the virtual machine to automatically scale when the CPU load is high.
- C.** 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.
- D.** Deploy the website to a virtual machine. Configure a Scale Set to increase the virtual machine

instance count when the CPU load

**Answer:** C

Explanation

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.

**NO.74** 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

Policy section	Answer Area	Policy	Policy section
Inbound		Set-variable	policy section
Outbound		Cache-lookup-value	policy section
		Cache-store-value	policy section
		Find-and-replace	policy section

**Answer:**

Policy section	Answer Area	Policy	Policy section
Inbound		Set-variable	Inbound
Outbound		Cache-lookup-value	Inbound
		Cache-store-value	Outbound
		Find-and-replace	Outbound

Explanation

Policy	Policy section
Set-variable	Inbound
Cache-lookup-value	Inbound
Cache-store-value	Outbound
Find-and-replace	Outbound

**NO.75** 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/weapp>.

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.

```
$gitrepo="https://github.com/TailSpinToys/weapp"
$webappname="TailSpinToysWeb"
$location="WestUS2"
 New-AzWebAppSlot -Name myResourceGroup -Location $location
 New-AzWebApp
 New-AzAppServicePlan
 New-AzResourceGroup
 New-AzWebAppSlot -Name $webappname -Location $location -ResourceGroupName myResourceGroup -Tier Standard
 New-AzWebApp
 New-AzAppServicePlan
 New-AzResourceGroup
 New-AzWebAppSlot -Name $webappname -Location $location -AppServicePlan $webappname -ResourceGroupName myResourceGroup
 New-AzWebApp
 New-AzAppServicePlan
 New-AzResourceGroup
 New-AzWebAppSlot -Name $webappname -ResourceGroupName myResourceGroup -Slot review
 New-AzWebApp
 New-AzAppServicePlan
 New-AzResourceGroup
$PropertiesObject = @{repoUrl = "$gitrepo";branch = "master";}
Set-AzResource -PropertyObject $PropertiesObject -ResourceGroupName myResourceGroup -ResourceType Microsoft.Web/sites/slots/sourcecontrols -ResourceName $webappname/review/web -ApiVersion 2015-08-01 -Force
Switch-AzWebAppSlot -Name $webappname -ResourceGroupName myResourceGroup -
-SourceSlotName review -DestinationSlotName production
```

**Answer:**

```
$gitrepo="https://github.com/TailSpinToys/webapp"
$webappname="TailSpinToysWeb"
$location="WestUS2"
New-AzWebAppSlot -Name myResourceGroup -Location $location
New-AzWebApp
New-AzAppServicePlan
New-AzResourceGroup
New-AzWebAppSlot -Name $webappname -Location $location -ResourceGroupName myResourceGroup -Tier Standard
New-AzWebApp
New-AzAppServicePlan
New-AzResourceGroup
New-AzWebAppSlot -Name $webappname -Location $location -AppServicePlan $webappname -ResourceGroupName myResourceGroup
New-AzWebApp
New-AzAppServicePlan
New-AzResourceGroup
New-AzWebAppSlot -Name $webappname -ResourceGroupName myResourceGroup -Slot review
New-AzWebApp
New-AzAppServicePlan
New-AzResourceGroup

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

## Explanation

```
$gitrepo="https://github.com/TailSpinToys/webapp"
$webappname="TailSpinToysWeb"
$location="WestUS2"
New-AzWebAppSlot -Name myResourceGroup -Location $location
New-AzWebApp
New-AzAppServicePlan
New-AzResourceGroup
New-AzWebAppSlot -Name $webappname -Location $location -ResourceGroupName myResourceGroup -Tier Standard
New-AzWebApp
New-AzAppServicePlan
New-AzResourceGroup
New-AzWebAppSlot -Name $webappname -Location $location -AppServicePlan $webappname -ResourceGroupName myResourceGroup
New-AzWebApp
New-AzAppServicePlan
New-AzResourceGroup
New-AzWebAppSlot -Name $webappname -ResourceGroupName myResourceGroup -Slot review
New-AzWebApp
New-AzAppServicePlan
New-AzResourceGroup

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

The New-AzResourceGroup cmdlet creates an Azure resource group.

The New-AzAppServicePlan cmdlet creates an Azure App Service plan in a given location  
The New-AzWebApp cmdlet creates an Azure Web App in a given a resource group  
The New-AzWebAppSlot cmdlet creates an Azure Web App slot.

## References:

<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>

**NO.76** A company runs an international travel and bookings management service. The company plans to begin offering restaurant bookings. You must develop a solution that uses Azure Search and meets the following requirements:

\* Users must be able to search for restaurants by name, description, location, and cuisine.

- \* Users must be able to narrow the results further by location, cuisine, rating, and family-friendliness.
- \* All words in descriptions must be included in searches.

You need to add annotations to the restaurant class.

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.

PDFDUMPS.COM

```
[SerializePropertyNameAsCamelCase]
public class Restaurant
{
    [Key, IsFilterable]
    public int RestaurantId { get; set; }
    [IsSearchable, IsFilterable, IsSortable]
    public string Name { get; set; }

    [IsSearchable, IsFilterable, IsSortable, IsFacetable]
    [IsFilterable, IsFacetable, Required]
    [IsSearchable]
    [IsSearchable, Required]

    public string location { get; set; }
    public string Phone { get; set; }

    [Required]
    [IsSearchable]
    [IsFilterable, IsFacetable, Required]
    [IsFilterable, IsFacetable, IsSortable]

    public string Description { get; set; }

    [IsFiltrable, IsSortable, IsSearchable]
    [IsFilterable, IsSortable, IsFacetable]
    [IsFiltrable, IsSortable, Key]
    [IsFilterable, IsSortable, IsSearchable, Required]

    public double Rating { get; set; }

    [IsSearchable, IsFilterable, IsFacetable]
    [IsFilterable, IsSortable, Key]
    [IsFilterable, IsSortable, IsSearchable]
    [IsFilterable, IsSortable, Key, Required]

    public List<string> Cuisines { get; set; }

    [IsFilterable, IsSortable, Key, Required]
    [IsSearchable, IsSortable, IsFacetable]
    [IsFilterable, IsSortable, Key, IsSearchable]
    [IsFilterable, IsFacetable]

    public bool FamilyFriendly { get; set; }
```

**Answer:**

PDFDUMPS.COM

```
[SerializePropertyNameAsCamelCase]
public class Restaurant
{
    [Key, IsFilterable]
    public int RestaurantId { get; set; }
    [IsSearchable, IsFilterable, IsSortable]
    public string Name { get; set; }

    [IsSearchable, IsFilterable, IsSortable, IsFacetable]
    [IsFilterable, IsFacetable, Required]
    [IsSearchable]
    [IsSearchable, Required]

    public string location { get; set; }
    public string Phone { get; set; }

    [Required]
    [IsSearchable]
    [IsFilterable, IsFacetable, Required]
    [IsFilterable, IsFacetable, IsSortable]

    public string Description { get; set; }

    [IsFiltrable, IsSortable, IsSearchable]
    [IsFilterable, IsSortable, IsFacetable]
    [IsFiltrable, IsSortable, Key]
    [IsFilterable, IsSortable, IsSearchable, Required]

    public double Rating { get; set; }

    [IsSearchable, IsFilterable, IsFacetable]
    [IsFilterable, IsSortable, Key]
    [IsFilterable, IsSortable, IsSearchable]
    [IsFilterable, IsSortable, Key, Required]

    public List<string> Cuisines { get; set; }

    [IsFilterable, IsSortable, Key, Required]
    [IsSearchable, IsSortable, IsFacetable]
    [IsFilterable, IsSortable, Key, IsSearchable]
    [IsFilterable, IsFacetable]

    public bool FamilyFriendly { get; set; }
```

Explanation

## Answer Area

```
[SerializePropertyNameAsCamelCase]
public class Restaurant
{
    [Key, IsFilterable]
    public int RestaurantId { get; set; }
    [IsSearchable, IsFilterable, IsSortable]
    public string Name { get; set; }

    [IsSearchable, IsFilterable, IsSortable, IsFacetable]
    [IsFilterable, IsFacetable, Required]
    [IsSearchable]
    [IsSearchable, Required]

    public string location { get; set; }
    public string Phone { get; set; }

    [Required]
    [IsSearchable]
    [IsFilterable, IsFacetable, Required]
    [IsFilterable, IsFacetable, IsSortable]

    public string Description { get; set; }
```

```

[IsFiltrable, IsSortable, IsSearchable]
[IsFilterable, IsSortable, IsFacetable]
[IsFiltrable, IsSortable, Key]
[IsFilterable, IsSortable, IsSearchable, Required]

public double Rating { get; set; }

[IsSearchable, IsFilterable, IsFacetable]
[IsFilterable, IsSortable, Key]
[IsFilterable, IsSortable, IsSearchable]
[IsFilterable, IsSortable, Key, Required]

public List<string> Cuisines { get; set; }

[IsFilterable, IsSortable, Key, Required]
[IsSearchable, IsSortable, IsFacetable]
[IsFilterable, IsSortable, Key, IsSearchable]
[IsFilterable, IsFacetable]

public bool FamilyFriendly { get; set; }

```

Box 1: [IsSearchable,IsFilterable,IsSortable,IsFacetable]

#### Location

Users must be able to search for restaurants by name, description, location, and cuisine.

Users must be able to narrow the results further by location, cuisine, rating, and family-friendliness.

Box 2: [IsSearchable,IsFilterable,IsSortable,Required]

#### Description

Users must be able to search for restaurants by name, description, location, and cuisine.

All words in descriptions must be included in searches.

Box 3: [IsFilterable,IsSortable,IsFacetable]

#### Rating

Users must be able to narrow the results further by location, cuisine, rating, and family-friendliness.

Box 4: [IsSearchable,IsFilterable,IsFacetable]

#### Cuisines

Users must be able to search for restaurants by name, description, location, and cuisine.

Users must be able to narrow the results further by location, cuisine, rating, and family-friendliness.

Box 5: [IsFilterable,IsFacetable]

#### FamilyFriendly

Users must be able to narrow the results further by location, cuisine, rating, and family-friendliness.

#### References:

<https://www.henkboelman.com/azure-search-the-basics/>

**NO.77** You are implementing an Azure API app that uses built-in authentication and authorization functionality.

All app actions must be associated with information about the current user.

You need to retrieve the information about the current user.

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.** HTTP headers
- B.** environment variables
- C.** /.auth/me HTTP endpoint
- D.** /.auth/login endpoint

**Answer:** A C

Explanation

A: After App Service Authentication has been configured, users trying to access your API are prompted to sign in with their organizational account that belongs to the same Azure AD as the Azure AD application used to secure the API. After signing in, you are able to access the information about the current user through the `HttpContext.Current.User` property.

C: While the server code has access to request headers, client code can access GET `/.auth/me` to get the same access tokens ( References:

<https://docs.microsoft.com/en-us/azure/app-service/app-service-web-tutorial-auth-aad>

<https://docs.microsoft.com/en-us/sharepoint/dev/spfx/web-parts/guidance/connect-to-api-secured-with-aad>

**NO.78** 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.

<b>Property</b>	<b>Value</b>
Client certificate location	<ul style="list-style-type: none"> <li>HTTP request header</li> <li>Client cookie</li> <li>HTTP message body</li> <li>URL query string</li> </ul>
Encoding type	<ul style="list-style-type: none"> <li>HTML</li> <li>URL</li> <li>Unicode</li> <li>Base64</li> </ul>

**Answer:**

<b>Property</b>	<b>Value</b>
Client certificate location	<ul style="list-style-type: none"> <li>HTTP request header</li> <li>Client cookie</li> <li>HTTP message body</li> <li>URL query string</li> </ul>
Encoding type	<ul style="list-style-type: none"> <li>HTML</li> <li>URL</li> <li>Unicode</li> <li>Base64</li> </ul>

Explanation

Property	Value
Client certificate location	<div style="border: 1px solid black; padding: 5px; width: fit-content; margin-left: auto; margin-right: auto;">         HTTP request header          Client cookie          HTTP message body  <b>URL query string</b> </div>
Encoding type	<div style="border: 1px solid black; padding: 5px; width: fit-content; margin-left: auto; margin-right: auto;">         HTML  <b>URL</b>          Unicode          Base64       </div>

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.

References:

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

**NO.79** 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

**Answer:** B

**NO.80** 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 question, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

Margie's Travel is an international travel and bookings management service. The company is expanding into restaurant bookings. You are tasked with implementing Azure Search for the restaurants listed in their solution.

You create the index in Azure Search.

You need to import the restaurant data into the Azure Search service by using the Azure Search .NET SDK.

Solution:

1. Create a SearchIndexClient object to connect to the search index.
2. Create a DataContainer that contains the documents which must be added.
3. Create a DataSource instance and set its Container property to the DataContainer.
4. Call the Documents.Suggest method of the SearchIndexClient and pass the DataSource.

Does the solution meet the goal?

**A.** Yes

**B.** No

**Answer:** B

Explanation

Use the following method:

- \* - Create a SearchIndexClient object to connect to the search index
- \* - Create an IndexBatch that contains the documents which must be added.
- \* - Call the Documents.Index method of the SearchIndexClient and pass the IndexBatch.

References:

<https://docs.microsoft.com/en-us/azure/search/search-howto-dotnet-sdk>

**NO.81** 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")]
        ITableWriter tableBindings,
        ILogger log)
    {
        log.Info($"Processing Order: {myQueueItem.Id}");
        log.Info($"Queue Insertion Time: {myQueueItem.InsertionTime}");
        log.Info($"Queue Expiration Time: {myQueueItem.ExpirationTime}");
        tableBindings.Add(new OrderItem(myQueueItem.AsString));
    }

    [FunctionName("ProcessOrders_Failed")]
    public static void ProcessFailedOrders([QueueTrigger("incoming-orders-failed")] CloudQueueMessage myQueueItem, ITableWriter log)
    {
        log.Error($"Failed to process order: {myQueueItem.AsString}");
    }
}
```

NOTE: Each correct selection is worth one point.

	Yes	No
The code will log the time that the order was processed from the queue.	<input type="radio"/>	<input type="radio"/>
When the ProcessOrders function fails, the function will retry up to five times for a given order, including the first try.	<input type="radio"/>	<input type="radio"/>
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.	<input type="radio"/>	<input type="radio"/>
The ProcessOrders function will output the order to an Orders table in Azure Table Storage.	<input type="radio"/>	<input type="radio"/>

**Answer:**

	Yes	No
The code will log the time that the order was processed from the queue.	<input type="radio"/>	<input checked="" type="radio"/>
When the ProcessOrders function fails, the function will retry up to five times for a given order, including the first try.	<input checked="" type="radio"/>	<input type="radio"/>
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.	<input checked="" type="radio"/>	<input type="radio"/>
The ProcessOrders function will output the order to an Orders table in Azure Table Storage.	<input checked="" type="radio"/>	<input type="radio"/>

**Explanation**

	Yes	No
The code will log the time that the order was processed from the queue.	<input type="radio"/>	<input checked="" type="radio"/>
When the ProcessOrders function fails, the function will retry up to five times for a given order, including the first try.	<input type="radio"/>	<input checked="" type="radio"/>
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.	<input type="radio"/>	<input checked="" type="radio"/>
The ProcessOrders function will output the order to an Orders table in Azure Table Storage.	<input checked="" type="radio"/>	<input type="radio"/>

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

References:

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

**NO.82** Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution. After you answer a question in this section, you will NOT be able to return to it. As a result these questions will not appear in the review screen.

Margie's Travel is an international travel and bookings management service. The company is expanding into restaurant bookings. You are tasked with implementing Azure Search for the restaurants listed in their solution.

You create the index in Azure Search.

You need to import the restaurant data into the Azure Search service by using the Azure Search .NET SDK.

Solution:

1. Create a SearchIndexClient object to connect to the search index.
2. Create a DataContainer that contains the documents which must be added.
3. Create a DataSource instance and set its Container property to the DataContainer
- 4 Call the Documents.Suggest method of the SearchIndexClient and pass the DataSource.

Does the solution meet the goal?

**A.** Yes

**B.** No

**Answer:** B

**NO.83** 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 command from the list of commands to the answer area and arrange them in the correct order.

**Azure CLI commands**

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

**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 $dockerHibContainerPath
--name $appName
--resource-group fourthCoffeePublicWebResourceGroup
```

**Answer:****Azure CLI commands**

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

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



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

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

**Explanation**

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

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

```
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
```

Step 1: #bin/bash

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

Step 2: az webapp config hostname add

The webapp-name is used when the webapp is created in step 3.

