

AZ-203 Dumps

Developing Solutions for Microsoft Azure

<https://www.certleader.com/AZ-203-dumps.html>



NEW QUESTION 1

Create a DataContainer that contains the documents which must be added.

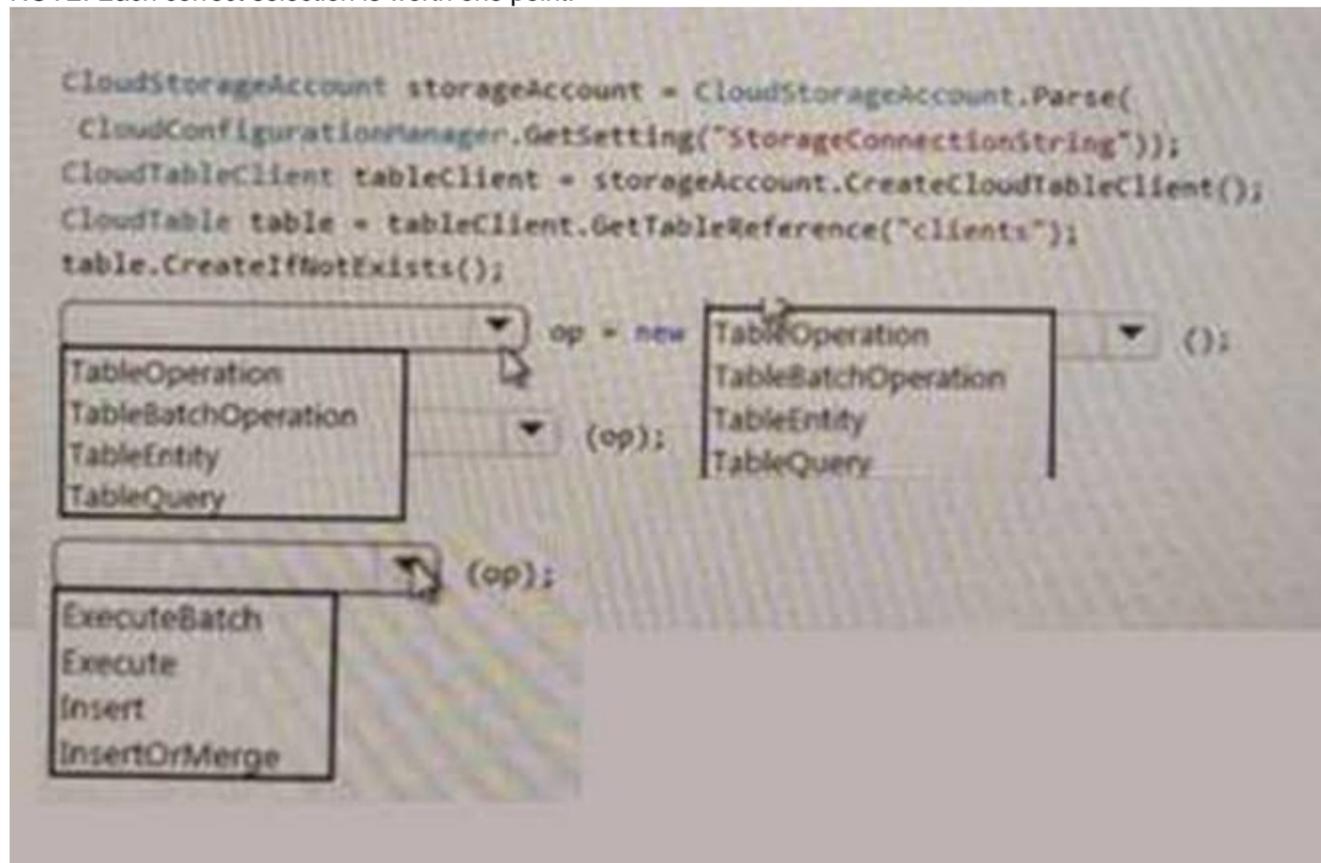
- A. Mastered
- B. Not Mastered

Answer: A

NEW QUESTION 2

HOTSPOT

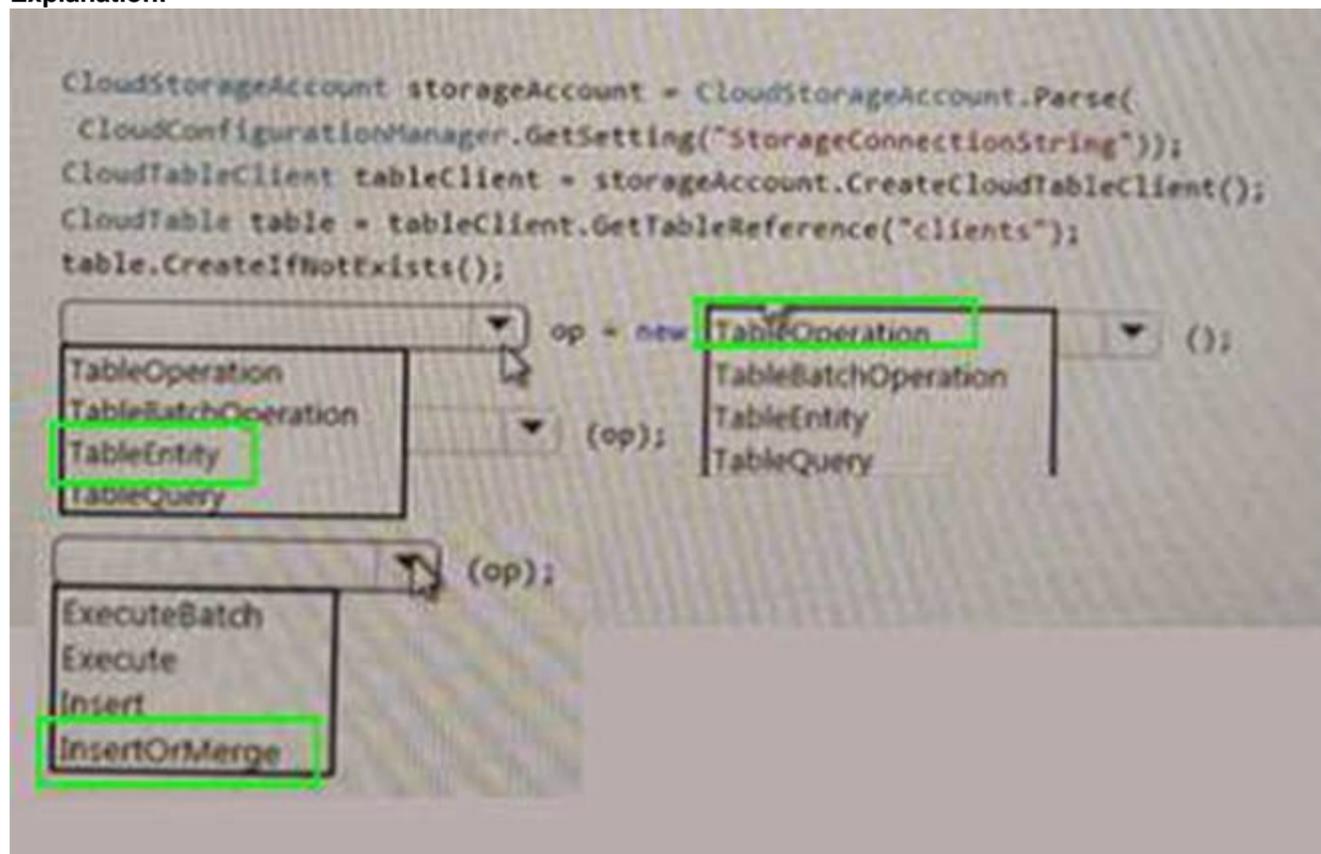
You are developing a data storage solution for a social networking app. The solution requires a mobile app that stores user information using Azure Table Storage. You need to develop code that can insert multiple sets of user information. How should you complete the code? To answer, select the appropriate options in the answer area. NOTE: Each correct selection is worth one point.



- A. Mastered
- B. Not Mastered

Answer: A

Explanation:



