Az 204 final Exam3

Done by Ahmed Fouad
Ahmed Fouad | LinkedIn

Nokhba Academy FB page

https://shorturl.at/UUDf2

1. You are managing Azure Batch jobs and pools programmatically, suppose you submit a newly configured Batch job to the Batch service for processing. Assuming a valid Batch client instance (batchClient) along with relevant parameters and IDs are defined, identify the suitable pair consisting of a Class and its corresponding Method to accomplish this objective effectively.

A) JobOperations | SubmitJob()

B) BatchClient | StartJob()

C) CloudJob | SubmitAsync()

D) JobOperations | CommitJobAsync(CloudJob, CancellationToken)

Option: D

Feedback(if correct):- Once the Batch job is created using the JobOperations.CreateJob() method, you can associate the required properties and details with the job before finally committing it to the Batch service using the JobOperations.CommitJobAsync(job, CancellationToken.None). The CommitJobAsync() method takes care of persisting and scheduling the job onto the designated Batch pool within the Batch service, ensuring seamless integration and management of your Batch workloads.

The most suitable pair of classes and methods to submit a newly configured Batch job to the Batch service for processing, assuming a valid Batch client instance (batchClient) and relevant parameters and IDs are defined, is:

D) JobOperations | CommitJobAsync(CloudJob, CancellationToken)

Feedback(if wrong):-

JobOperations | SubmitJob(): This method has been deprecated and replaced by `CommitJobAsync` in newer versions of the Azure Batch SDK.

BatchClient | StartJob(): The `StartJob` method is used to start a previously submitted job that is in the "Idle" state. It's not for submitting new jobs.

CloudJob | SubmitAsync(): The `CloudJob` class itself doesn't have a `SubmitAsync` method. Submitting a job requires interacting with the Batch service, which is done through the `JobOperations` class.

Here's an example code snippet demonstrating how to use this method:

```csharp
// Create a new CloudJob object with necessary properties
CloudJob job = new CloudJob(...);

// Submit the job using JobOperations.CommitJobAsync
await batchClient.JobOperations.CommitJobAsync(job, cancellationToken);

// Handle any potential exceptions
catch (BatchException ex)
{
   // Handle the exception
}
```

Skill Mapping:

Skill: Azure Developer Certification (AZ-204)

Subskill: Develop Azure compute solutions

Competency: Submit Batch jobs to the Batch service programmatically

Bloom's Taxonomy Level: Application

Difficulty Level: Intermediate

2. Imagine you are engineering a software solution for an Internet of Things (IoT) implementation amassing telemetry from countless gadgets. This cutting-edge apparatus employs Azure Parallel Processing to scrutinize incoming data streams simultaneously. To ensure high-performance and cost-efficient computation, designate adequate Azure Batch resources to administer the escalating workload. Which measure should you adopt?

A. Generate a Python script involving the "azure.mgmt.batch" library and its "PoolAddParameter" class to structure the Azure Batch pool

B. Formulate an Azure Resource Manager (ARM) blueprint incorporating the resource type "Microsoft.Batch/pools" to delineate the Azure Batch pool layout

C. Construct a fresh Azure Batch account in the Azure interface to commandeer the computer assets

D. Orchestrate a C# environment, taking advantage of the method "BatchClient.PoolOperations.CreatePoolAsync()" to materialize the Azure Batch pool

Skill Mapping:

Skill: Azure Developer Certification (AZ-204)

Subskill: Develop Azure compute solutions

Competency: Submit Batch jobs to the Batch service programmatically

Bloom's Taxonomy Level: Application

Difficulty Level: Intermediate

1. You're tasked with registering a web application with Azure Active Directory (Azure AD) for authentication purposes.

Which three steps should you follow in sequence? Choose the correct order of actions.

A. Select Azure AD instance in App Registrations, then select New registration.

B. Create a new application, providing the name, account type, and redirect URL.

C. Add a Cryptographic key.

D. In Enterprise Applications, select New application.

Answer: A, B, C

Feedback(if correct):

To register a web application with Azure Active Directory (Azure AD) for authentication, you first need to select the Azure AD instance in the App Registrations section and then create a new registration. After providing the necessary details such as the application name, account type, and redirect URL, you should add a cryptographic key to ensure secure communication between the application and Azure AD.