Step 3: 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 : 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.

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

References:

<https://docs.microsoft.com/en-us/azure/app-service/containers/tutorial-custom-docker-image>

**NO.84** You are developing a data storage solution for a social networking app.

The solution requires a mobile app that stores user information using Azure Table Storage.

You need to develop code that can insert multiple sets of user information.

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

NOTE: Each correct selection is worth one point.

```
CloudStorageAccount storageAccount = CloudStorageAccount.Parse(
    CloudConfigurationManager.GetSetting("StorageConnectionString"));
CloudTableClient tableClient = storageAccount.CreateCloudTableClient();
CloudTable table = tableClient.GetTableReference("clients");
Table.CreateIfNotExists();
```

	op = new <span style="color: red;">()</span> ;
TableOperation TableBatchOperaton TableEntity TableQuery	TableOperation TableBatchOperaton TableEntity TableQuery
. . .	
table. <span style="color: red;">(op)</span> ;	
ExecuteBatch Execute Insert InsertOrMerge	

**Answer:**

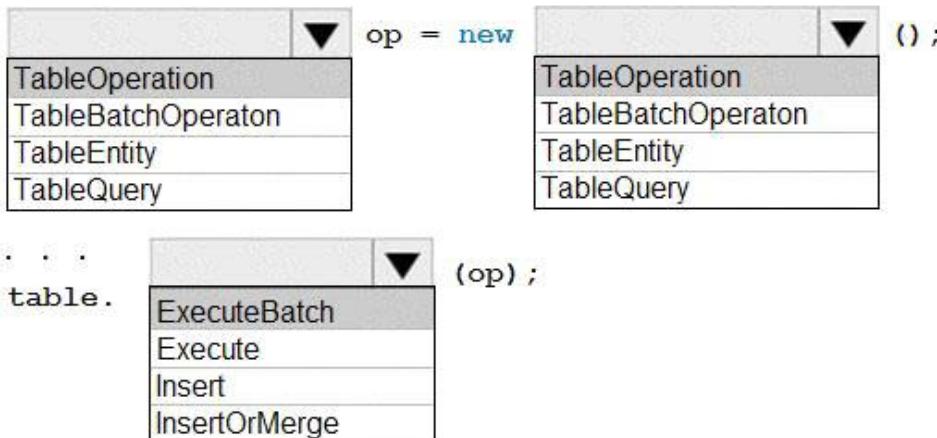
```
CloudStorageAccount storageAccount = CloudStorageAccount.Parse(
    CloudConfigurationManager.GetSetting("StorageConnectionString"));
CloudTableClient tableClient = storageAccount.CreateCloudTableClient();
CloudTable table = tableClient.GetTableReference("clients");
Table.CreateIfNotExists();
```

	op = new <span style="color: red;">()</span> ;
TableOperation_   TableBatchOperaton TableEntity TableQuery	TableOperation   TableBatchOperaton TableEntity TableQuery
. . .	
table. <span style="color: red;">(op)</span> ;	
ExecuteBatch	
Execute Insert InsertOrMerge	

Explanation

## Answer Area

```
CloudStorageAccount storageAccount = CloudStorageAccount.Parse(
    CloudConfigurationManager.GetSetting("StorageConnectionString"));
CloudTableClient tableClient = storageAccount.CreateCloudTableClient();
CloudTable table = tableClient.GetTableReference("clients");
Table.CreateIfNotExists();
```



Box 1, Box 2: TableBatchOperation

Create the batch operation.

`TableBatchOperation op = new TableBatchOperation();`

Box 3: ExecuteBatch

/ Execute the batch operation.

`table.ExecuteBatch(op);`

Note: You can insert a batch of entities into a table in one write operation. Some other notes on batch operations:

You can perform updates, deletes, and inserts in the same single batch operation.

A single batch operation can include up to 100 entities.

All entities in a single batch operation must have the same partition key.

While it is possible to perform a query as a batch operation, it must be the only operation in the batch.

References:

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

**NO.85** 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.

**Actions****Answer Area**

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 files from cache if the TTL has not expired.

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

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

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

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 files from cache if the TTL has not expired.

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

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



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

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

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

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

**Explanation**

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

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

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

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

Step 1: A user requests the image..

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

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

If no edge servers in the POP have the file in their cache, the POP requests the file from the origin server. The origin server can be an Azure Web App, Azure Cloud Service, Azure Storage account, or any publicly accessible web server.

Step 3: The origin server returns the..

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

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

Step 4: Subsequent requests for..

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

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

References:

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

**NO.86** 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.

PDFDUMPS.COM

**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:

- Total
- Maximum
- Average
- Count

\* Threshold:

Action:

\* Operation:

- Increase count by
- Increase count to
- Decrease count by
- Decrease count to

\* Instance count:

\* Cool down (minutes):

**Answer:**

PDFDUMPS.COM

**Answer Area**

**Scale rule**

Metric source

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

Resource type

Service Bus Namespaces

Resource

MessageQueue1103

\* Queues

itemqueue

Criteria

\* Metric name

- Message Count
- Active Message Count

1 minute time grain

\* Time grain statistic

- Total
- Maximum
- Average
- Count

Greater than

Greater than or equal to

Less than

Less than or equal to

\* Threshold

1000

Action

\* Operation

- Increase count by
- Increase count to
- Decrease count by
- Decrease count to

\* Instance count

1

\* Cool down (minutes)

5

## Explanation

**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 ⓘ

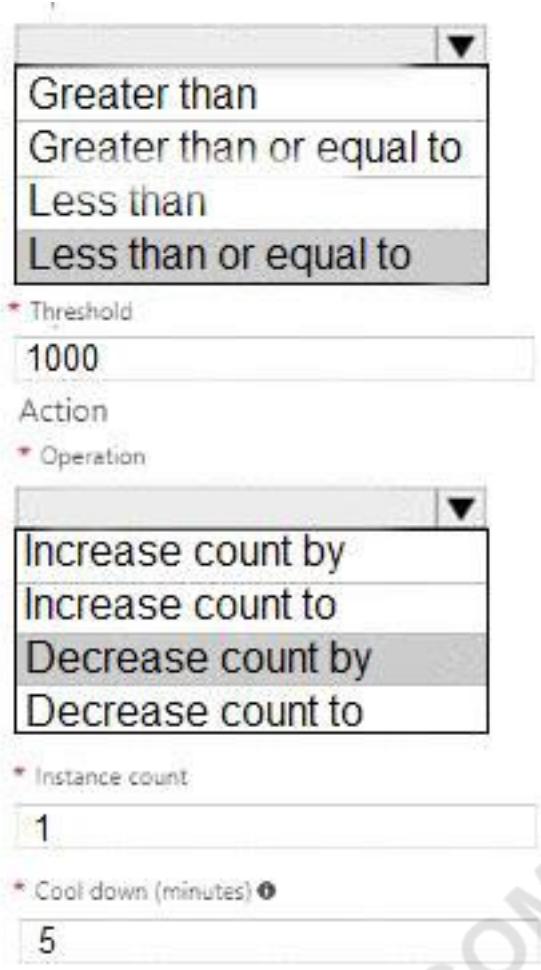
Total

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

**NO.87** 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.

## Filter types

## Answer Area

Subscription	Filter type
SQLFilter	FutureOrders
CorrelationFilter	HighPriorityOrders
No Filter	InternationalOrders
	HighQuantityOrders
	AllOrders

**Answer:**

**Filter types**

SQLFilter

CorrelationFilter

No Filter

**Answer Area****Subscription**

FutureOrders

HighPriorityOrders

InternationalOrders

HighQuantityOrders

AllOrders

**Filter type**

SQLFilter

CorrelationFilter

SQLFilter

SQLFilter

No Filter

Explanation

**Answer Area****Subscription**

FutureOrders

HighPriorityOrders

InternationalOrders

HighQuantityOrders

AllOrders

**Filter type**

SQLFilter

CorrelationFilter

SQLFilter

SQLFilter

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.

References:

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

**NO.88** You plan to create a Docker image that runs as 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 setupScript.ps1 when the container is built.
- \* Run ContosoApp.dll when the container starts.

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

Which four 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.

**Commands****Answer Area**

```
RUN powershell ./setupScript.ps1
CMD ["dotnet", "ContosoApp.dll"]
```

```
EXPOSE ./ContosoApp/ /apps/ContosoApp
```

```
COPY ./
```



```
FROM microsoft/aspnetcore:2.0
```

```
WORKDIR /apps/ContosoApp
```

```
CMD powershell ./setupScript.ps1
ENTRYPOINT ["dotnet", "ContosoApp.dll"]
```

**Answer:****Commands****Answer Area**

```
RUN powershell ./setupScript.ps1
CMD ["dotnet", "ContosoApp.dll"]
```

```
WORKDIR /apps/ContosoApp
```

```
EXPOSE ./ContosoApp/ /apps/ContosoApp
```

```
COPY ./
```



```
FROM microsoft/aspnetcore:2.0
```

```
EXPOSE ./ContosoApp/ /apps/ContosoApp
```

```
WORKDIR /apps/ContosoApp
```

```
CMD powershell ./setupScript.ps1
ENTRYPOINT ["dotnet", "ContosoApp.dll"]
```

```
CMD powershell ./setupScript.ps1
ENTRYPOINT ["dotnet", "ContosoApp.dll"]
```

**Explanation**

```
WORKDIR /apps/ContosoApp
```

```
COPY ./
```

```
EXPOSE ./ContosoApp/ /apps/ContosoApp
```

```
CMD powershell ./setupScript.ps1  
ENTRYPOINT ["dotnet", "ContosoApp.dll"]
```

Step 1: WORKDIR /apps/ContosoApp

Step 2: COPY ./

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

Step 3: EXPOSE ./ContosoApp/ /app/ContosoApp

Step 4: CMD powershell ./setupScript.ps1

ENTRYPOINT ["dotnet", "ContosoApp.dll"]

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

- \* Call setupScript.ps1 when the container is built.
- \* Run ContosoApp.dll when the container starts.

References:

<https://docs.microsoft.com/en-us/azure/app-service/containers/tutorial-custom-docker-image>

**NO.89** You develop a solution that uses an Azure SQL Database to store user information for a mobile app.

The app stores sensitive information about users.

You need to hide sensitive information from developers that query the data for the mobile app.

Which three items must you identify when configuring dynamic data masking? Each correct answer presents a part of the solution.

NOTE: Each correct selection is worth one point.

- A.** Column
- B.** Table
- C.** Trigger
- D.** Index
- E.** Schema

**Answer:** A B E

Explanation

In the Dynamic Data Masking configuration page, you may see some database columns that the

recommendations engine has flagged for masking. In order to accept the recommendations, just click Add Mask for one or more columns and a mask is created based on the default type for this column. You can change the masking function by clicking on the masking rule and editing the masking field format to a different format of your choice.

The screenshot shows the 'Dynamic Data Masking' interface for the 'demo\_database'. The top bar includes a save icon, a discard icon, and an 'Add Mask' button. A message box displays a warning: 'Downlevel clients require the use of Security Enabled Connection Strings.' Below this, the 'Masking Rules' section indicates 'You haven't created any masking rules.' A note below states 'SQL users excluded from masking (administrators are always excluded)' with a checked checkbox. The main area, highlighted with a red border, lists 'Recommended fields to mask' for the 'Customer' and 'CustomerAddress' tables in the 'SalesLT' schema. For each row, there is a 'COLUMN' column (FirstName, LastName, EmailAddress, Phone, AddressID) and an 'ADD MASK' button.

SCHEMA	TABLE	COLUMN	
SalesLT	Customer	FirstName	<b>ADD MASK</b>
SalesLT	Customer	LastName	<b>ADD MASK</b>
SalesLT	Customer	EmailAddress	<b>ADD MASK</b>
SalesLT	Customer	Phone	<b>ADD MASK</b>
SalesLT	CustomerAddress	AddressID	<b>ADD MASK</b>

#### References:

<https://docs.microsoft.com/en-us/azure/sql-database/sql-database-dynamic-data-masking-get-started>

started-portal

**NO.90** You are developing a software solution for an autonomous transportation system. The solution uses large data sets and Azure Batch processing to simulate navigation sets for entire fleets of vehicles.

You need to create compute nodes for the solution on Azure Batch.

What should you do?

- A.** In the Azure portal, create a Batch account.
- B.** In a .NET method, call the method: BatchClient.PoolOperations.CreatePool
- C.** In Python, implement the class: JobAddParameter
- D.** In Python, implement the class: TaskAddParameter

**Answer:** B

Explanation

A Batch job is a logical grouping of one or more tasks. A job includes settings common to the tasks, such as priority and the pool to run tasks on. The app uses the BatchClient.JobOperations.CreateJob method to create a job on your pool.

**NO.91** You develop an Azure web app. You monitor performance of the web app by using Application Insights. You need to ensure the cost for Application Insights does not exceed a preset budget. What should you do?

- A.** Implement ingestion sampling using the Azure portal.
- B.** Set a daily cap for the Application Insights instance.
- C.** Implement adaptive sampling using the Azure portal.
- D.** Implement adaptive sampling using the Application Insights SDK.
- E.** Implement ingestion sampling using the Application Insights SDK.

**Answer:** D

Explanation

Sampling is an effective way to reduce charges and stay within your monthly quota.

You can set sampling manually, either in the portal on the Usage and estimated costs page; or in the ASP.NET SDK in the .config file; or in the Java SDK in the ApplicationInsights.xml file, to also reduce the network traffic.

Adaptive sampling is the default for the ASP.NET SDK. Adaptive sampling automatically adjusts to the volume of telemetry that your app sends. It operates automatically in the SDK in your web app so that telemetry traffic on the network is reduced.

References:

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

**NO.92** 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 SequenceNumber property.
- B. Assign the value of the hazard message SequenceNumber property to the DeliveryCount property.
- C. Assign the value of the hazard message MessageId property to the DeliveryCount property.
- D. Assign the value of the hazard message SessionID property to the ReplyToSessionId property.
- E. Assign the value of the hazard message MessageId property to the SequenceNumber property.