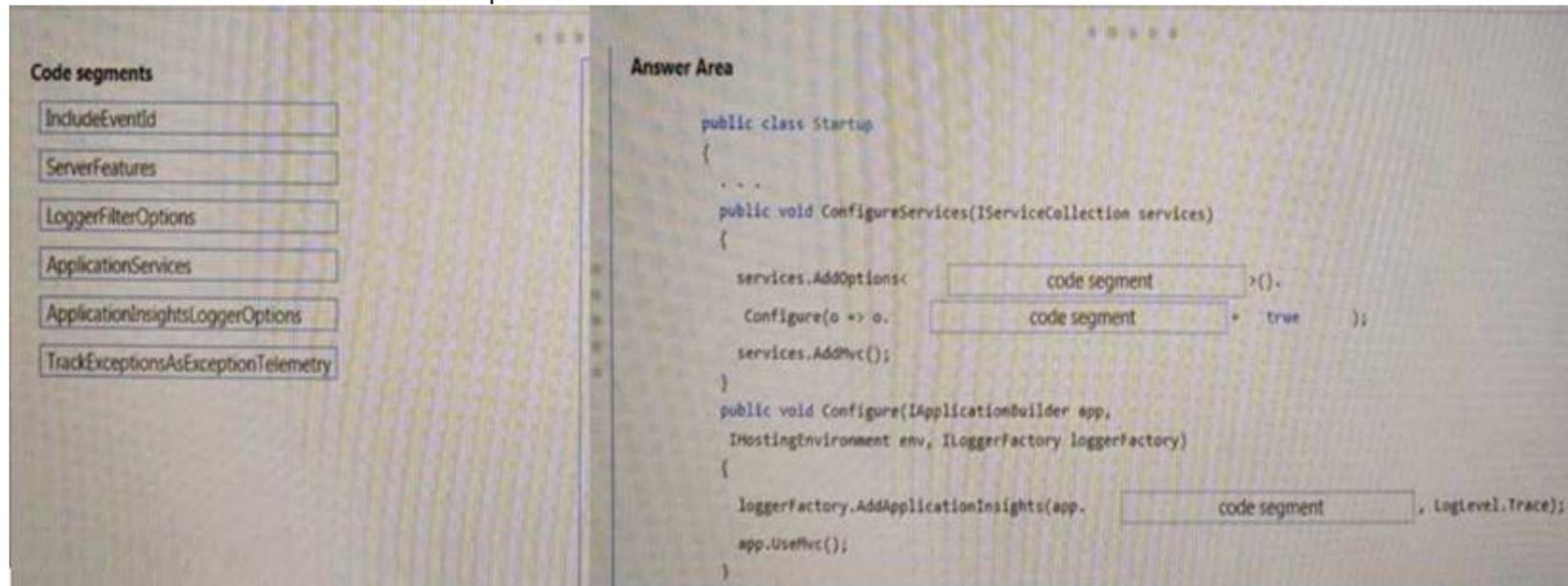
NEW QUESTION 3

DRAG DROP

You are developing an ASP.NET Core Web API web service that uses Azure Application Insights to monitor performance and trace 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.



- A. Mastered
- B. Not Mastered

Answer: A

Explanation:



NEW QUESTION 4

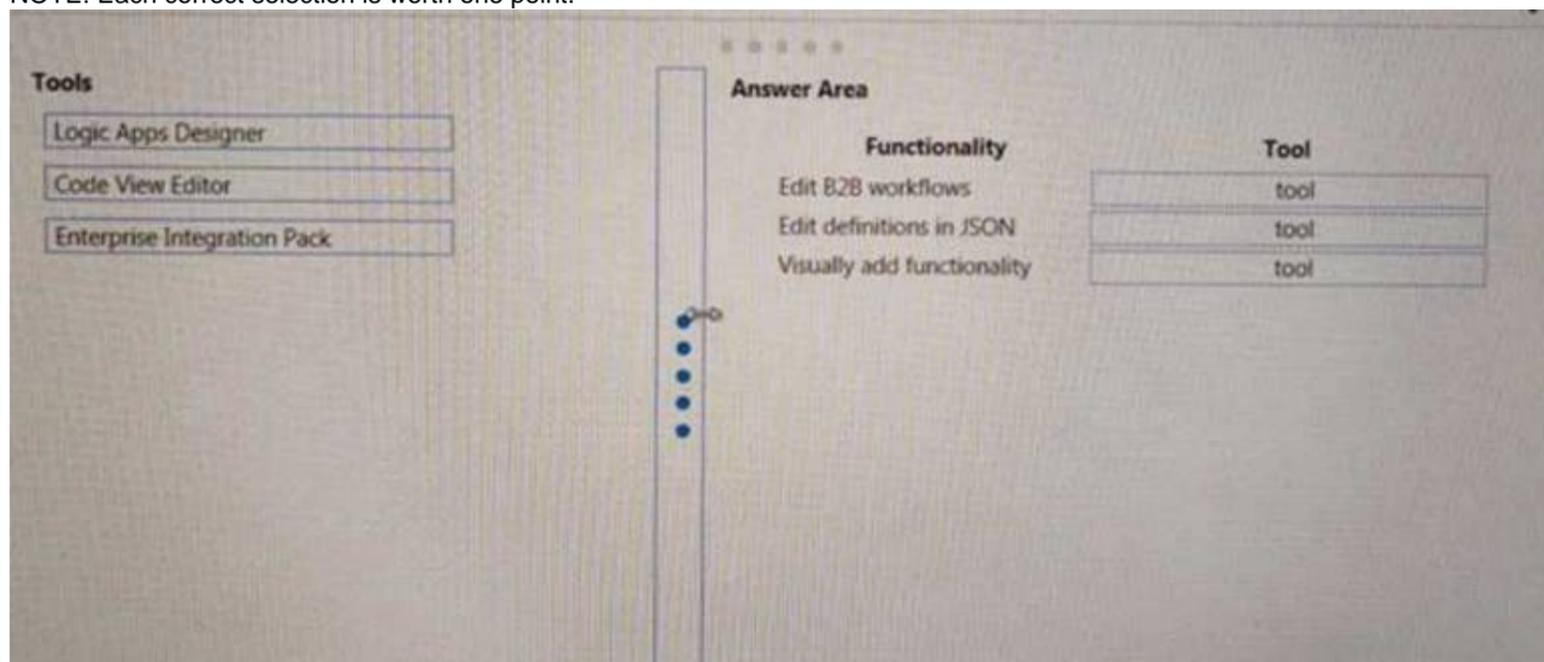
DRAG DROP

You manage several existing Logic Apps.

You need to change definitions, add new logic and optimize these apps on a regular basis.

What should you use? To answer, drag the appropriate tools to the coned 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.