Feedback(if wrong):

To register a web application with Azure Active Directory (Azure AD) for authentication, you should first select the Azure AD instance in the App Registrations section (A), then create a new registration by providing essential details like the application name, account type, and redirect URL (B), and finally, add a cryptographic key (C) to secure communication between the application and Azure AD. Option D, "In Enterprise Applications, select New application," is incorrect as it is not a step involved in registering a new application in Azure AD.

3. You are tasked with migrating Fourth Coffee's ASP.NET Core web application, which runs in Docker and is mapped to the www.fourthcoffee.com domain, to Azure. The goal is to provision an App Service Web App to host this Docker image and map the custom domain to the App Service web app. The resource group named FourthCoffeePublicWebResourceGroup already exists in the WestUS region and contains an App Service Plan named AppServiceLinuxDockerPlan.

Select all the correct steps required to develop the solution using Azure CLI commands:

A) `az webapp create`: This command is used to create a web app in the specified App Service Plan.

B) `#bin/bash The appName is used when the webapp-name is created in step 2.`: This step might be necessary for scripting purposes but is not directly related to creating the web app.

C) `az webapp config hostname add`: This command is used to add a custom domain hostname to the web app.

D) `az webapp config container set`: This command is used to configure the Docker container settings for the web app.

Answer: A, C, D

Feedback (if correct):

Here are the correct steps required to develop the solution using Azure CLI commands:

A) `az webapp create`: This command creates a web app in the specified App Service Plan. Since a resource group and an App Service Plan have already been created, you just need to specify the names of these resources during the creation process.

C) `az webapp config hostname add`: After creating the web app, you need to associate the custom domain (www.fourthcoffee.com) with the web app. This command allows you to add the custom domain to the web app.

D) `az webapp config container set`: Once the custom domain is associated with the web app, you need to configure the web app to use the Docker image hosting the ASP.NET Core application. This command sets the Docker container settings for the web app, allowing you to specify the location of the Docker image in Docker Hub.

So the final answer consists of options A, C, and D arranged in ascending chronological order.

Feedback (if incorrect):

Step B (`#bin/bash The appName is used when the webapp-name is created in step 2.`) is not a valid Azure CLI command. It appears to be a comment or placeholder within a bash script and is not directly related to Azure CLI commands.

Skill: Azure Developer Certification (AZ-204)

Subskill: Develop Azure Compute Solutions

Competency: Azure web App

Difficulty Level: comprehension

Bloom's Taxonomy Level: Application

4. As a cloud architect responsible for designing a highly available and globally accessible platform on Azure, you aim to implement automatic failover capabilities and global HTTP load balancing to ensure optimal performance and resilience for your services. Considering the scenario outlined above, which combination of Azure services would you recommend for achieving both automatic failover capabilities and global HTTP load balancing for your platform?

A) Azure Traffic Manager and Azure Virtual Machine Scale Set

B) Azure Application Gateway and Azure CDN

C) Azure Front Door and Azure Traffic Manager

D) Azure Application Gateway and Azure VMSS

Answer: C

Feedback(if correct):

The correct answer is C) Azure Front Door and Azure Traffic Manager. Azure Front Door provides global HTTP load balancing and automatic failover capabilities by routing traffic to the nearest healthy backend. Azure Traffic Manager complements this by offering DNS-based traffic routing to the closest available endpoint, ensuring high availability and optimal performance.

Feedback(if wrong):

A) Azure Traffic Manager and Azure Virtual Machine Scale Set: While Azure Traffic Manager provides DNS-based traffic routing, Azure Virtual Machine Scale Set (VMSS) alone does not offer automatic failover capabilities or global HTTP load balancing.

B) Azure Application Gateway and Azure CDN: Azure Application Gateway offers HTTP load balancing at the application level, but it does not inherently provide global load balancing or automatic failover. Azure CDN primarily focuses on content delivery and acceleration, rather than traffic routing and failover.

D) Azure Application Gateway and Azure VMSS: Similar to option B, while Azure Application Gateway offers HTTP load balancing, Azure VMSS lacks built-in global HTTP load balancing and automatic failover features.