**Answer:** A B

**NO.93** 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.

### Azure CLI Commands

### Answer Area

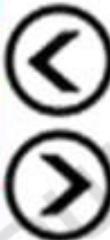
az group create

az group update

az webapp update

az webapp create

az appservice plan create



**Answer:**

**Azure CLI Commands**

az group create

az group update

az webapp update

az webapp create

az appservice plan create

**Answer Area**

az group create

az appservice plan create

az webapp create

**Explanation**

az group create

az appservice plan create

az webapp 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 with a unique app name, and <docker-ID> with your Docker ID.

References:

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

**NO.94** Your company is migrating applications to Azure. The IT department must allow internal developers to communicate with Microsoft support.

The service agents of the IT department must only have view resources and create support ticket permissions to all subscriptions. A new custom role must be created by reusing a default role definition and changing the permissions.

You need to create the custom role.

To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Item	Value
Powershell command	<pre>Get-AzureRmRoleDefinition-Name "Reader"   ConvertTo-Json   Out-File C:\SupportRole.json Get-AzureRmRoleDefinition-Name "Operator"   ConvertTo-Json   Out-File C:\SupportRole.json Set-AzureRmRoleDefinition-Name "Reader"   Input-File C:\SupportRole.json Set-AzureRmRoleDefinition Input-File C:\SupportRole.json</pre>
Actions section	<div style="border: 1px solid black; padding: 5px;"> <pre>"/read*, *Microsoft.Support/*" "/read* */*, *Microsoft.Support/* */*</pre> </div>

### Answer:

Item	Value
Powershell command	<pre>Get-AzureRmRoleDefinition-Name "Reader"   ConvertTo-Json   Out-File C:\SupportRole.json Get-AzureRmRoleDefinition-Name "Operator"   ConvertTo-Json   Out-File C:\SupportRole.json Set-AzureRmRoleDefinition-Name "Reader"   Input-File C:\SupportRole.json Set-AzureRmRoleDefinition Input-File C:\SupportRole.json</pre>
Actions section	<div style="border: 1px solid black; padding: 5px;"> <pre>"/read*, *Microsoft.Support/*" "/read* */*, *Microsoft.Support/* */*</pre> </div>

### Explanation

Item	Value
Powershell command	<pre>Get-AzureRmRoleDefinition-Name "Reader"   ConvertTo-Json   Out-File C:\SupportRole.json Get-AzureRmRoleDefinition-Name "Operator"   ConvertTo-Json   Out-File C:\SupportRole.json Set-AzureRmRoleDefinition-Name "Reader"   Input-File C:\SupportRole.json Set-AzureRmRoleDefinition Input-File C:\SupportRole.json</pre>
Actions section	<div style="border: 1px solid black; padding: 5px;"> <pre>"/read*, *Microsoft.Support/*" "/read* */*, *Microsoft.Support/* */*</pre> </div>

Box 1: Set-AzureRmRoleDefinition Input-File C:\SupportRole.json

The Set-AzureRmRoleDefinition cmdlet updates an existing custom role in Azure Role-Based Access Control.

Provide the updated role definition as an input to the command as a JSON file or a PSRoleDefinition object.

The role definition for the updated custom role MUST contain the Id and all other required properties of the role even if they are not updated: DisplayName, Description, Actions, AssignableScope Box 2:  
 /\*/read/\* Microsoft.Support/\* Microsoft.Support/\* Create and manage support tickets  
 "Microsoft.Support" role definition azure

**NO.95** 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.

```
gitrepo=https://github.com/Contoso/webapp
webappname=businesswebapp
resourcegroupname=BusinessAppResourceGroup
az group create --location centralus --name $resourcegroupname
az create --name $webappname --resource-group $resourcegroupname
--sku S3
az create --name $webappname --resource-group $resourcegroupname
\ --plan $webapname
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
```

az	group
	webapp
	appservice plan
	webapp deployment slot
	webapp deployment source

az	group
	webapp
	appservice plan
	webapp deployment slot
	webapp deployment source

az	group
	webapp
	appservice plan
	webapp deployment slot
	webapp deployment source

az	group
	webapp
	appservice plan
	webapp deployment slot
	webapp deployment source

az	group
	webapp
	appservice plan
	webapp deployment slot
	webapp deployment source

**Answer:**

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

```
az group create --location centralus --name $resourcegroupname
az webapp create --name $webappname --resource-group $resourcegroupname --sku S3
az appservice plan create --name $webappname --resource-group $resourcegroupname \ --plan $webappname
az webapp deployment slot create --name $webappname --resource-group $resourcegroupname \ --slot staging
az config --name $webappname --resource-group $resourcegroupname \ --slot staging --repo-url $gitrepo --branch master --manual-integration
```

az

group  
webapp  
appservice plan  
webapp deployment slot  
webapp deployment source

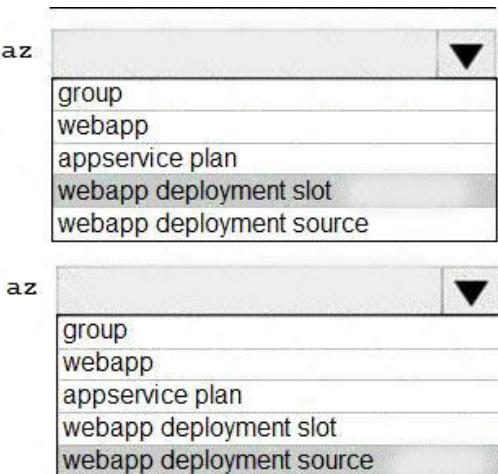
## Explanation

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

```
az group create --location centralus --name $resourcegroupname
az webapp create --name $webappname --resource-group $resourcegroupname
--sku S3
az appservice plan create --name $webappname --resource-group $resourcegroupname
\ --plan $webappname
az webapp create --name $webappname --resource-group $resourcegroupname
\ --slot staging
az config --name $webappname --resource-group $resourcegroupname
\ --slot staging --repo-url
$gitrepo --branch master --manual-integration

az group create --location centralus --name $resourcegroupname
az webapp create --name $webappname --resource-group $resourcegroupname
--sku S3
az appservice plan create --name $webappname --resource-group $resourcegroupname
\ --plan $webappname
az webapp create --name $webappname --resource-group $resourcegroupname
\ --slot staging
az config --name $webappname --resource-group $resourcegroupname
\ --slot staging --repo-url
$gitrepo --branch master --manual-integration

az group create --location centralus --name $resourcegroupname
az webapp create --name $webappname --resource-group $resourcegroupname
--sku S3
az appservice plan create --name $webappname --resource-group $resourcegroupname
\ --plan $webappname
az webapp 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 References:
https://docs.microsoft.com/en-us/azure/app-service/scripts/cli-deploy-staging-environment
```

**NO.96** You are developing an internal website for employees to view sensitive data. The website uses Azure Active Directory (AAD) for authentication. You need to implement multifactor authentication for the website.

What should you do? Each correct answer presents part of the solution.

NOTE: Each correct selection is worth one point.

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

**Answer:** A E

Explanation

References:

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

**NO.97** You develop a serverless application using several Azure Functions. These functions connect to data from within the code.

You want to configure tracing for an Azure Function App project.

You need to change configuration settings in the host.json file.

Which tool should you use?

- A.** Azure portal
- B.** Azure PowerShell
- C.** Azure Functions Core Tools (Azure CLI)
- D.** Visual Studio

**Answer:** A

Explanation

The function editor built into the Azure portal lets you update the function.json file and the code file for a function. The host.json file, which contains some runtime-specific configurations, is in the root folder of the function app.

```
FunctionApp
| - host.json
| - Myfirstfunction
| | - function.json
| | - ...
| - mysecondfunction
| | - function.json
| | - ...
| - SharedCode
| - bin
```

References:

<https://docs.microsoft.com/en-us/azure/azure-functions/functions-reference#fileupdate>

**NO.98** You are developing a microservices solution. You plan to deploy the solution to a multi-node 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 component to the requirements. Each component may be used, once, more than once, or not at all. You need to drag the split bar between panes or scroll to view content.

NOTE: Each correct selection is worth one point.

The screenshot shows a software interface with three main sections: Components, Answer area, and Action.

- Components:** A vertical list of items: Helm, Draft, Brigade, KubeCtl, Ingress Controller, CoreDNS, and Virtual Kubelet. The first four are highlighted with green dashed boxes, while the last three are highlighted with red dashed boxes.
- Answer area:** A section containing five horizontal dots followed by a cursor icon.
- Action:** A table with two columns:
 

Action	Component
Deploy solution.	Component
View cluster and external IP addressing.	Component
Implement a single, public IP endpoint that is routed to multiple microservices.	Component

**Answer:**

This screenshot is identical to the one above, showing the same interface with components, answer area, and action table. The highlighting of components (Helm, Draft, Brigade, KubeCtl) is identical to the first screenshot, but the Ingress Controller, CoreDNS, and Virtual Kubelet components are now highlighted with red dashed boxes instead of green.

**NO.99** 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 in real time when specific requirements are met, excluding backup blob copies.

What should you do?

- Download the blob to a virtual machine and then upload the blob to Container2.
- Run the Azure PowerShell command Start-AzureStorageBlobCopy.
- Copy blobs to Container2 by using the Put Blob operation of the Blob Service REST API.
- Use AzCopy with the Snapshot switch blobs to Container2.

**Answer:** B

Explanation

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.

## References:

<https://docs.microsoft.com/en-us/powershell/module/azure.storage/start-azurestorageblobcopy?view=azurermps>

**NO.100** A company develops a series of mobile games. All games use a single leaderboard service.

You have the following requirements:

- \* Code should 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 and Id based on the series title.

You have the following code. (Line numbers are included for reference only.) You store customer information in an Azure Cosmos database. The following data already exists in the database:

You develop the following code. (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.Generate.And, TableQuery.GenerateFilterCondition("Email", QueryComparisons.Equal, "Smith")
06      TableOperators.And, TableQuery.GenerateFilterCondition("Email", QueryComparisons.Equal,
07      "ssmith@contoso.com")
08    ));
09  await table.ExecuteQuerySegmentedAsync<CustomerEntity>(query, null);

```

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

NOTE: Each correct selection is worth one point.

	Yes	No
The code will work with Cosmos DB.	<input type="radio"/>	<input type="radio"/>
The save score function will update and replace a record if one already exists with the same playerId and gameId.	<input type="radio"/>	<input type="radio"/>
The data for the game will be automatically partitioned.	<input type="radio"/>	<input type="radio"/>
This code will store the values for the gameId and playerId parameters in the database.	<input type="radio"/>	<input type="radio"/>

**Answer:**

	Yes	No
The code will work with Cosmos DB.	<input checked="" type="checkbox"/>	<input type="radio"/>
The save score function will update and replace a record if one already exists with the same playerId and gameId.	<input type="radio"/>	<input checked="" type="checkbox"/>
The data for the game will be automatically partitioned.	<input type="radio"/>	<input checked="" type="checkbox"/>
This code will store the values for the gameId and playerId parameters in the database.	<input checked="" type="checkbox"/>	<input type="radio"/>

Explanation

	Yes	No
The code will work with Cosmos DB.	<input checked="" type="radio"/>	<input type="radio"/>
The save score function will update and replace a record if one already exists with the same playerId and gameId.	<input type="radio"/>	<input checked="" type="radio"/>
The data for the game will be automatically partitioned.	<input type="radio"/>	<input checked="" type="radio"/>
This code will store the values for the gameId and playerId parameters in the database.	<input checked="" type="radio"/>	<input type="radio"/>

Box 1: Yes

Code for CosmosDB, example:

```
// Parse the connection string and return a reference to the storage account.
CloudStorageAccount storageAccount = CloudStorageAccount.Parse(
    CloudConfigurationManager.GetSetting("StorageConnectionString"));
// Create the table client.
CloudTableClient tableClient = storageAccount.CreateCloudTableClient();
// Retrieve a reference to the table.
CloudTable table = tableClient.GetTableReference("people");
// Create the TableOperation object that inserts the customer entity.
TableOperation insertOperation = TableOperation.Insert(customer1);
```

Box 2: No

A new record will always be added as TableOperation.Insert is used, instead of TableOperation.InsertOrReplace.

Box 3: No

No partition key is used.

Box 4: Yes

References:

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

**NO.101** 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.

Consistency levels	Answer Area
Strong	Return the most recent patient status.
Consistent Prefix	Return health monitoring data that is no less than one version behind.
Eventual	After patient is discharged and all changes are assessed, retrieve the correct billing data with the final charges

**Answer:**

Consistency levels	Answer Area
Strong	Return the most recent patient status.
Consistent Prefix	Return health monitoring data that is no less than one version behind.
Eventual	After patient is discharged and all changes are assessed, retrieve the correct billing data with the final charges

**Explanation**

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 changes 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.

**NO.102** You have a web app named MainApp. You are developing a triggered App Service background task by using the WebJobs SDK. This task automatically invokes a function code whenever any new data is received in a queue.

You need to configure the services.

Which service should you use for each scenario? To answer, drag the appropriate services to the correct scenarios. Each service 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.

Services	Scenario	Service
Logic Apps	Process a queue data item.	
WebJobs	Manage all code segments from the same DevOps environment.	
Flow		

**Answer:**

Services	Scenario	Service
Logic Apps	Process a queue data item.	
WebJobs	Manage all code segments from the same DevOps environment.	WebJobs
Flow		Flow

Explanation

Scenario	Service
Process a queue data item.	WebJobs
Manage all code segments from the same DevOps environment.	Flow

#### Box 1: WebJobs

A WebJob is a simple way to set up a background job, which can process continuously or on a schedule.

WebJobs differ from a cloud service as it gives you get less fine-grained control over your processing environment, making it a more true PaaS service.

#### Box 2: Flow

**NO.103** You have an Azure App Services Web App. Azure SQL Database instance. Azure Storage Account and an Azure Redis Cache instance in a resource group.

A developer must be able to publish code to the web app. You must grant the developer the Contribute role to the web app. You need to grant the role.

What two commands can you use? Each correct answer presents a complete solution.

NOTE: Each correct selection is worth one point.

- A.** New-AzureRmRoleAssignment
- B.** az role assignment create
- C.** az role definition create
- D.** New-AzureRmRoleDefinition

**Answer:** A B

Explanation

References:

<https://docs.microsoft.com/en-us/cli/azure/role/assignment?view=azure-cli-latest#az-role-assignment-create>

<https://docs.microsoft.com/en-us/powershell/module/azurerm.resources/new-azurermroleassignment?view=azure>

**NO.104** 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.** validate-jwt
- B.** jsonp
- C.** authentication-certificate
- D.** check-header

**Answer:** A

Explanation

Add the validate-jwt policy to validate the OAuth token for every incoming request.

**NO.105** You are developing a mobile instant messaging app for a company.

The mobile app must meet the following requirements:

- \* Support offline data sync.
- \* Update the latest messages during normal sync cycles.

You need to implement Offline Data Sync.

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

NOTE: Each correct selection is worth one point.

- A.** Retrieve records from Offline Data Sync on every call to the PullAsync method.
- B.** Retrieve records from Offline Data Sync using an Incremental Sync.
- C.** Push records to Offline Data Sync using an Incremental Sync.
- D.** Return the updatedAt column from the Mobile Service Backend and implement sorting by using the column.
- E.** Return the updatedAt column from the Mobile Service Backend and implement sorting by the message id.

**Answer:** B E

### Explanation

B: Incremental Sync: the first parameter to the pull operation is a query name that is used only on the client. If you use a non-null query name, the Azure Mobile SDK performs an incremental sync. Each time a pull operation returns a set of results, the latest updatedAt timestamp from that result set is stored in the SDK local system tables. Subsequent pull operations retrieve only records after that timestamp.

E (not D): To use incremental sync, your server must return meaningful updatedAt values and must also support sorting by this field. However, since the SDK adds its own sort on the updatedAt field, you cannot use a pull query that has its own orderBy clause.

### References:

<https://docs.microsoft.com/en-us/azure/app-service-mobile/app-service-mobile-offline-data-sync>

**NO.106** You are developing a project management service by using ASP.NET. The service hosts conversations, files, to-do lists, and a calendar that users can interact with at any time.

The application uses Azure Search for allowing users to search for keywords in the project data. You need to implement code that creates the object which is used to create indexes in the Azure Search service.

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

NOTE: Each correct selection is worth one point.

- A. SearchService
- B. SearchIndexClient
- C. SearchServiceClient
- D. SearchCredentials

**Answer:** B C

### Explanation

The various client libraries define classes like Index, Field, and Document, as well as operations like Indexes.Create and Documents.Search on the SearchServiceClient and SearchIndexClient classes.

### Example:

The sample application we'll be exploring creates a new index named "hotels", populates it with a few documents, then executes some search queries. Here is the main program, showing the overall flow:

```
/ This sample shows how to delete, create, upload documents and query an index static void
```

```
Main(string[] args)
```

```
{
```

```
IConfigurationBuilder builder = new ConfigurationBuilder().AddJsonFile("appsettings.json");
```

```
IConfigurationRoot configuration = builder.Build(); SearchServiceClient serviceClient =
```

```
CreateSearchServiceClient(configuration); Console.WriteLine("{0}", "Deleting index...\n");
```

```
DeleteHotelsIndexIfExists(serviceClient); Console.WriteLine("{0}", "Creating index...\n");
```

```
CreateHotelsIndex(serviceClient); ISearchIndexClient indexClient =
```

```
serviceClient.Indexes.GetClient("hotels"); References:
```

<https://docs.microsoft.com/en-us/azure/search/search-howto-dotnet-sdk>

**NO.107** A company is developing a mobile app for field service employees using Azure App Service Mobile Apps as the backend.

The company's network connectivity varies throughout the day. The solution must support offline use and synchronize changes in the background when the app is online app.

You need to implement the solution.

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.

```
var client = new MobileServiceClient("MOBILE_APP_URL");
var store = new MobileServiceSQLiteStore
(Constants.OfflineDbPath);
store.DefineTable<TodoItem>();
await client.SyncContext.InitializeAsync(store);
```

```
var todoTable = client.GetSyncTable<TodoItem>();
var todoTable = client.GetTable<TodoItem>();
var todoTable = client.SyncTable;
var todoTable = client.Table;

await client.SyncContext.PushAsync();
```

```
await todoTable.PullAsync("allTodos", todoTable.CreateQuery());
await todoTable.UpdateAsync();
todoTable.PullAsync("allTodos", todoTable.CreateQuery());
todoTable.UpdateAsync();
```

**Answer:**

```

var client = new MobileServiceClient("MOBILE_APP_URL");
var store = new MobileServiceSQLiteStore
(Constants.OfflineDbPath);
store.DefineTable<TodoItem>();
await client.SyncContext.InitializeAsync(store);

```

```

var todoTable = client.GetSyncTable<TodoItem>();
var todoTable = client.GetTable<TodoItem>();
var todoTable = client.SyncTable;
var todoTable = client.Table;

await client.SyncContext.PushAsync();

```

```

await todoTable.PullAsync("allTodos", todoTable.CreateQuery());
await todoTable.UpdateAsync();
todoTable.PullAsync("allTodos", todoTable.CreateQuery());
todoTable.UpdateAsync();

```

Explanation

```

var client = new MobileServiceClient("MOBILE_APP_URL");
var store = new MobileServiceSQLiteStore
(Constants.OfflineDbPath);
store.DefineTable<TodoItem>();
await client.SyncContext.InitializeAsync(store);

```

```

var todoTable = client.GetSyncTable<TodoItem>();
var todoTable = client.GetTable<TodoItem>();
var todoTable = client.SyncTable;
var todoTable = client.Table;

await client.SyncContext.PushAsync();

```

```

await todoTable.PullAsync("allTodos", todoTable.CreateQuery());
await todoTable.UpdateAsync();
todoTable.PullAsync("allTodos", todoTable.CreateQuery());
todoTable.UpdateAsync();

```

Box 1: var todoTable = client.GetSyncTable<TodoItem>()

To setup offline access, when connecting to your mobile service, use the method GetSyncTable

instead of GetTable (example):

```
IMobileServiceSyncTable todoTable = App.MobileService.GetSyncTable(); / Box 2: await
todoTable.PullAsync("allTodos", todo.Table.CreateQuery()); Your app should now use
IMobileServiceSyncTable (instead of IMobileServiceTable) for CRUD operations.
```

This will save changes to the local database and also keep a log of the changes. When the app is ready to synchronize its changes with the Mobile Service, use the methods PushAsync and PullAsync (example):

```
await App.MobileService.SyncContext.PushAsync();
await todoTable.PullAsync();
```

References:

<https://azure.microsoft.com/es-es/blog/offline-sync-for-mobile-services/>

**NO.108** You are creating a CLI script that creates an Azure web app related services in Azure App Service. The web app uses the following variables:

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

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

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

NOTE: Each correct selection is worth one point.

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

`az webapp create - --name $webappname - --resource-group myResourceGroup - --sku FREE`

`az webapp create`

`az appservice plan create`

`az webapp deployment`

`az group delete`

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

`az webapp create`

`az appservice plan create`

`az webapp deployment`

`az group delete`

`- --repo-url $gitrepo - --branch master - --manual-integration`

`git clone $gitrepo`

`- --plan $webappname`

`source config - --name $webappname`

`az webapp create`

`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`

**Answer:**

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

## Explanation

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

#### 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

References:

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

**NO.109** A company uses Azure SQL Database to store data for an app. The data includes sensitive information.

You need to implement measures that allow only members of the managers group to see sensitive information.

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

NOTE: Each correct selection is worth one point.

- A. Include the managers group.
- B. Exclude the managers group.
- C. Exclude the administrators group.
- D. Navigate to the following URL:

```
PUT https://management.azure.com/subscriptions/00000000-1111-2222-3333-444444444444
/resourceGroups/rg01/providers/Microsoft.Sql/servers/server01/databases/customers
/transparentDataEncryption/current?api-version=2014-04-01
```

- E. Run the following Azure PowerShell command:

```
New-AzureRmSqlDatabaseDataMaskingRule -SchemaName "dbo" -TableName "customers" '
-ColumnName "ssn" -MaskingFunction "Default"
```

**A. Option A**

**B. Option B**

**C. Option C**

**D. Option D**

**E. Option E**

**Answer:** B E

Explanation

Dynamic data masking helps prevent unauthorized access to sensitive data by enabling customers to designate how much of the sensitive data to reveal with minimal impact on the application layer.

SQL users excluded from masking - A set of SQL users or AAD identities that get unmasked data in the SQL query results.

Note: The New-AzureRmSqlDatabaseDataMaskingRule cmdlet creates a data masking rule for an Azure SQL database.

References:

<https://docs.microsoft.com/en-us/powershell/module/azurerm.sql/new-azurermsqldatabasedatamaskingrule?view>

**NO.110** 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.

Margie's Travel is an international travel and bookings management service. The company is expanding into restaurant bookings. You are tasked with implementing Azure Search for the restaurants listed in their solution.

You create the index in Azure Search.

You need to import the restaurant data into the Azure Search service by using the Azure Search .NET SDK.

Solution:

1. Create a `SearchServiceClient` object to connect to the search index.
2. Create a `DataContainer` that contains the documents which must be added.
3. Create a `DataSource` instance and set its `Container` property to the `DataContainer`.
4. Set the `DataSource` property of the `SearchServiceClient`

Does the solution meet the goal?

**A.** Yes

**B.** No

**Answer:** B

Explanation

Use the following method:

- \* Create a `SearchIndexClient` object to connect to the search index
- \* Create an `IndexBatch` that contains the documents which must be added.
- \* Call the `Documents.Index` method of the `SearchIndexClient` and pass the `IndexBatch`.

References:

<https://docs.microsoft.com/en-us/azure/search/search-howto-dotnet-sdk>

**NO.111** 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 tools (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.

**Answer Area**

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

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

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

**NO.112** You maintain an existing Azure SQL Database instance. Management of the database is performed by an external party. All cryptographic keys are stored in an Azure Key Vault.

You must ensure that the external party cannot access the data in the SSN column of the Person table.

Will each protection method meet the requirement? To answer, drag the appropriate responses to the correct protection methods. Each response 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.

Responses	Protection method	Response
<input type="checkbox"/> Yes	Enable AlwaysOn encryption.	<input type="checkbox"/>
<input type="checkbox"/> No	Set the column encryption setting to disabled.	<input type="checkbox"/>
	Assign users to the Public fixed database role.	<input type="checkbox"/>
	Store column encryption keys in the system catalog view in the database.	<input type="checkbox"/>

**Answer:**

Responses	Protection method	Response
<input checked="" type="checkbox"/> Yes	Enable AlwaysOn encryption.	<input type="checkbox"/> Yes
<input type="checkbox"/> No	Set the column encryption setting to disabled.	<input type="checkbox"/> No
	Assign users to the Public fixed database role.	<input type="checkbox"/> Yes
	Store column encryption keys in the system catalog view in the database.	<input checked="" type="checkbox"/> No

Explanation

Responses	Protection method	Response
<input type="checkbox"/> Yes	Enable AlwaysOn encryption.	<input type="checkbox"/> Yes
<input type="checkbox"/> No	Set the column encryption setting to disabled.	<input type="checkbox"/> No
	Assign users to the Public fixed database role.	<input type="checkbox"/> Yes
	Store column encryption keys in the system catalog view in the database.	<input type="checkbox"/> No

Box 1: Yes

You can configure Always Encrypted for individual database columns containing your sensitive data. When setting up encryption for a column, you specify the information about the encryption algorithm and cryptographic keys used to protect the data in the column.

Box 2: No

Box 3: Yes

In SQL Database, the VIEW permissions are not granted by default to the public fixed database role. This enables certain existing, legacy tools (using older versions of DacFx) to work properly. Consequently, to work with encrypted columns (even if not decrypting them) a database administrator must explicitly grant the two VIEW permissions.

Box 4: No

All cryptographic keys are stored in an Azure Key Vault.

References:

<https://docs.microsoft.com/en-us/sql/relational-databases/security/encryption/always-encrypted-database-engine>

**NO.113** 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.

Actions	Answer Area
Enable autoscaling on the web app.	
Configure a Scale condition.	
Configure the web app to the Standard App Service tier.	(L)
Configure the web app to the Premium App Service tier.	(R)
Switch to an Azure App Services consumption plan.	
Add a Scale rule.	

### Answer:

Actions	Answer Area
Enable autoscaling on the web app.	
Configure a Scale condition.	
Configure the web app to the Standard App Service tier.	(L)
Configure the web app to the Premium App Service tier.	(R)
Switch to an Azure App Services consumption plan.	
Add a Scale rule.	(L)
	(R)

Explanation

Configure the web app to the Standard App Service tier.

Enable autoscaling on the web app.

Add a Scale rule.

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 condidation

References:

<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/>

**NO.114** You are developing a .NET Core MVC application for customers to research hotels. The application will use Azure Search. The application will search the index by using various criteria to locate documents related to hotels. The index will include search fields for rate, a list of amenities, and distance to the nearest airport.

The application must support the following scenarios for specifying search criteria and organizing results:

- \* Search the index by using regular expressions.
- \* Organize results by counts for name-value pairs.
- \* List hotels within a specified distance to an airport and that fall within a specific price range.

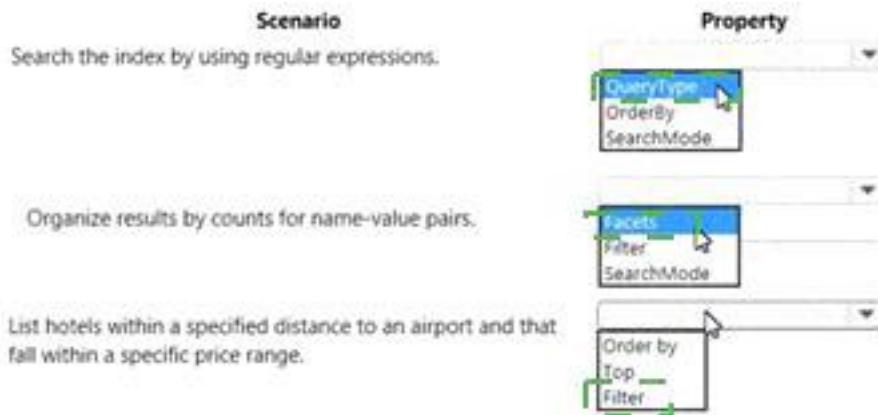
You need to configure the `SearchParameters` class.

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

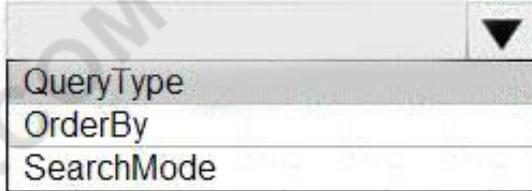
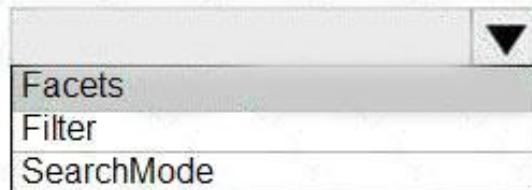
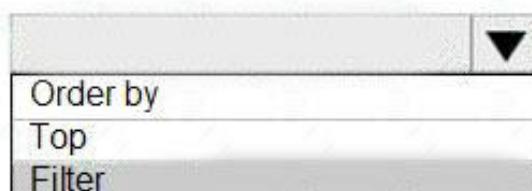
NOTE Each correct selection is worth one point.

Scenario	Property
Search the index by using regular expressions.	<input type="checkbox"/> QueryType <input type="checkbox"/> OrderBy <input type="checkbox"/> SearchMode
Organize results by counts for name-value pairs.	<input type="checkbox"/> Facets <input type="checkbox"/> Filter <input type="checkbox"/> SearchMode
List hotels within a specified distance to an airport and that fall within a specific price range.	<input type="checkbox"/> Order by <input type="checkbox"/> Top <input type="checkbox"/> Filter

**Answer:**



### Explanation

Scenario	Property
Search the index by using regular expressions.	
Organize results by counts for name-value pairs.	
List hotels within a specified distance to an airport and that fall within a specific price range.	

#### Box 1: QueryType

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.

#### Box 2: Facets

The facets property gets or sets the list of facet expressions to apply to the search query. Each facet expression contains a field name, optionally followed by a comma-separated list of name:value pairs.

#### Box 3: Filter

The `Filter` property gets or sets the OData \$filter expression to apply to the search query.

#### References:

<https://docs.microsoft.com/en-us/dotnet/api/microsoft.azure.search.models.searchparameters>

<https://docs.microsoft.com/en-us/azure/search/query-lucene-syntax>

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

us/dotnet/api/microsoft.azure.search.models.searchparameters.querytype

**NO.115** You are working for a company that designs mobile applications. They maintain a server where player records are assigned to their different games. The tracking system is new and in development.

The application uses Entity Framework to connect to an Azure Database. The database holds a Player table and Game table.

When adding a player, the code should insert a new player record, and add a relationship between an existing game record and the new player record.

The application will call CreatePlayerWithGame with the correct gameId and the playerId to start the process.

(Line numbers are included for reference only.)