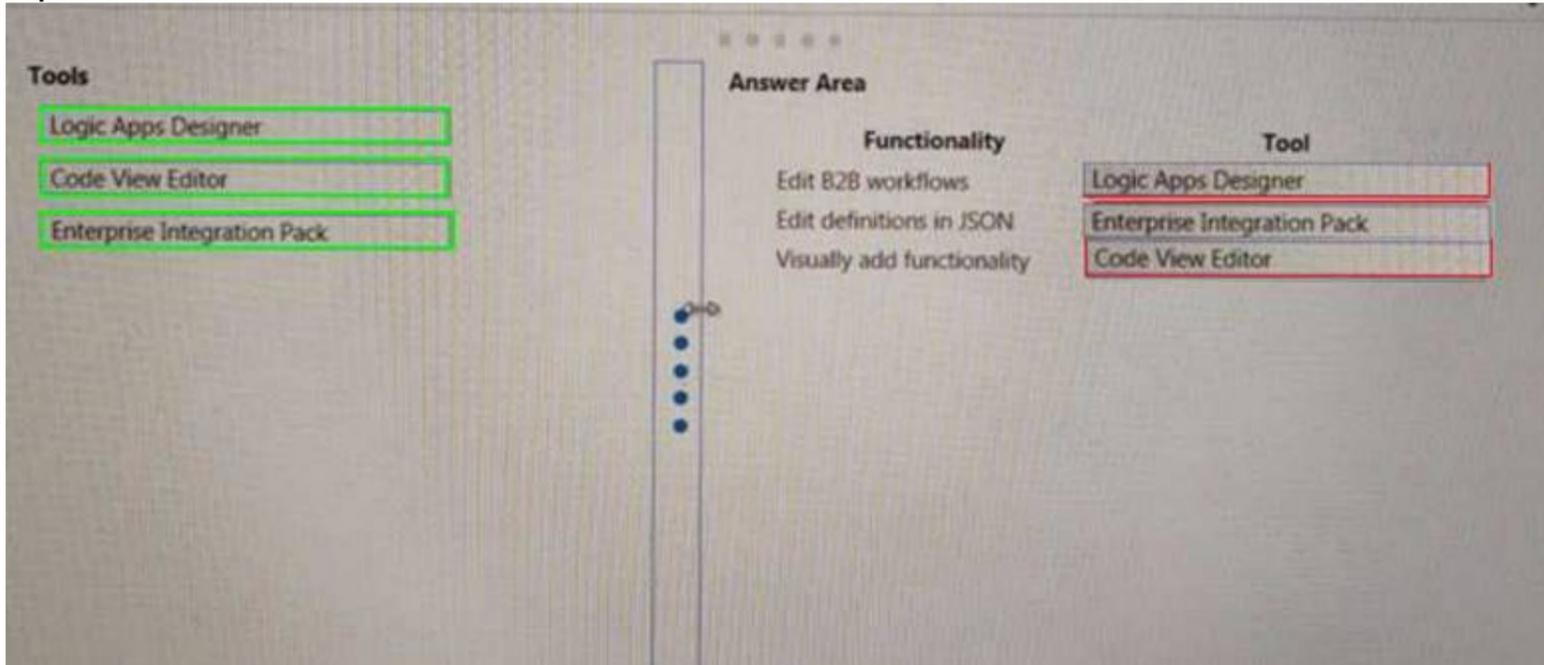


- A. Mastered

B. Not Mastered

Answer: A

Explanation:



NEW QUESTION 5

HOTSPOT

You are creating an app that uses Event Grid to connect with other services. Your app's event data will be sent to a serverless function that checks compliance. This function is maintained by your company.

You write a new event subscription at the scope of your resource. The event must be invalidated after 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 the answer area.

NOTE: Each correct selection is worth one point

Authentication	Type
WebHook event delivery	<div style="border: 1px solid black; padding: 5px;"> <div style="text-align: right; margin-bottom: 5px;">▼</div> <div style="padding: 5px;"> <p>SAS tokens</p> <p>Key authentication</p> <p>JWT token</p> </div> </div>
Topic publishing	<div style="border: 1px solid black; padding: 5px;"> <div style="text-align: right; margin-bottom: 5px;">▼</div> <div style="padding: 5px;"> <p>ValidationCode handshake</p> <p>ValidationURL handshake</p> <p>Management Access Control</p> </div> </div>

A. Mastered
B. Not Mastered

Answer: A

Explanation:

Box 1: SAS tokens

Custom topics use either Shared Access Signature (SAS) or key authentication. Microsoft recommends SAS, but key authentication provides simple programming, and is compatible with many existing webhook publishers.

In this case we need the expiration time provided by SAS tokens. Box 2: ValidationCode handshake

Event Grid supports two ways of validating the subscription:

ValidationCode handshake (programmatic) and ValidationURL handshake (manual).

If you control the source code for your endpoint, this method is recommended. Incorrect Answers:

ValidationURL handshake (manual): In certain cases, you can't access the source code of the endpoint to implement the ValidationCode handshake. For example, if you use a third-party service (like Zapier or IFTTT), you can't programmatically respond with the validation code.