Skill Mapping:

Skills: Azure Developer Certification (AZ-204)

Subskills: Develop Azure compute solutions

Competencies: Azure Compute Solutions, Virtual Machines, Azure App Service, Azure Functions, Azure Kubernetes Service (AKS)

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

5. SetUniverse, an international logistics service, is expanding into limousine bookings. As part of this expansion, you are tasked with implementing Azure Search for the services listed in their solution. You create the index in Azure Search and need to import the service data using the Azure Search .NET SDK. Which of the following steps should be included in the solution to meet the goal?

A) Create a SearchIndexClient object to connect to the search index.

B) Create an IndexBatch that contains the documents to be added.

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

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

Answer: D

Feedback(if correct):-

Option D is the correct answer. This step involves calling the Documents.Index method of the SearchIndexClient and passing the IndexBatch to import the service data into Azure Search. It's essential to use Documents.Index for data import, not Documents.Suggest.

Feedback(if wrong):-

Option A is necessary to connect to the search index, but it is not sufficient on its own to import data. Option B involves creating an IndexBatch, but it needs to be passed to the Documents.Index method for data import. Option C is not required for importing data into Azure Search.

Skill Mapping:

Skills: Azure Developer Certification (AZ-204)

Subskills: Develop Azure compute solutions

Competencies: Azure Compute Solutions, Azure Functions, Implement IaaS solutions.

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

6. You are developing a web application that needs to search for keywords in project data using Azure Search. You need to implement the code that creates the object used to create indexes in the Azure Search service. Which two objects should you use?

A) SearchService

B) SearchIndexClient

C) SearchServiceClient

D) SearchCredentials

Answer: B, C

B) SearchIndexClient

The SearchIndexClient object is used to interact with the Azure Search service and perform operations related to indexes, such as creating, updating, and deleting indexes. It provides methods to manage the schema, documents, and other settings of the index.

C) SearchServiceClient

The SearchServiceClient object is used to manage the Azure Search service itself. It provides methods to create, delete, and manage indexes, as well as other service-level operations.

Feedback (if wrong):

A) SearchService

The SearchService object is not used to create indexes in the Azure Search service. It represents the Azure Search service itself and provides methods for managing the service, such as creating and deleting services.

D) SearchCredentials

The SearchCredentials object is used for authentication and authorization purposes when accessing the Azure Search service. It is not directly involved in creating indexes.

Skill Mapping:

Skill: Azure Search Implementation

Subskill: Develop Azure compute solutions

Competency: Azure Compute Solutions (Virtual Machines, Azure App Service, Azure Functions, Azure Kubernetes Service (AKS))

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

7. You are developing a Node.js application that requires preprocessing of CSS files before deployment to an Azure Web App. The preprocessing step involves running a script to compile Sass files into CSS. You plan to deploy this application to Azure Web App using continuous deployment from a GitHub repository. What are two possible methods to ensure that the Sass compilation script is executed before the Node.js application starts serving traffic on the Azure Web App? (Select two options)

A) Configure a PreBuild task in the package.json file of the Node.js application to run the Sass compilation script before building the application artifacts.

B) Add a startup script named "startup.sh" in the root directory of the application code, which executes the Sass compilation script and then starts the Node.js application.

C) Create a file named "precompile.sh" in the /scripts folder of the application, which automatically compiles Sass files when the Azure Web App starts.

D) Enable the "WEBSITE_RUN_FROM_PACKAGE" setting in the host.json file and include the path to the Sass compiler in the deployment package.

Answer: A, B

Feedback (if correct):

The correct answers for the given question are A) Configure a "prebuild" task in the package.json file of the Node.js application to run the Sass compilation script before building the application artifacts, and B) Add a startup script named "startup.sh" in the root directory of the application code, which executes the Sass compilation script and then starts the Node.js application.

A) Configuring a "prebuild" task in the package.json file allows you to define a script that will be executed before the application is built. By running the Sass compilation script as a "prebuild" task, the CSS files will be generated before the application artifacts are built.

B) Adding a startup script named "startup.sh" in the root directory of the application code allows you to define a script that will be executed when the application starts. By including the Sass compilation script in the startup script, the Sass files can be compiled before the Node.js application starts serving traffic.

Feedback(if wrong):-

C) Create a file named precompile.sh in the /scripts folder of the application, which automatically compiles Sass files when the Azure Web App starts.