```

01. namespace ContosoCraft
02. {
03.     public class PlayerDbContext : DbContext
04.     {
05.         public PlayerDbContext() : base ("name=dBConnectionString") { }
06.         public DbSet<Player> Players { get ; set ; }
07.         public DbSet<Game> Games { get ; set ; }
08.         protected override void OnModelCreating(ModelBuilder modelBuilder)
09.         {
10.             modelBuilder.Entity<Player>().HasMany(x => x.Games). WithMany (x => x.Players);
11.         }
12.     }
13.     internal sealed class dbConfiguration : DbMigrationsConfiguration<PlayerDbContext>
14.     {
15.         public dbConfiguration() { AutomaticMigrationsEnabled = true ; }
16.     }
17.     public class mp
18.     {
19.         public void CreatePlayerWithGame(int playerId, int gameId) => AddPlayer(playerId, GetGame(gameId));
20.         public Game GetGame(int gameId)
21.         {
22.             using (var db = new PlayerDbContext())
23.             {
24.                 return db.Games.FirstOrDefault(x => x.GameId == gameId);
25.             }
26.         }
27.         public Player AddPlayer (int playerId, Game game)
28.         {
29.             using (var db = new PlayerDbContext())
30.             {
31.                 var player = new Player
32.                 {
33.                     PlayerId = playerId,
34.                     Games = new List <Game> {game },
35.                 };
36.                 db.Players.Add(player);
37.                 db.SaveChanges();
38.                 return player;
39.             }
40.         }
41.         public class Player
42.         {
43.             public int PlayerId { get ; set; }
44.             public string PlayerName { get ; set; }
45.             public virtual List<Game> Games { get ; set; }
46.         }
47.         public class Game
48.         {
49.             public int GameId { get ; set }
50.             public string Title { get ; set; }
51.             public string Platform { get ; set; }
52.             public virtual List<Player> Players { get ; set; }
53.         }
54.     }

```

For each of the following statements, select Yes if the statement is true. Otherwise, select No.  
 NOTE: Each correct selection is worth one point.

	<b>Yes</b>	<b>No</b>
The code will successfully insert a player record.	<input type="radio"/>	<input type="radio"/>
The code has a bug and will insert an additional copy of the Game record with a new Id.	<input type="radio"/>	<input type="radio"/>
The code has a bug and will insert the wrong gameld value.	<input type="radio"/>	<input type="radio"/>
There is a valid many-to-many relationship between Players and Games.	<input type="radio"/>	<input type="radio"/>

**Answer:**

	<b>Yes</b>	<b>No</b>
The code will successfully insert a player record.	<input checked="" type="radio"/>	<input type="radio"/>
The code has a bug and will insert an additional copy of the Game record with a new Id.	<input type="radio"/>	<input checked="" type="radio"/>
The code has a bug and will insert the wrong gameld value.	<input checked="" type="radio"/>	<input type="radio"/>
There is a valid many-to-many relationship between Players and Games.	<input type="radio"/>	<input checked="" type="radio"/>

Explanation

	<b>Yes</b>	<b>No</b>
The code will successfully insert a player record.	<input type="radio"/>	<input type="radio"/>
The code has a bug and will insert an additional copy of the Game record with a new Id.	<input type="radio"/>	<input type="radio"/>
The code has a bug and will insert the wrong gameld value.	<input type="radio"/>	<input type="radio"/>
There is a valid many-to-many relationship between Players and Games.	<input type="radio"/>	<input type="radio"/>

Many-to-many relationships without an entity class to represent the join table are not yet supported.

However, you can represent a many-to-many relationship by including an entity class for the join table and mapping two separate one-to-many relationships.

```
protected override void OnModelCreating(ModelBuilder modelBuilder)
{
    modelBuilder.Entity<PostTag>()
        HasKey(t => new { t.PostId, t.TagId });
    modelBuilder.Entity<PostTag>()
        HasOne(pt => pt.Post)
        WithMany(p => p.PostTags)
        HasForeignKey(pt => pt.PostId);
    modelBuilder.Entity<PostTag>()
        HasOne(pt => pt.Tag)
        WithMany(t => t.PostTags)
        HasForeignKey(pt => pt.TagId);
}
```

**NO.116** 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.

NOTE: Each correct selection is worth one point.

- A. Assign the value of the hazard message MessageId property to the DeliveryCount property.
- B. Assign the value of the hazard message SequenceNumber property to the DeliveryCount property.
- C. Assign the value of the hazard message MessageId 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 SessionID property to the SequenceNumber property..
- F. Assign the value of the hazard message SessionID property to the ReplyToSessionId property.

**Answer:** D F

Explanation

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.

F: ReplyToSessionId: This value augments the ReplyTo information and specifies which SessionId should be set for the reply when sent to the reply entity.

**NO.117** 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.

Margie's Travel is an international travel and bookings management service. The company is expanding into restaurant bookings. You are tasked with implementing Azure Search for the restaurants listed in their solution. You create the index in Azure Search.

You need to import the restaurant data into the Azure Search service by using the Azure Search .NET SDK.

Solution:

- 1 Create a `SearchIndexClient` object to connect to the search index
2. Create an `IndexBatch` that contains the documents which must be added.
3. Call the `Documents.Index` method of the `SearchIndexClient` and pass the `IndexBatch`.

Does the solution meet the goal?

**A.** Yes

**B.** No

**Answer:** A

Explanation

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);
}
```

References:

<https://docs.microsoft.com/en-us/azure/search/search-howto-dotnet-sdk>

**NO.118** 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 store in the database? Each correct answer presents part of the solution.

NOTE: Each correct selection is worth one point.

**A.** `Telemetry.Context.Operation.Id`

- B.** Telemetry.Context.Cloud.RoleInstance
- C.** Telemetry.Id
- D.** Telemetry.ContextSession.Id
- E.** Telemetry.Name

**Answer:** A C

Explanation

References:

<https://docs.microsoft.com/en-us/azure/azure-monitor/app/custom-operations-tracking>

**NO.119** You use Azure Table storage to store customer information for an application. The data contains customer details and is partitioned by last name. You need to create a query that returns all customers with the last name Smith. Which code segment should you use?

- A.** TableQuery.GenerateFilterCondition("PartitionKey", Equals, "Smith")
- B.** TableQuery.GenerateFilterCondition("LastName", Equals, "Smith")
- C.** TableQuery.GenerateFilterCondition("PartitionKey", QueryComparisons.Equal, "Smith")
- D.** TableQuery.GenerateFilterCondition("LastName", QueryComparisons.Equal, "Smith")

**Answer:** C

Explanation

Retrieve all entities in a partition. The following code example specifies a filter for entities where 'Smith' is the partition key. This example prints the fields of each entity in the query results to the console.

Construct the query operation for all customer entities where PartitionKey="Smith".

```
TableQuery<CustomerEntity> query = new  
TableQuery<CustomerEntity>().Where(TableQuery.GenerateFilterCondition("PartitionKey",  
QueryComparisons.Equal, "Smith")); References:  
https://docs.microsoft.com/en-us/azure/cosmos-db/table-storage-how-to-use-dotnet
```

**NO.120** You are preparing to deploy a medical records application to an Azure virtual machine (VM). The application will be deployed by using a VHD produced by an on-premises build server. You need to ensure that both the application and related data are encrypted during and after deployment to Azure.

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

**Actions**

- Encrypt the on-premises VHD by using BitLocker without a TPM. Upload the VM to Azure Storage.
- Run the Azure PowerShell command `Set-AzureRmVMDiskEncryptionExtension`.
- Run the Azure PowerShell command `Set-AzureRmVMOSDisk`.
- Encrypt the on-premises VHD by using BitLocker with a TPM. Upload the VM to Azure Storage.
- Run the Azure PowerShell command `New-AzureRmVM`.

**Answer area****Answer:****Actions**

- Encrypt the on-premises VHD by using BitLocker without a TPM. Upload the VM to Azure Storage.
- Run the Azure PowerShell command `Set-AzureRmVMDiskEncryptionExtension`.
- Run the Azure PowerShell command `Set-AzureRmVMOSDisk`.
- Encrypt the on-premises VHD by using BitLocker with a TPM. Upload the VM to Azure Storage.
- Run the Azure PowerShell command `New-AzureRmVM`.

**Answer area**

- Encrypt the on-premises VHD by using BitLocker without a TPM. Upload the VM to Azure Storage.
- Run the Azure PowerShell command `Set-AzureRmVMOSDisk`.
- Run the Azure PowerShell command `Set-AzureRmVMDiskEncryptionExtension`.

**Explanation**

**Encrypt the on-premises VHD by using BitLocker without a TPM.  
Upload the VM to Azure Storage.**

**Run the Azure PowerShell command `Set-AzureRmVMDSDisk`.**

**Run the Azure PowerShell command `Set-AzureRmVMDiskEncryptionExtension`.**

Step 1: Encrypt the on-premises VHD by using BitLocker without a TPM. Upload the VM to Azure Storage  
 Step 2: Run the Azure PowerShell command `Set-AzureRMVMDisk` To use an existing disk instead of creating a new disk you can use the `Set-AzureRMVMDisk` command.

Example:

```
$osDiskName = $vmname+'_osDisk'
```

```
$osDiskCaching = 'ReadWrite'
```

```
$osDiskVhdUri = "https://$stoname.blob.core.windows.netvhds/" + $vmname + "_os.vhd"
```

\$vm = Set-AzureRmVMOSDisk -VM \$vm -VhdUri \$osDiskVhdUri -name \$osDiskName -Create Step 3: Run the Azure PowerShell command Set-AzureRmVMDiskEncryptionExtension Use the Set-AzVMDiskEncryptionExtension cmdlet to enable encryption on a running IaaS virtual machine in Azure.

Incorrect:

Not TPM: BitLocker can work with or without a TPM. A TPM is a tamper resistant security chip on the system board that will hold the keys for encryption and check the integrity of the boot sequence and allows the most secure BitLocker implementation. A VM does not have a TPM.

References:

<https://www.itprotoday.com/iaaspaas/use-existing-vhd-azurerm-vm>

**NO.121** 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.

.....

Actions
Create an integration account in the Azure portal.
Link the custom connector to the Logic App.
Update the Logic App to use the partners, schemas, certificates, maps, and agreements.
Create a custom connector for the Logic App.
Add partners, schemas, certificates, maps, and agreements.
Link the Logic App to the integration account.

Answer area



**Answer:**

.....

Actions
Create an integration account in the Azure portal.
Link the custom connector to the Logic App.
Update the Logic App to use the partners, schemas, certificates, maps, and agreements.
Create a custom connector for the Logic App.
Add partners, schemas, certificates, maps, and agreements.
Link the Logic App to the integration account.

Answer area

Create a custom connector for the Logic App.
Create an integration account in the Azure portal.
Link the Logic App to the integration account.
Add partners, schemas, certificates, maps, and agreements.



**NO.122** 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.

#### **Answer Area**

Setting	Action
Caching behavior	<div style="border: 1px solid black; padding: 5px; width: fit-content;"> <input checked="" type="checkbox"/> Bypass cache  <input type="checkbox"/> Override  <input type="checkbox"/> Set if missing         </div>
Cache expiration duration	<div style="border: 1px solid black; padding: 5px; width: fit-content;"> <input type="checkbox"/> 1 second  <input type="checkbox"/> 1 minute  <input checked="" type="checkbox"/> 1 hour  <input type="checkbox"/> 1 day         </div>
Query string caching behavior	<div style="border: 1px solid black; padding: 5px; width: fit-content;"> <input type="checkbox"/> Ignore query strings  <input checked="" type="checkbox"/> Bypass caching for query strings  <input type="checkbox"/> Cache every unique URL         </div>

#### **Answer:**

#### **Answer Area**

Setting	Action
Caching behavior	<div style="border: 1px solid black; padding: 5px; width: fit-content;"> <input checked="" type="checkbox"/> Bypass cache  <input checked="" type="checkbox"/> Override  <input type="checkbox"/> Set if missing         </div>
Cache expiration duration	<div style="border: 1px solid black; padding: 5px; width: fit-content;"> <input type="checkbox"/> 1 second  <input checked="" type="checkbox"/> 1 minute  <input checked="" type="checkbox"/> 1 hour  <input type="checkbox"/> 1 day         </div>
Query string caching behavior	<div style="border: 1px solid black; padding: 5px; width: fit-content;"> <input type="checkbox"/> Ignore query strings  <input checked="" type="checkbox"/> Bypass caching for query strings  <input type="checkbox"/> Cache every unique URL         </div>

#### **Explanation**

Setting	Action
Caching behavior	<ul style="list-style-type: none"><li>Bypass cache</li><li>Override</li><li>Set if missing</li></ul>
Cache expiration duration	<ul style="list-style-type: none"><li>1 second</li><li>1 minute</li><li>1 hour</li><li>1 day</li></ul>
Query string caching behavior	<ul style="list-style-type: none"><li>Ignore query strings</li><li>Bypass caching for query strings</li><li>Cache every unique URL</li></ul>

**NO.123** You are developing an app that manages users for a video game. You plan to store the region, email address, and phone number for the player. Some players may not have a phone number. The player's region will be used to load-balance data.

Data for the app must be stored in Azure Table Storage.

You need to develop code to retrieve data for an individual player.