References:

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

NEW QUESTION 6

DRAG DROP

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

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

The system has the following requirements for subscriptions:

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

You need to implement filtering and maximize throughput while evaluating filters. Which filter types should you implement? To answer, drag the appropriate filter types to the correct subscriptions. Each filter type may be used once, more than once, or not at all. You may need to drag the split bar between panes or scroll to view content.

NOTE: Each correct selection is worth one point.

Filter types

- SQLFilter
- CorrelationFilter
- No Filter

Answer Area

Subscription	Filter type
FutureOrders	<input type="text"/>
HighPriorityOrders	<input type="text"/>
InternationalOrders	<input type="text"/>
HighQuantityOrders	<input type="text"/>
AllOrders	<input type="text"/>

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

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>

NEW QUESTION 7

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.

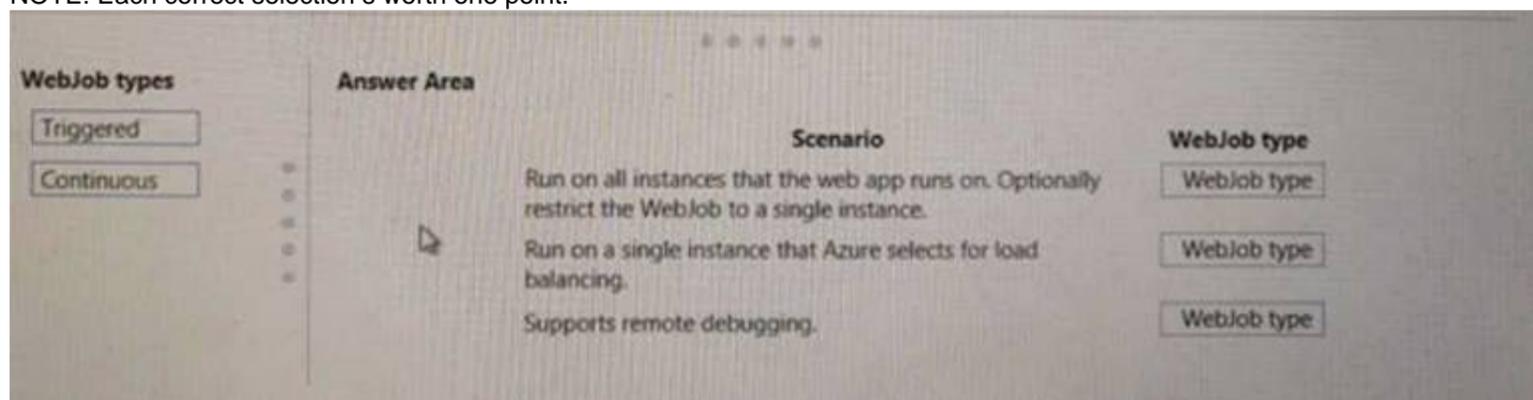
- A. Certificate Authentication
- B. Basic Authentication
- C. OAuth Client Credential Grant
- D. Digest Authentication

Answer: AC

NEW QUESTION 8

DRAG DROP

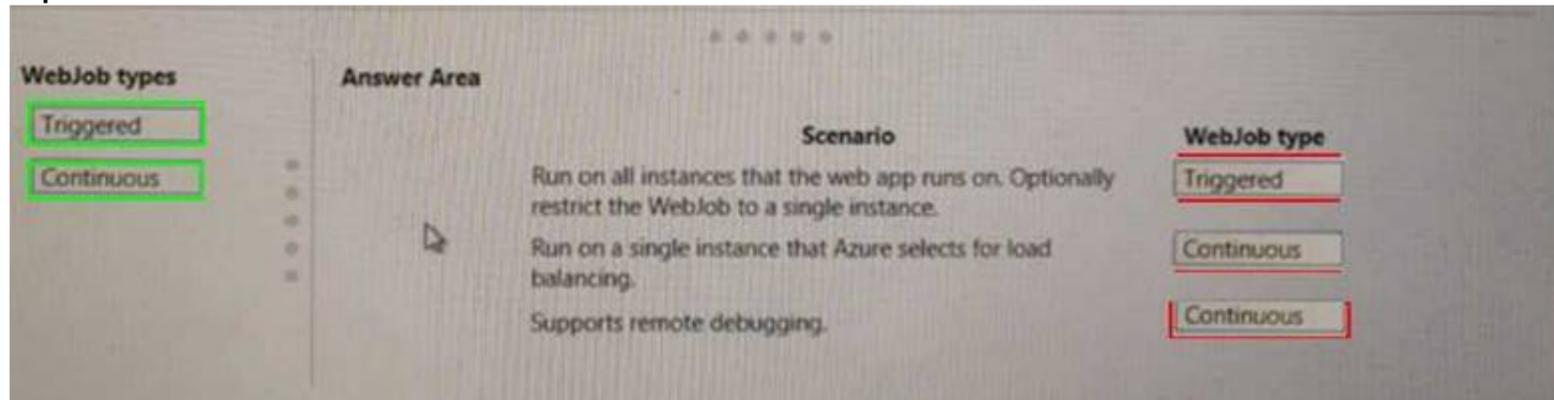
You are developing Azure WebJobs. You need to recommend a WebJob type for each scenario. Which WebJob type should you recommend? To answer, drag the appropriate WebJob types to the correct scenarios. Each WebJob type may be used once more than once, or not at all. You may need to drag the split bar between panes or scroll to view content NOTE: Each correct selection is worth one point.