This option is incorrect because Azure Web App does not automatically execute a specific file named "precompile.sh" when it starts. The startup script needs to be defined at the root level, not within a specific folder.

D) Enable the WEBSITE_RUN_FROM_PACKAGE setting in the host.json file and include the path to the Sass compiler in the deployment package.

This option is incorrect because the WEBSITE_RUN_FROM_PACKAGE setting in the host.json file is used for running Azure Functions from a deployment package, not for running a Node.js application with a Sass compilation script.

Skill Mapping:

Skills: Azure Developer Certification (AZ-204)

Subskills: Develop Azure compute solutions

Competencies: Azure Compute Solutions, Azure Functions, Azure Blob Storage

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

8. You are developing a web application that allows users to upload images to an Azure Blob Storage account named "MediaStorage." The application consists of two main features: image uploading and image processing. After images are uploaded, they need to be automatically resized and moved to a separate container named "ProcessedImages." Additionally, the original images must remain in the "MediaStorage" container. What is the most suitable approach to automatically resize and move the uploaded images to the "ProcessedImages" container while keeping the original images in the "MediaStorage" container?

A) Use Azure Functions triggered by blob storage events to resize and move the images to the "ProcessedImages" container.

B) Implement a virtual machine to monitor the "MediaStorage" container and manually resize and move the images to the "ProcessedImages" container.

C) Utilize the Azure Storage REST API to resize the images on the client-side and directly upload them to the "ProcessedImages" container.

D) Utilize the AzCopy tool with the Sync option to continuously synchronize the contents of the "MediaStorage" container with the "ProcessedImages" container.

Answer: A

Feedback(if correct):-

Option A aligns with the scenario and is the most suitable approach for automatically resizing and moving uploaded images to the "ProcessedImages" container. Azure Functions can be triggered by blob storage events, such as new blob creation, which allows for seamless processing of images as they are uploaded to the "MediaStorage" container. This approach enables efficient, scalable, and event-driven image processing without manual intervention.

Feedback(if wrong):-

Option B is incorrect because manually monitoring and resizing images on a virtual machine is not efficient or scalable, especially in a cloud-based environment where serverless solutions like Azure Functions are available.

Option C is not ideal because client-side resizing would require additional processing resources on the client devices, potentially causing performance issues and making it difficult to ensure consistent image quality and size.

Option D is not the best approach because using the AzCopy tool with the Sync option would synchronize the entire contents of the "MediaStorage" container with the "ProcessedImages" container, including the original images, which contradicts the requirement of keeping the original images in the "MediaStorage" container. Additionally, this approach does not involve resizing the images.

Skill Mapping:

Skills: Azure Developer Certification (AZ-204)

Subskills: Develop Azure compute solutions

Competencies: Azure Compute Solutions, Azure Functions, Azure Blob Storage

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

9. You are developing a real-time chat application that is expected to handle a high volume of concurrent users. The application needs to be highly available and responsive while minimizing costs. Which Azure service should you use to host the chat application?

A) Deploy the application to an Azure Functions app and configure it to automatically scale based on the number of incoming requests.

B) Host the application on an Azure Virtual Machine and configure it to automatically scale when the CPU load is high.

C) Deploy the application to an Azure App Service using the Standard service tier and configure auto-scaling based on CPU utilization.

D) Use Azure Kubernetes Service (AKS) to deploy and manage containers for the chat application, allowing for automatic scaling based on resource usage.

You are engineering a real-time group discourse application anticipating extensive simultaneous users. Striving for heightened reachability and efficacy whilst curtailing expenditure, which Azure facility should you engage to hoist the conversational program?

A) Mount the communique construct onto an Azure Features application and ordain it to mechanically magnify contingent on the numerosity of arriving petitions.

B) Harbor the dialogue edifice on an Azure Digital Machine and prepare it to automatically expand when the processor burden is lofty.

C) Hoist the interaction prototype on an Azure App Facility exploitation the Commonplace assistance stratum and decree autoscale founded on CPU employment.

D) Engender recourse to Azure Kubernetes Utility (AKS) to instigate and administer recipients representing the colloquy construction, enabling automatic expansion contingent on resource consumption.