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

NOTE: Each correct selection is worth one point.

```

public class PlayerEntity : TableEntity
{
    public PlayerEntity()
    {
    }
    public PlayerEntity(string region, string email)
    {
        PartitionKey = 

|        |   |
|--------|---|
|        | ▼ |
| email  |   |
| phone  |   |
| region |   |

 ;
        RowKey = 

|        |   |
|--------|---|
|        | ▼ |
| email  |   |
| phone  |   |
| region |   |

 ;
    }
    public string Phone { get; set; }
}
public class Player
{
}

protected PlayerEntity player;
async void GetPlayer(string cs, 

|                    |   |
|--------------------|---|
|                    | ▼ |
| CloudTable         |   |
| CloudTableClient   |   |
| TableEntity        |   |
| TableEntityAdapter |   |

 table, string pk, string rk)
{
    

|                                                                       |   |
|-----------------------------------------------------------------------|---|
| TableEntity query =TableEntity.Retrieve<PlayerEntity>(pk, rk);        | ▼ |
| TableOperation query =TableOperation.Retrieve<PlayerEntity>(pk,rk);   |   |
| TableResult query =TableQuery.Retrieve<PlayerEntity>(pk,rk);          |   |
| TableResultSegment query =TableResult.Retrieve<PlayerEntity>(pk, rk); |   |



|                                                       |   |
|-------------------------------------------------------|---|
| TableEntity data =await table.ExecuteAsync(query);    | ▼ |
| TableOperation data =await table.ExeucteAsync(query); |   |
| TableQuery data =await table.ExecuteAsync(query);     |   |
| TableResult data =await table.ExecuteAsync(query);    |   |


    player=data.Result as PlayerEntity;
}
}

```

**Answer:**

```

public class PlayerEntity : TableEntity
{
    public PlayerEntity()
    {
    }
    public PlayerEntity(string region, string email)
    {
        PartitionKey = 

|        |
|--------|
| email  |
| phone  |
| region |

 ;
        RowKey = 

|        |
|--------|
| email  |
| phone  |
| region |

 ;
    }
    public string Phone { get; set; }
}
public class Player
}

protected PlayerEntity player;
async void GetPlayer(string cs, 

|                    |
|--------------------|
| CloudTable         |
| CloudTableClient   |
| TableEntity        |
| TableEntityAdapter |

 table, string pk, string rk)
{
    TableEntity query = TableEntity.Retrieve<PlayerEntity>(pk, rk);
    TableOperation query = TableOperation.Retrieve<PlayerEntity>(pk, rk);
    TableResult query = TableQuery.Retrieve<PlayerEntity>(pk, rk);
    TableResultSegment query = TableResult.Retrieve<PlayerEntity>(pk, rk);

    TableEntity data = await table.ExecuteAsync(query);
    TableOperation data = await table.ExecuteAsync(query);
    TableQuery data = await table.ExecuteAsync(query);
    TableResult data = await table.ExecuteAsync(query);

    player = data.Result as PlayerEntity;
}
}

```

## Explanation

## Answer Area

```

public class PlayerEntity : TableEntity
{
    public PlayerEntity()
    {
    }

    public PlayerEntity(string region, string email)
    {
        PartitionKey = 

|        |   |
|--------|---|
|        | ▼ |
| email  |   |
| phone  |   |
| region |   |

 ;
        RowKey = 

|        |   |
|--------|---|
|        | ▼ |
| email  |   |
| phone  |   |
| region |   |

 ;
    }

    public string Phone { get; set; }
}

public class Player
{
    protected PlayerEntity player;
    async void GetPlayer(string cs, 

|                    |   |
|--------------------|---|
|                    | ▼ |
| CloudTable         |   |
| CloudTableClient   |   |
| TableEntity        |   |
| TableEntityAdapter |   |

 table, string pk, string rk)
{
    TableEntity query = TableEntity.Retrieve<PlayerEntity>(pk, rk);
    TableOperation query = TableOperation.Retrieve<PlayerEntity>(pk, rk);
    TableResult query = TableQuery.Retrieve<PlayerEntity>(pk, rk);
    TableResultSegment query = TableResult.Retrieve<PlayerEntity>(pk, rk);

    TableEntity data = await table.ExecuteAsync(query);
    TableOperation data = await table.ExecuteAsync(query);
    TableQuery data = await table.ExecuteAsync(query);
    TableResult data = await table.ExecuteAsync(query);

    player = data.Result as PlayerEntity;
}
}

```

Box 1: region

The player's region will be used to load-balance data.

Choosing the PartitionKey.

The core of any table's design is based on its scalability, the queries used to access it, and storage operation requirements. The PartitionKey values you choose will dictate how a table will be partitioned and the type of queries that can be used. Storage operations, in particular inserts, can also affect your choice of PartitionKey values.

Box 2: email

Not phone number some players may not have a phone number.

Box 3: CloudTable

Box 4 : TableOperation query =..

Box 5: TableResult

References:

<https://docs.microsoft.com/en-us/rest/api/storageservices/designing-a-scalable-partitioning-strategy-for-azure-table-storage>

**NO.124** You are writing code to create and run an Azure Batch job.

You have created a pool of compute nodes.

You need to choose the right class and its method to submit a batch job to the Batch service.

Which method should you use?

- A.** JobOperations.CreateJob()
- B.** CloudJob.Enable(IEnumerable<BatchClientBehavior>)
- C.** CloudJob.CommitAsync(IEnumerable<BatchClientBehavior>, CancellationToken)
- D.** JobOperations.EnableJob(String, IEnumerable<BatchClientBehavior>)
- E.** JobOperations.EnableJobAsync(String, IEnumerable<BatchClientBehavior>, CancellationToken)

**Answer:** C

Explanation

A Batch job is a logical grouping of one or more tasks. A job includes settings common to the tasks, such as priority and the pool to run tasks on. The app uses the BatchClient.JobOperations.CreateJob method to create a job on your pool.

The Commit method submits the job to the Batch service. Initially the job has no tasks.

```
{
CloudJob job = batchClient.JobOperations.CreateJob();
job.Id = JobId;
job.PoolInformation = new PoolInformation { PoolId = PoolId };
job.Commit();
}
```

References:

<https://docs.microsoft.com/en-us/azure/batch/quick-run-dotnet>

**NO.125** 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.

Actions	Answer area
Create a Service Bus topic for each restaurant for which a driver can receive messages.	
Create a single Service Bus topic.	
Create a single Service Bus subscription.	
Create a single Service Bus Namespace.	
Create a Service Bus Namespace 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.	

**Answer:**

Actions	Answer area
Create a Service Bus topic for each restaurant for which a driver can receive messages.	
Create a single Service Bus topic.	
Create a single Service Bus subscription.	
Create a single Service Bus Namespace.	
Create a Service Bus Namespace 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.	

**Explanation****Answer area**

Create a 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.

References:

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

**NO.126** 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 blocks to Archive tier after they have not been accessed for 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.

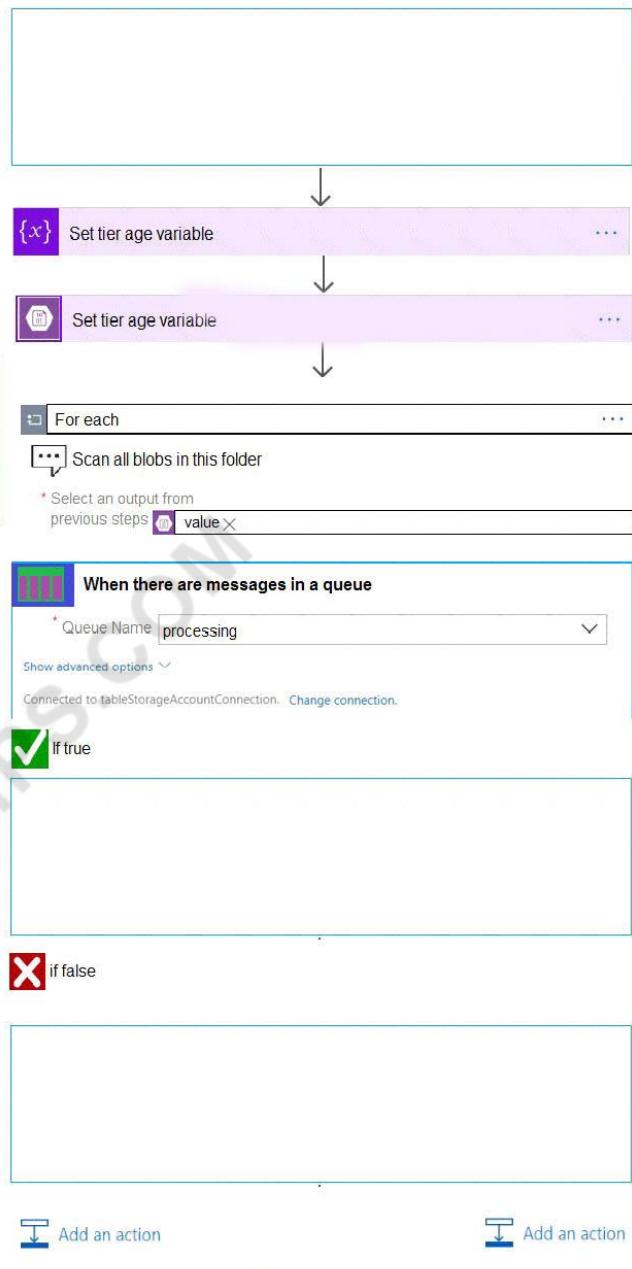
You may need to drag the split bar between panes or scroll to view content.

NOTE: Each correct selection is worth one point.

#### Triggers and Action Blocks

- Insert Entity**: \*Table processing, \*Entity Path
- Tier blob**: If blob is older than the defined value, tier it to Cool or Archive tier. Blob path Path, Blob Tier Archive
- When there are messages in a queue**: Queue Name processing
- Recurrence**: Interval 1, Frequency Month
- When there are messages in a queue**: Queue Name processing

#### Answer Area



#### Answer:

**Triggers and Action Blocks**

**Insert Entity**

\*Table: processing  
\*Entity: Path X

Show advanced options ▾

**Tier blob**

... If blob is older than the defined value, tier it to Cool or Archive tier  
\* Blob path: Path X  
\* Blob Tier: Archive

**When there are messages in a queue**

\* Queue Name: processing  
Show advanced options ▾  
Connected to tableStorageAccountConnection. Change connection.

**Recurrence**

\* Interval: 1 \* Frequency: Month  
Show advanced options ▾

**Answer Area**

**Recurrence**

\* Interval: 1 \* Frequency: Month  
Show advanced options ▾

{x} Set tier age variable

Set tier age variable

For each  
... Scan all blobs in this folder  
\* Select an output from previous steps  
value X

**When there are messages in a queue**

\* Queue Name: processing  
Show advanced options ▾  
Connected to tableStorageAccountConnection. Change connection.

✓ if true  
**Recurrence**

\* Interval: 1 \* Frequency: Month  
Show advanced options ▾

✗ if false

**When there are messages in a queue**

\* Queue Name: processing  
Show advanced options ▾  
Connected to tableStorageAccountConnection. Change connection.

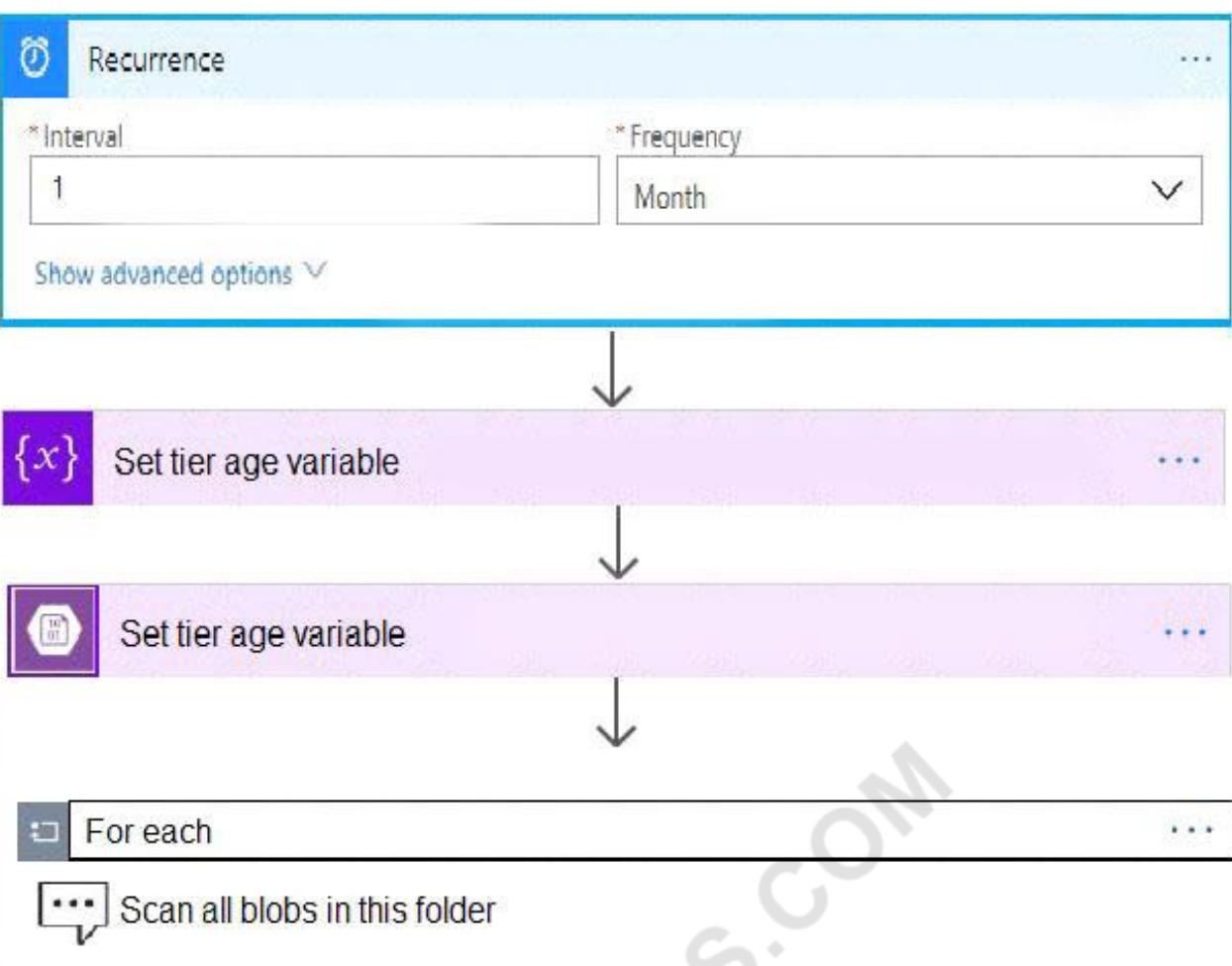
Add an action

Add an action

Add an action

**Explanation**

## Answer Area



\* Select an output from previous steps  value 

 When there are messages in a queue

\* Queue Name processing 

Show advanced options 

Connected to tableStorageAccountConnection. Change connection.

 If true

 Recurrence 

\* Interval 1  \* Frequency Month 

Show advanced options 

 if false

 When there are messages in a queue

\* Queue Name processing 

Show advanced options 

Connected to tableStorageAccountConnection. Change connection.

 Add an action  Add an action

 Add an action

- Box 1: Recurrence
- Box 2: Insert Entity
- Box 3 (if true): Tier Blob
- Box 4: (if false):  
Leave blank.
- References:

<https://docs.microsoft.com/en-us/azure/logic-apps/logic-apps-perform-data-operations>

**NO.127** 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 3 specific period of time. You need to configure Event Grid to ensure security.