- A. Mastered
- B. Not Mastered

Answer: A

Explanation:



NEW QUESTION 9

You need to meet the LabelMaker security requirement. What should you do?

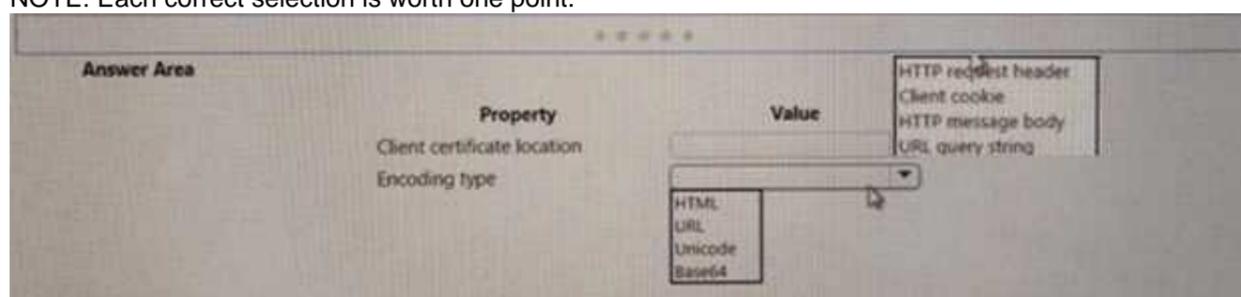
- A. Create a conditional access policy and assign it to the Azure Kubernetes Service cluster
- B. Place the Azure Active Directory account into an Azure AD group
- C. Create a ClusterRoleBinding and assign it to the group.
- D. Create a Microsoft Azure Active Directory service principal and assign it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

Answer: D

NEW QUESTION 10

HOTSPOT

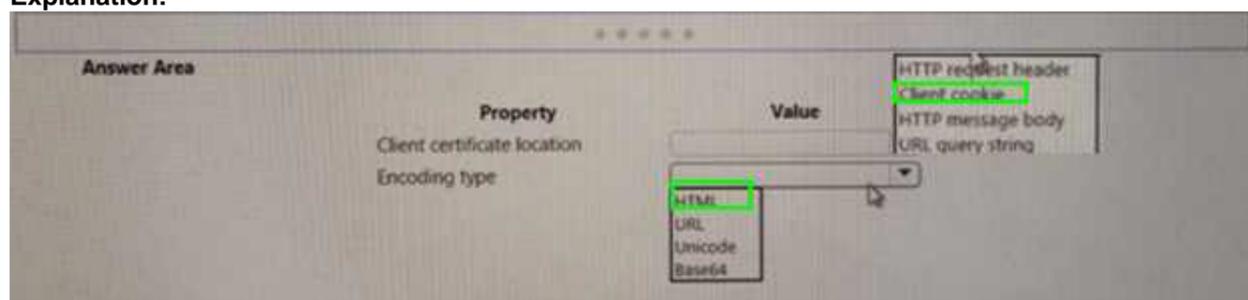
You are developing an Azure Web App. You configure TLS mutual authentication for the web app. You need to validate the client certificate in the web app. To answer, select the appropriate options in the answer area. NOTE: Each correct selection is worth one point.



- A. Mastered
- B. Not Mastered

Answer: A

Explanation:



NEW QUESTION 10

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 correct 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: BD