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

NOTE: Each correct selection is worth one point

## Authentication

## Type

WebHook event delivery

SAS tokens
Key authentication
JWT token

Topic publishing

ValidationCode handshake
ValidationURL handshake
Management Access Control

**Answer:**

## Authentication

## Type

WebHook event delivery

SAS tokens
Key authentication
JWT token

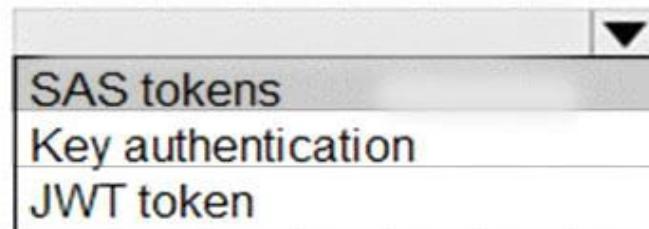
Topic publishing

ValidationCode handshake
ValidationURL handshake
Management Access Control

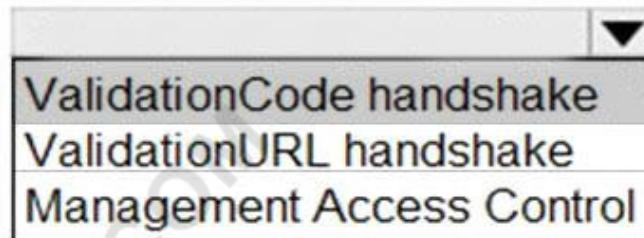
Explanation

## Authentication Type

WebHook event delivery



Topic publishing



### 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.

**NO.128** You must implement Application Insights instrumentation capabilities utilizing the Azure Mobile Apps SDK to provide meaningful analysis of user interactions with a mobile app. You need to capture the data required to implement the Usage Analytics feature of Application Insights.

Which three data values should you capture? Each correct answer presents part of the solution

NOTE: Each correct selection is worth one point.

- A. Trace
- B. Session Id
- C. Exception
- D. User Id
- E. Events

**Answer:** A D E

### Explanation

Application Insights is a service for monitoring the performance and usage of your apps. This module allows you to send telemetry of various kinds (events, traces, etc.) to the Application Insights service where your data can be visualized in the Azure Portal.

Application Insights manages the ID of a session for you.

References:

<https://github.com/microsoft/ApplicationInsights-Android>

**NO.129** You need to migrate on-premises shipping data to Azure What should you us\*?

- A. Azure Migrate
- B. Azure Cosmos DB Data Migration tool (dt.exe)
- C. AzCopy
- D. Azure Database Migration service

**Answer:** B

Explanation

<https://azure.microsoft.com/en-us/updates/documentdb-data-migration-tool/>

**NO.130** 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.

Azure Resource
HTTP(s) endpoint
Basic
Client cert

Configuration parameter	Value
Target	value
Gateway credentials	value

**Answer:**

Azure Resource
HTTP(s) endpoint
Basic
Client cert

Configuration parameter	Value
Target	Azure Resource
Gateway credentials	Client cert

Explanation

Configuration parameter	Value
Target	Azure Resource
Gateway credentials	Client cert

Box 1: Azure Resource

Box 2: Client cert

API Management allows to secure access to the back-end service of an API using client certificates.

References:

<https://docs.microsoft.com/en-us/rest/api/apimanagement/apimanagementrest/azure-api-management-rest-api-ba>

**NO.131** You have an app that stores player scores for an online game. The app stores data in Azure tables using a class named PlayerScore as the table entity. The table is populated with 100,000 records.

You are reviewing the following section of code that is intended to retrieve 20 records where the player score exceeds 15,000. (Line numbers are included for reference only.)

```

1 public void GetScore(string playerId, int score, string gameId)
2 {
3     TableQuery<DynamicTableEntity> query = new TableQuery<DynamicTableEntity>().Select(new string[] { "Score" })
        .Where(TableQuery.GenerateFilterCondition("Score", QueryComparisons.GreaterThanOrEqual, 15000)).Take
(20);
4     EntityResolver<KeyValuePair<string, int?>> resolver =
        (partitionKey, rowKey, ts, props, etag) => new KeyValuePair<string, int?>(rowKey, props["Score"].Int32Value);
5     foreach (var scoreItem in scoreTable.ExecuteQuery(query, resolver, null, null))
6     {
        Console.WriteLine($"{scoreItem.Key} {scoreItem.Value}");
7     }
8 }

9 public class PlayerScore : TableEntity
10 {
11     public PlayerScore(string gameId, string playerId, int score, long timePlayed)
12     {
13         PartitionKey = gameId;
14         RowKey = playerId;
15         Score = score;
16         TimePlayed = timePlayed;
17     }
18     public int Score { get; set; }
19     public long TimePlayed { get; set; }
20 }

```

You have the following code. (Line numbers are included for reference only.) You store customer information in an Azure Cosmos database. The following data already exists in the database:

You develop the following code. (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.GenerateAnd, TableQuery.GenerateFilterCondition("Email", QueryComparisons.Equal, "Smith")
06         TableOperstors.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.

	Yes	No
The code queries the Azure table and retrieves the TimePlayed property from the table	<input type="radio"/>	<input type="radio"/>
The code will display a maximum of twenty records.	<input type="radio"/>	<input type="radio"/>
All records will be sent to the client. The client will display records for scores greater than or equal to 15,000.	<input type="radio"/>	<input type="radio"/>
The scoreItem.Key property of the KeyValuePairs that ExecuteQuery returns will contain a value for PlayerID.	<input type="radio"/>	<input type="radio"/>

**Answer:**

	Yes	No
The code queries the Azure table and retrieves the TimePlayed property from the table	<input type="radio"/>	<input checked="" type="radio"/>
The code will display a maximum of twenty records.	<input checked="" type="radio"/>	<input type="radio"/>
All records will be sent to the client. The client will display records for scores greater than or equal to 15,000.	<input checked="" type="radio"/>	<input type="radio"/>
The scoreItem.Key property of the KeyValuePairs that ExecuteQuery returns will contain a value for PlayerID.	<input checked="" type="radio"/>	<input type="radio"/>

Explanation

	Yes	No
The code queries the Azure table and retrieves the TimePlayed property from the table	<input type="radio"/>	<input checked="" type="radio"/>
The code will display a maximum of twenty records.	<input type="radio"/>	<input checked="" type="radio"/>
All records will be sent to the client. The client will display records for scores greater than or equal to 15,000.	<input type="radio"/>	<input checked="" type="radio"/>
The scoreItem.Key property of the KeyValuePairs that ExecuteQuery returns will contain a value for PlayerID.	<input type="radio"/>	<input checked="" type="radio"/>

Box 1: No

Box 2: Yes

The TableQuery.Take method defines the upper bound for the number of entities the query returns.

Example:

query.Take(10);

Box 3: Yes

Box 4: Yes

References:

<https://www.vkinfotek.com/azureqa/how-do-i-query-azure-table-storage-using-tablequery-class.html>

**NO.132** You have an Azure Batch project that processes and converts files and stores the files in Azure storage. You are developing a function to start the batch job.

You add the following parameters to the function.

Parameter name	Description
fileTasks	a list of tasks to be run
jobId	the identifier that must be assigned to the job
outputContainerSasUrl	a storage SAS URL to store successfully converted files
failedContainerSasUrl	a storage SAS URL to store copies of files that failed to convert.

You must ensure that converted files are placed in the container referenced by the outputContainerSasUrl parameter. Files which fail to convert are places in the container referenced

by the failedContainerSasUrl parameter.

You need to ensure the files are correctly processed.

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

```
public List<CloudTasks> StartTasks(List<FileTask> fileTasks, string jobId,
    string outputContainerSasUrl, string failedContainerSasUrl)
{
    BatchSharedKeyCredentials sharedKeyCredentials =
        new BatchSharedKeyCredentials(batchAccountUrl, batchAccountName,
    batchAccountKey);
    List<CloudTask> tasks = new List<CloudTask>();
    using (BatchClient batchClient = BatchClient.Open(sharedKeyCredentials))
    {
        CloudJob = batchClient.JobOperations.▼ ();
        GetJob
        GetTask
        EnableJob
        CreateJob
        job.Id = jobId;
        job.PoolInformation = new PoolInformation { PoolId = poolId };
        job.Commit();
        fileTasks.ForEach((fileTask) =>
        {
            string taskId = $"Task{DateTime.NowToFileTimeUtc().ToString()}";
            CloudTask task = new CloudTask(taskId, fileTask.Command);
            List<OutputFile> fileList = new List<OutputFile>();
            OutputFileBlobContainerDestination outputContainer =
                new OutputFileBlobContainerDestination(outputContainerSasUrl);
            OutputFileBlobContainerDestination failedContainer =
                new OutputFileBlobContainerDestination(failedContainerSasUrl);
            fileList.Add(new OutputFile(fileTask.Output,
                new OutputFileDestination(outputContainer),
                new OutputFileUploadOptions(OutputFileUploadCondition.▼ ))));
            fileList.Add(new OutputFile(fileTask.Output,
                new OutputFileDestination(failedContainer),
                new OutputFileUploadOptions(OutputFileUploadCondition.▼ ))));
            task.▼ =outputFileList;
            OutputFiles
            FilesToStage
            ResourceFiles
            StageFiles
            task.Add(task);
        });
    }
    return tasks;
}
```

The screenshot shows three separate dropdown menus, each with a downward-pointing arrow icon to its right:

- The first dropdown menu contains the options: GetJob, GetTask, EnableJob, and CreateJob.
- The second dropdown menu contains the options: TaskSuccess, TaskFailure, and TaskCompletion.
- The third dropdown menu contains the options: TaskSuccess, TaskFailure, and TaskCompletion.

### Answer:

## Answer Area

```

public list<CloudTasks> StartTasks(List<FileTask> fileTasks, string jobId,
string outputContainerSasUrl, string failedContainerSasUrl)
{
    BatchSharedKeyCredentials sharedKeyCredentials =
        new BatchSharedKeyCredentials(batchAccountUrl, batchAccountName,
batchAccountKey);
    List<CloudTask> tasks = new List<CloudTask>();
    using (BatchClient batchClient = BatchClient.Open(sharedKeyCredentials))
    {
        CloudJob = batchClient.JobOperations. ▼ () ;
        GetJob
        GetTask
        EnableJob
        CreateJob | ▼
        job.Id = jobId,
        job.PoolInfromation = new PoolInformation { PoolId = poolId } ;
        job.Commit();
        fileTasks.ForEach((fileTask) =>
        {
            string taskId = $"Task{DateTime.NowToFileTimeUtc().ToString()}";
            CloudTask task = new CloudTask (taskId, fileTask.Command);
            List<OutputFile> output fileList = new List<OutputFile>();
            OutputFileBlobContainerDestination outputContainer =
                new OutputFileBlobContainerDestination(outputContainerSasUrl);
            OutputFileBlobContainerDestination failedContainer =
                new OutputFileBlobContainerDestination (failedContainerSasUrl);
            outputFileList.Add(new OutputFile(fileTask.Output,
                new OutputFileDestination(outputContainer),
                new OutputFileUploadOptions(OutputFileUploadCondition. ▼ )) );
            TaskSuccess
            TaskFailure
            TaskCompletion | ▼
            outputFileList.Add(new OutputFile(fileTask.Output,
                new OutputFileDestination(failedContainer),
                new OutputFileUploadOptions(OutputFileUploadCondition, ▼ )) );
            TaskSuccess
            TaskFailure | ▼
            TaskCompletion
            task. ▼ =output fileList;
            task.OutputFiles
            FilesToStage
            ResourceFiles
            StageFiles | ▼
            task.Add(task);
        });
    }
    return tasks,
}

```

## Explanation

```

CloudJob = batchClient.JobOperations. ▼ ();
GetJob
GetTask
EnableJob
CreateJob

job.Id = jobId,
job.PoolInformation = new PoolInformation { PoolId = poolId };
job.Commit();
fileTasks.ForEach((fileTask) =>
{
    string taskId = $"Task{DateTime.NowToFileTimeUtc().ToString()}";
    CloudTask task = new CloudTask (taskId, fileTask.Command);
    List<OutputFile> fileList = new List<OutputFile>();
    OutputFileBlobContainerDestination outputContainer =
        new OutputFileBlobContainerDestination(outputContainerSasUrl);
    OutputFileBlobContainerDestination failedContainer =
        new OutputFileBlobContainerDestination (failedContainerSasUrl);
    fileList.Add(new OutputFile(fileTask.Output,
        new OutputFileDestination(outputContainer),
        new OutputFileUploadOptions(OutputFileUploadCondition. ▼ ))) );
    TaskSuccess
    TaskFailure
    TaskCompletion
    fileList.Add(new OutputFile(fileTask.Output,
        new OutputFileDestination(failedContainer),
        new OutputFileUploadOptions(OutputFileUploadCondition, ▼ )) );
    TaskSuccess
    TaskFailure
    TaskCompletion

    task.▼ =outputFileList;
    OutputFiles
    FilesToStage
    ResourceFiles
    StageFiles

    task.Add(task);
}) ;
]
return tasks,
}

```

Box 1: CreateJob

Box 2: TaskSuccess

TaskSuccess: Upload the file(s) only after the task process exits with an exit code of 0.

Incorrect: TaskCompletion: Upload the file(s) after the task process exits, no matter what the exit code was.

Box 3: TaskFailure

TaskFailure: Upload the file(s) only after the task process exits with a nonzero exit code.

Box 4: OutputFiles

To specify output files for a task, create a collection of `OutputFile` objects and assign it to the `CloudTask.OutputFiles` property when you create the task.

## References:

<https://docs.microsoft.com/en-us/dotnet/api/microsoft.azure.batch.protocol.models.outputfileuploadcondition>  
<https://docs.microsoft.com/en-us/azure/batch/batch-task-output-files>

**NO.133** You are developing an Azure App Service hosted ASP.NET Core API app by using C#. The API app will allow users to authenticate by using Twitter and Azure Active Directory (Azure AD). Users must be authenticated before calling API methods. You must log the user's name for each method call.

You need to configure the API method calls.

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

NOTE: Each correct selection is worth one point.

Code segment	Value
Attribute	<div style="border: 1px solid black; padding: 5px;"><input type="checkbox"/> Authorize <input type="checkbox"/> AllowAnonymous <input type="checkbox"/> AutoValidateAntiforgeryToken</div>
Request Header	<div style="border: 1px solid black; padding: 5px;"><input type="checkbox"/> X-MS-CLIENT-PRINCIPAL-NAME <input type="checkbox"/> Proxy-Authorization <input type="checkbox"/> X-Forwarded-For <input type="checkbox"/> X-MS-CLIENT-PRINCIPAL-ID</div>

**Answer:**

Code segment	Value
Attribute	<pre>Authorize AllowAnonymous AutoValidateAntiforgeryToken</pre>
Request Header	<pre>X-MS-CLIENT-PRINCIPAL-NAME Proxy-Authorization X-Forwarded-For X-MS-CLIENT-PRINCIPAL-ID</pre>

Explanation

Code segment	Value
Attribute	<pre>Authorize AllowAnonymous AutoValidateAntiforgeryToken</pre>
Request Header	<pre>X-MS-CLIENT-PRINCIPAL-NAME Proxy-Authorization X-Forwarded-For X-MS-CLIENT-PRINCIPAL-ID</pre>

+ Explanation:

Box 1: Authorize

Box 2: X-MS-CLIENT-PRINCIPAL-NAME

App Service passes user claims to your application by using special headers. External requests aren't allowed to set these headers, so they are present only if set by App Service. Some example headers

include:

X-MS-CLIENT-PRINCIPAL-NAME

X-MS-CLIENT-PRINCIPAL-ID

Here's the set of headers you get from Easy Auth for a Twitter authenticated user:

```
{
  "cookie": "AppServiceAuthSession=Lx43...xHDTA==",
  "x-ms-client-principal-name": "evilSnobu",
  "x-ms-client-principal-id": "35....",
  "x-ms-client-principal-idp": "twitter",
  "x-ms-token-twitter-access-token": "35...Dj",
  "x-ms-token-twitter-access-token-secret": "OK3...Jx",
}
```

References:

<https://docs.microsoft.com/en-us/azure/app-service/app-service-authentication-how-to>

### NO.134 You are validating the configuration of an Azure Search indexer.

The service has been configured with an indexer that uses the Import Data option. The index is configured using options as shown in the Index Configuration exhibit. (Click the Index Configuration tab.)

The screenshot shows two overlapping windows. The top window is titled 'Import data' and contains a sidebar with options: 'Data Source' (set to 'tablesource'), 'Cognitive Search' (with a link to 'Add cognitive skills (Optional)'), 'Index' (with a link to 'Customize target index'), and 'Indexer' (with a link to 'Import your data'). The bottom window is titled 'Index' and displays the 'Index Configuration' tab. It includes a message: 'We provided a default index for you. You can delete the fields you don't need. Everything is editable, but once the index is built, deleting or changing existing fields will require re-indexing your documents.' Below this are sections for 'Index name' (set to 'azuretable-index') and 'Key' (set to 'RowKey'). The 'Analyzer' and 'Suggester' dropdowns are also visible. A large table lists index fields with columns for 'FIELD NAME', 'TYPE', 'RETRIEVABLE', 'FILTERABLE', 'SORTABLE', 'PAGINABLE', 'SEARCHABLE', and '...'. Fields listed include PartitionKey, RowKey, InStockCount, ItemDescription, ItemName, LocationRow, LocationShelf, and SKU. Most fields are of type Edm.String or Edm.Int32. The 'OK' button at the bottom is highlighted in blue.

You use an Azure table as the data source for the import operation. The table contains three records with item inventory data that matches the fields in the Storage data exhibit. These records were imported when the index was created. (Click the Storage Data tab.) When users search with no filter, all three records are displayed.

Category	RowKey	Timestamp	InStockCount	ItemDescription	ItemName	LocationRow	LocationShelf	SKU
Tool	3	2018-08-25T15:42:29.135Z	33	A box of CIRCULAR CHAIN SAWS	Chainsaw	3	3	123456
Hardware	2	2018-08-25T15:48:08.403Z	2	A bag of bolts	Bolts	1	4	678904
Hardware	1	2018-08-25T15:48:41.403Z	23	A box of nails	Nails	2	5	554368

When users search for items by description, Search explorer returns no records. The Search Explorer exhibit shows the query and results for a test. In the test, a user is trying to search for all items in the table that have a description that contains the word bag. (Click the Search Explorer tab.)

The screenshot shows the 'Search explorer' interface. In the 'Query string' field, the user has typed 'search:bag'. Below the query, the 'Request URL' is displayed as <https://itemsearch1103/search/windows.net/indexes/available-index/docs?api-version=2017-11-11&search=bag>. The 'Results' section shows the following JSON output:

```

1 [
2   {
3     "Metadata": {
4       "Value": []
5     }
6   }
]
  
```

You need to resolve the issue.

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

NOTE: Each correct selection is worth one point.

	Yes	No
You can resolve the issue by recreating the search index with the same settings for all fields except ItemDescription. Select the SEARCHABLE option for this field	<input type="radio"/>	<input type="radio"/>
You can resolve the issue by selecting the index, editing the ItemDescription field, and selecting the SEARCHABLE option for the field.	<input type="radio"/>	<input type="radio"/>
You can resolve the issue by running the indexer.	<input type="radio"/>	<input type="radio"/>
You can resolve the issue by changing the query string in Search explorer to <code>bag of</code> to return the correct results	<input type="radio"/>	<input type="radio"/>

### Answer:

	Yes	No
You can resolve the issue by recreating the search index with the same settings for all fields except ItemDescription. Select the SEARCHABLE option for this field	<input checked="" type="checkbox"/>	<input type="radio"/>
You can resolve the issue by selecting the index, editing the ItemDescription field, and selecting the SEARCHABLE option for the field.	<input type="radio"/>	<input checked="" type="checkbox"/>
You can resolve the issue by running the indexer.	<input checked="" type="checkbox"/>	<input type="radio"/>
You can resolve the issue by changing the query string in Search explorer to <code>bag of</code> to return the correct results	<input type="radio"/>	<input checked="" type="checkbox"/>

### Explanation

	Yes	No
You can resolve the issue by recreating the search index with the same settings for all fields except ItemDescription. Select the SEARCHABLE option for this field	<input checked="" type="radio"/>	<input type="radio"/>
You can resolve the issue by selecting the index, editing the ItemDescription field, and selecting the SEARCHABLE option for the field.	<input type="radio"/>	<input checked="" type="radio"/>
You can resolve the issue by running the indexer.	<input checked="" type="radio"/>	<input type="radio"/>
You can resolve the issue by changing the query string in Search explorer to <code>bag of</code> to return the correct results	<input type="radio"/>	<input checked="" type="radio"/>

Box 1: Yes

The ItemDescription field is not searchable.

Box 2: No

The ItemDescription field is not searchable, but we would need to recreate the index.

Box 3: Yes

An indexer in Azure Search is a crawler that extracts searchable data and metadata from an external Azure data source and populates an index based on field-to-field mappings between the index and your data source.

This approach is sometimes referred to as a 'pull model' because the service pulls data in without you having to write any code that adds data to an index.

Box 4: No

References:

<https://docs.microsoft.com/en-us/azure/search/search-what-is-an-index>

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

### NO.135 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.

Tools	Functionality	Tool
Logic Apps Designer	Edit B2B workflows	
Code View Editor	Edit definitions in JSON	
Enterprise Integration Pack	Visually add functionality	

**Answer:**

Tools	Functionality	Tool
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

Explanation

## Functionality

## Tool

Edit B2B workflows

Enterprise Integration Pack

Edit definitions in JSON

Code View Editor

Visually add functionality

Logic Apps Designer

### Box 1: Enterprise Integration Pack

After you create an integration account that has partners and agreements, you are ready to create a business to business (B2B) workflow for your logic app with the Enterprise Integration Pack.

### Box 2: Code View Editor

To work with logic app definitions in JSON, open the Code View editor when working in the Azure portal or in Visual Studio, or copy the definition into any editor that you want.

### Box 3: Logical Apps Designer

You can build your logic apps visually with the Logic Apps Designer, which is available in the Azure portal through your browser and in Visual Studio.

### References:

<https://docs.microsoft.com/en-us/azure/logic-apps/logic-apps-enterprise-integration-b2b>

<https://docs.microsoft.com/en-us/azure/logic-apps/logic-apps-author-definitions>

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

**NO.136** You are developing an ASP.NET Core Web API web service that uses Azure Application Insights to monitor performance and track events.

You need to enable logging and ensure that log messages can be correlated to events tracked by Application Insights.

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.

#### Code segments

IncludeEventId  
ServerFeatures  
LoggerFilterOptions  
ApplicationServices  
ApplicationInsightsLoggerOptions  
TrackExceptionsAsExceptionTelemetry

#### Answer Area

```
public class Startup
{
    .
    .
    public void ConfigureServices(IServiceCollection services)
    {
        services.AddOptions<ApplicationInsightsLoggerOptions>().Configure(o => o.IncludeEventId = true);
        services.AddMvc();
    }

    public void Configure(IApplicationBuilder app,
        IHostingEnvironment env, ILoggerFactory loggerFactory)
    {
        loggerFactory.AddApplicationInsights(app, LogLevel.Trace);
        app.UseMvc();
    }
}
```

#### Answer:

#### Code segments

IncludeEventId  
ServerFeatures  
LoggerFilterOptions  
ApplicationServices  
ApplicationInsightsLoggerOptions  
TrackExceptionsAsExceptionTelemetry

#### Answer Area

```
public class Startup
{
    .
    .
    public void ConfigureServices(IServiceCollection services)
    {
        services.AddOptions<ApplicationInsightsLoggerOptions>().Configure(o => o.IncludeEventId = true);
        services.AddMvc();
    }

    public void Configure(IApplicationBuilder app,
        IHostingEnvironment env, ILoggerFactory loggerFactory)
    {
        loggerFactory.AddApplicationInsights(app, ApplicationServices, LogLevel.Trace);
        app.UseMvc();
    }
}
```

#### Explanation

```
public class Startup
{
    .
    .
    public void ConfigureServices(IServiceCollection services)
    {
        services.AddOptions<ApplicationInsightsLoggerOptions>().Configure(o => o.IncludeEventId = true);
        services.AddMvc();
    }

    public void Configure(IApplicationBuilder app,
        IHostingEnvironment env, ILoggerFactory loggerFactory)
    {
        loggerFactory.AddApplicationInsights(app, ApplicationServices, LogLevel.Trace);
        app.UseMvc();
    }
}
```

#### Box 1: ApplicationInsightsLoggerOptions

If you want to include the EventId and EventName properties, then add the following to the ConfigureServices method:

```
services
    .AddOptions<ApplicationInsightsLoggerOptions>()
    .Configure(o => o.IncludeEventId = true);
```

#### Box 2: IncludeEventID

**Box 3: ApplicationServices**

In Asp.Net core apps it turns out that trace logs do not show up in Application Insights out of the box. We need to add the following code snippet to our Configure method in Startup.cs:

```
loggerFactory.AddApplicationInsights(app.ApplicationServices, logLevel); References:  
https://blog.computedcloud.com/enabling-application-insights-trace-logging-in-asp-net-core/
```

**NO.137** Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution. After you answer a question in this 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.

**A.** Yes

**B.** No

**Answer:** A

Explanation

References:

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

**NO.138** Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while others might not have a correct solution. After you answer a question in this 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.

**A.** Yes

**B.** No

**Answer:** B

Explanation

References:

<https://docs.microsoft.com/en-us/azure/event-grid/event-filtering>

**NO.139** You develop a news and blog content delivery 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.

## 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 =
@""">" +
@"New item to view" + @"</text></binding></visual></toast>";
try
{
var result=
await hub.
```

SendWindowsNativeNotificationAsync
SubmitNotificationHubJobAsync
ScheduleNotificationAsync
SendAppleNativeNotificationAsync

```
...
}
catch (System.Exception ex)
{
...
}
...
}
```

**Answer:**

## Answer Area

```
string notificationHubName = "contoso_hub";
string notificationHubConnection = "connection_string";
hub= NotificationHubClient.CreateClientFromConnectionString(notificationHubConnection, notificationHubName);
var result= await hub.SendWindowsNativeNotificationAsync(windowsToastPayload);
```

The screenshot shows a C# code editor with several intellisense dropdowns displayed:

- A top-level dropdown for `NotificationHubClient` contains: `NotificationHubClient`, `NotificationHubClientSettings`, `NotificationHubJob`, and `NotificationDetails`.
- A dropdown for `NotificationHubClient` contains: `GetInstallation`, `CreateClientFromConnectionString`, `CreateOrUpdateInstallation`, and `PatchInstallation`.
- A dropdown for `NotificationHubClient` (under `NotificationHubClient`) contains: `SendWindowsNativeNotificationAsync`, `SubmitNotificationHubJobAsync`, `ScheduleNotificationAsync`, and `SendAppleNativeNotificationAsync`.
- A dropdown for `windowsToastPayload` contains: `(windowsToastPayload)`.

```
(notificationHubConnection, notificationHubName);
string windowsToastPayload =
@""""+
@"New item to view" + @"</text></binding></visual></toast>";
try
{
    var result=
        await hub.
```

...  
}  
catch (System.Exception ex)  
{  
 ...  
}  
...  
}

Explanation

```

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 =
@""">" +
@""New item to view" + @""</text></binding></visual></toast>";
try
{
var result=
await hub.
SendWindowsNativeNotificationAsync
SubmitNotificationHubJobAsync
ScheduleNotificationAsync
SendAppleNativeNotificationAsync
(windowsToastPayload);
}

```

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); References:

<https://docs.microsoft.com/en-us/azure/notification-hubs/notification-hubs-push-notification-registration-manage>

<https://github.com/MicrosoftDocs/azure-docs/blob/master/articles/app-service-mobile/app-service-mobile-windo>

**NO.140** You are creating a script that will run a large workload on an Azure Batch pool. Resources will be reused and do not need to be cleaned up after use.

You have the following parameters:

You need to write an Azure CLI script that will create the jobs, tasks, and the pool.

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

**Command segments**

```
az batch pool create
--id mypool --vm-size Standard_A1_v2
--target-dedicated-nodes 2
--image $image
--node-agent-sku-id $sku
```

**Answer Area**

```
az batch job
create
--id myjob
--pool-id mypool
```



```
for i in {1..$numberOfJobs}
do
```

```
az batch task create
--task-id mytask$i
--job-id myjob
--command-line $script
```

**Answer:****Command segments**

```
az batch pool create
--id mypool --vm-size Standard_A1_v2
--target-dedicated-nodes 2
--image $image
--node-agent-sku-id $sku
```

**Answer Area**

```
az batch pool create
--id mypool --vm-size Standard_A1_v2
--target-dedicated-nodes 2
--image $image
--node-agent-sku-id $sku
```

```
az batch job
create
--id myjob
--pool-id mypool
```



```
for i in {1..$numberOfJobs}
do
```

```
az batch task create
--task-id mytask$i
--job-id myjob
--command-line $script
```

```
az batch task create
--task-id mytask$i
--job-id myjob
--command-line $script
```

```
az batch job
create
--id myjob
--pool-id mypool
```

```
for i in {1..$numberOfJobs}
do
```

**Explanation**

```
az batch pool create
--id mypool --vm-size Standard_A1_v2
--target-dedicated-nodes 2
--image $image
--node-agent-sku-id $sku
```

```
) az batch task create
--task-id mytask$i
--job-id myjob
--command-line $script
```

```
) az batch job
create
--id myjob
--pool-id mypool
```

```
for i in {1..$numberOfJobs}
do
```

Step 1: az batch pool create

# Create a new Linux pool with a virtual machine configuration.

```
az batch pool create \
--id mypool \
--vm-size Standard_A1 \
--target-dedicated 2 \
--image canonical:ubuntuserver:16.04-LTS \
--node-agent-sku-id "batch.node.ubuntu 16.04"
```

Step 2: az batch job create

# Create a new job to encapsulate the tasks that are added.

```
az batch job create \
--id myjob \
--pool-id mypool
```

Step 3: az batch task create

# Add tasks to the job. Here the task is a basic shell command.

```
az batch task create \
--job-id myjob \
--task-id mytask1
```

--task-id task1 \

--command-line "/bin/bash -c 'printenv AZ\_BATCH\_TASK\_WORKING\_DIR'"

Step 4: for i in {1..\$numberOfJobs} do

References:

<https://docs.microsoft.com/bs-latn-ba/azure/batch/scripts/batch-cli-sample-run-job>

PDFDUMPS.COM