NEW QUESTION 15

HOTSPOT

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 gameName)
2 {
3     TableQuery<DynamicTableEntity> query = new TableQuery<DynamicTableEntity>().Select(new string[] { "Score" })
         .Where(TableQuery.GenerateFilterConditionForInt("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     {
7         Console.WriteLine($"{scoreItem.Key} {scoreItem.Value}");
8     }
9
10 public class PlayerScore : TableEntity
11 {
12     public PlayerScore(string gameId, string playerId, int score, long timePlayed)
13     {
14         PartitionKey = gameId;
15         RowKey = playerId;
16         Score = score;
17         TimePlayed = timePlayed;
18     }
19     public int Score { get; set; }
20     public long TimePlayed { get; set; }
21 }

```

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

NOTE: Each correct selection is worth one point

Answer Area

	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"/>

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

Answer Area



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

NEW QUESTION 17

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.AddRuleAsync(new RuleDescription(RuleDescription.DefaultRuleName, new TrueFilter()));
- B. subscriptionClient.RegisterMessageHandler(ProcessMessagesAsync, BiessageHandlerOptions);
- C. subscriptionClient « new SubscriptionClient(ServiceBusConnectionString, TopicName, SubscriptionName);
- D. await subscriptionClient.CloseAsync();

Answer: D

NEW QUESTION 19

DRAG DROP

You develop a gateway solution for a public facing news API.

The news API back end is implemented as a RESTful service and hosted in an Azure App Service instance.

You need to configure back-end authentication for the API Management service instance.

Which target and gateway credential type should you use? To answer, drag the appropriate values to the correct parameters. Each value may be used once, more than once, or not at all. You may need to drag the split bar between panes or scroll to view content.

NOTE: Each correct selection is worth one point.

Azure Resource	<table border="0" style="width: 100%;"> <thead> <tr> <th style="text-align: left;">Configuration parameter</th> <th style="text-align: left;">Value</th> </tr> </thead> <tbody> <tr> <td>Target</td> <td>value</td> </tr> <tr> <td>Gateway credentials</td> <td>value</td> </tr> </tbody> </table>	Configuration parameter	Value	Target	value	Gateway credentials	value
Configuration parameter		Value					
Target		value					
Gateway credentials		value					
HTTP(s) endpoint							
Basic							
Client cert							

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

Azure Resource	<table border="0" style="width: 100%;"> <thead> <tr> <th style="text-align: left;">Configuration parameter</th> <th style="text-align: left;">Value</th> </tr> </thead> <tbody> <tr> <td>Target</td> <td style="border: 1px solid red;">Azure Resource</td> </tr> <tr> <td>Gateway credentials</td> <td style="border: 1px solid red;">HTTP(s) endpoint</td> </tr> </tbody> </table>	Configuration parameter	Value	Target	Azure Resource	Gateway credentials	HTTP(s) endpoint
Configuration parameter		Value					
Target		Azure Resource					
Gateway credentials		HTTP(s) endpoint					
HTTP(s) endpoint							
Basic							
Client cert							

NEW QUESTION 21

HOTSPOT

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.

```

{
  [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 Description { get; set; }
  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; }

  [IsSearchable, IsFilterable, IsSortable, IsFacetable]
  [IsFilterable, IsFacetable, Required]
  [IsSearchable]
  [IsSearchable, Required]
  public string Description { get; set; }

  [IsSearchable, IsFilterable, IsSortable, IsFacetable]
  [IsFilterable, IsFacetable, Required]
  [IsSearchable]
  [IsSearchable, Required]
  public double Rating { get; set; }
}

```

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

```

{
  [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 Description { get; set; }
  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; }

  [IsSearchable, IsFilterable, IsSortable, IsFacetable]
  [IsFilterable, IsFacetable, Required]
  [IsSearchable]
  [IsSearchable, Required]
  public string Description { get; set; }

  [IsSearchable, IsFilterable, IsSortable, IsFacetable]
  [IsFilterable, IsFacetable, Required]
  [IsSearchable]
  [IsSearchable, Required]
  public double Rating { get; set; }
}

```

NEW QUESTION 26

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.

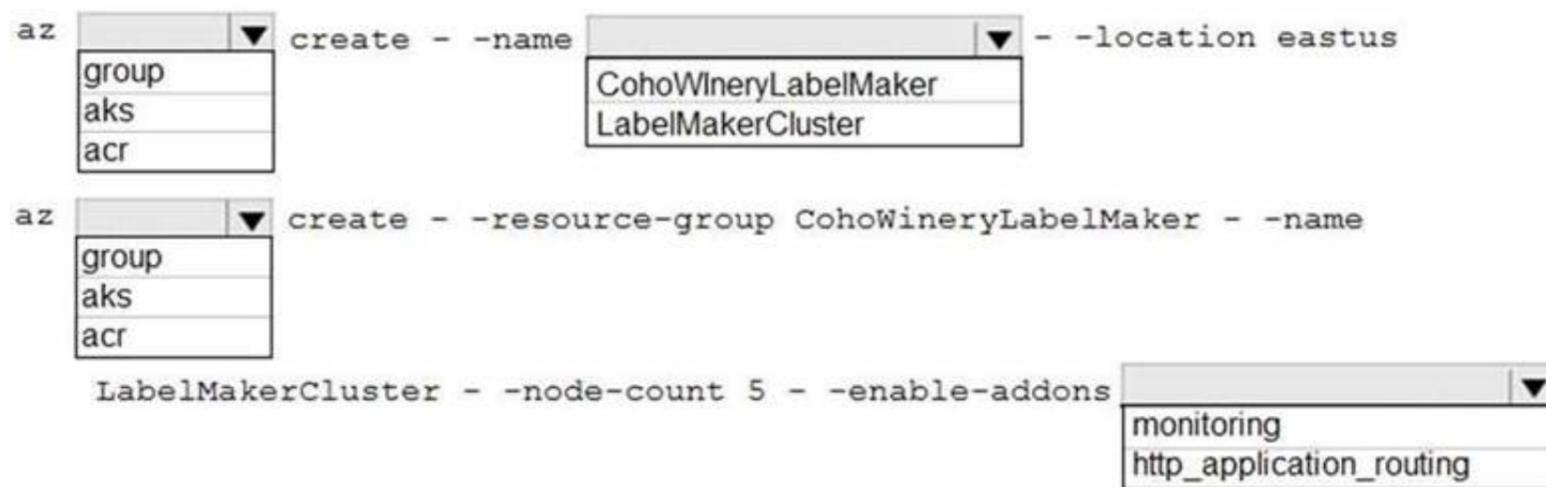
- A. Telemetry.Context.Operation.Id
- B. Telemetry.Context.Cloud.RoleInstance
- C. Telemetry.Id
- D. Telemetry.ContextSession.Id
- E. Telemetry.Name

Answer: BC

NEW QUESTION 31

HOTSPOT

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.



- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

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: CohoWinterLabelMaker

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.

NEW QUESTION 36

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/>

NEW QUESTION 38

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: CDF

Explanation:

Case Study: 2

Litware Inc

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

```

Database.cs

```

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

```

ReceiptUploader.cs

```

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

NEW QUESTION 43

HOTSPOT

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=datastore.database.windows.net;Initial Catalog=expense;

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

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

Box 1: Integrated Security=SSPI

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

For SqlConnection 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>

NEW QUESTION 45

HOTSPOT

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.

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.DefaultRequestOptions.RequireEncryption = x;
cloudBlobClient.AuthenticationScheme = x;
cloudBlobClient.DefaultRequestOptions.RequireEncryption = x;
cloudBlobClient.DefaultRequestOptions.EncryptionPolicy = x;
cloudBlobClient.SetServiceProperties(new ServiceProperties(deleteRetentionPolicy: x));

- A. Mastered
- B. Not Mastered

Answer: A

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.DefaultRequestOptions.RequireEncryption = x;
cloudBlobClient.AuthenticationScheme = x;
cloudBlobClient.DefaultRequestOptions.RequireEncryption = x;
cloudBlobClient.DefaultRequestOptions.EncryptionPolicy = x;
cloudBlobClient.SetServiceProperties(new ServiceProperties(deleteRetentionPolicy: x));

NEW QUESTION 50

DRAG DROP

You need to ensure disaster recovery requirements are met. What code should you add at line PC16?

To answer, drag the appropriate code fragments to the correct locations. Each code fragment may be used once, more than once, or not at all. You may need to drag the split bar between panes or scroll to view content.

NOTE: Each correct selection is worth one point.

Values

-
-
-
-
-
-

Answer Area

```
var copyOptions = new CopyOptions { };
var context = new Value (source, destination) => Task.From
context: Value (source, destination) => Task.From
await TransferManager.CopyAsync(blob, GetDRBlob(blob), isServiceCopy:
, context: context, options:copyOptions);copyOptions, context);
```

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

Values

-
-
-
-
-
-

Answer Area

```
var copyOptions = new CopyOptions { };
var context = new true (source, destination) => Task.From
context: SingleTransferContext (source, destination) => Task.From
await TransferManager.CopyAsync(blob, GetDRBlob(blob), isServiceCopy:
, context: context, options:copyOptions);copyOptions, context);
```

NEW QUESTION 51

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:

Case Study: 3
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 = JObject.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 }
```

NEW QUESTION 53

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

NEW QUESTION 55

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.

NEW QUESTION 57

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

NEW QUESTION 59

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>

NEW QUESTION 61

HOTSPOT

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	<div style="border: 1px solid black; padding: 5px;"> <div style="background-color: #f0f0f0; padding: 2px; display: flex; justify-content: space-between; align-items: center;"> ▼ </div> <div style="border-top: 1px solid black; border-bottom: 1px solid black; padding: 2px;">ADO.NET</div> <div style="border-top: 1px solid black; border-bottom: 1px solid black; padding: 2px;">Entity Framework</div> <div style="border-top: 1px solid black; border-bottom: 1px solid black; padding: 2px;">Entity Framework Core</div> <div style="border-top: 1px solid black; border-bottom: 1px solid black; padding: 2px;">WCF Data Services</div> </div>
Workflow	<div style="border: 1px solid black; padding: 5px;"> <div style="background-color: #f0f0f0; padding: 2px; display: flex; justify-content: space-between; align-items: center;"> ▼ </div> <div style="border-top: 1px solid black; border-bottom: 1px solid black; padding: 2px;">Model first</div> <div style="border-top: 1px solid black; border-bottom: 1px solid black; padding: 2px;">Database first</div> <div style="border-top: 1px solid black; border-bottom: 1px solid black; padding: 2px;">Code first</div> </div>

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

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-an-entity-framework-data-model-for-an-asp-net-mvc-application>

NEW QUESTION 62

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>

NEW QUESTION 64

.....

Thank You for Trying Our Product

* 100% Pass or Money Back

All our products come with a 90-day Money Back Guarantee.

* One year free update

You can enjoy free update one year. 24x7 online support.

* Trusted by Millions

We currently serve more than 30,000,000 customers.

* Shop Securely

All transactions are protected by VeriSign!

100% Pass Your AZ-203 Exam with Our Prep Materials Via below:

<https://www.certleader.com/AZ-203-dumps.html>