Answer: D

Feedback(if correct):-

Utilizing Azure Kubernetes Utility (AKS) delivers a fleet-footed atmosphere, compatible with microservices, like real-time dialog applications, and elastic scaling capacities. These characteristics fulfill the conditions for broad availability, agile conduct, and expense mitigation requisites. Notably, riveting attention to the remaining three options reveals flaws that substantiate the preference for AKS.

Feedback(if wrong):-

Option A falters by proposing Azure Features, which predominantly suits ephemeral, background processes. Such a foundation jeopardizes the continual contact demanded by a real-time chat application.

Option B missteps by advocating Azure Virtual Machines. Even though vertical scaling can augment system limits, it ultimately culminates in diminishing returns. More importantly, it neglected to address the inherent hurdles of provisioning and managing virtual machines, consequently escalating labor costs.

Option C erred by electing the Standard service coating on Azure App Services. Whilst this choice supplies fundamental autoscaling attributes, it encounters difficulties in catering to the fluctuating resource demands synonymous with real-time chat applications. This constraint emerges from the limited extensibility characteristic of conventional app services, inhibiting the achievement of outstanding performance and affordability.

Skill Mapping:

Skills: Azure Developer Certification (AZ-204)

Subskills: Develop Azure compute solutions

Competencies: Azure Compute Solutions, Virtual Machines, Azure App Service, Azure Functions, Azure Kubernetes Service (AKS)

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

10. You are tasked with developing a web application for a startup company that anticipates rapid growth in user traffic. The application must maintain high availability and responsiveness while keeping costs in check. Which of the following deployment options should you select to meet the requirements?

A) Host the application on a shared hosting plan.

B) Configure the hosting environment to manually scale based on predicted traffic patterns.

C) Utilize an Azure App Service with a Standard service tier.

D) Enable automatic scaling for the App Service plan to handle increased traffic.

Answer: C, D

Feedback(if correct): Options C and D are correct because using an Azure App Service with a Standard service tier provides a reliable platform with better performance while enabling automatic scaling ensures the application can dynamically adjust to handle increased traffic, meeting the requirements for high availability, responsiveness, and cost-effectiveness.

Feedback(if wrong): Option A is incorrect as shared hosting plans may not offer the scalability and performance needed for rapid growth in user traffic. Option B is also incorrect because manually scaling based on predicted traffic patterns may not be efficient or practical for handling unpredictable increases in traffic.

Skill Mapping:

Skills: Azure Developer Certification (AZ-204)

Subskills: Develop Azure compute solutions

Competencies: Choose the appropriate Azure services to meet scalability, availability, and cost requirements for web applications

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

11. You are tasked with managing API policies for Contoso, Ltd.'s web service, which handles various incoming requests from clients. The API policies are crucial for ensuring proper request processing and management. As part of your responsibilities, you are reviewing and configuring an API Policy object that includes XML markup to manage request processing based on specific conditions. You are configuring an API Policy object for Contoso, Ltd.'s web service, which utilizes Azure API Management to process incoming requests. The policy includes XML markup to control request processing based on predefined conditions. Review the provided XML markup and statements to assess their accuracy.

Analyze the fragment below:

```
<policies>

<inbound>

...

<set-variable name="payloadWeight" value="@(context.Request.Headers["Content-Length"] [0])"/>

<choose>

<when condition="@(int.Parse(context.Variables.GetValueOrDefault<string>("payloadWeight")) > 512000)">

<rewrite-uri template="/largePayloadUri"/>

<set-backend-service base-url="https://fabrikam.com/api/v10/" />

</when>

<otherwise>

<rewrite-uri template="/smallPayloadUri"/>

<set-backend-service base-url="https://fabrikam.com/api/v9/" />

</otherwise>

</choose>

...

</inbound>

...

</policies>
```

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

Statements:

1. Placing the XML segment in the <inbound> section of the policy ensures it handles incoming requests effectively.

2. If the size of the request body exceeds 256 KB, an error will be triggered.

3. The policy will prioritize the higher version of the API if the incoming request is directed to http://contoso.com/api/9.2/.

A. No, Yes, Yes
B. Yes, No, No
C. Yes, Yes, Yes
D. No, No, No

Answer: B

Feedback(if correct):-

1. Statement 1 (Placing the XML segment in the <inbound> section of the policy ensures it handles incoming requests effectively): This statement is correct because the <inbound> section of the API policy is where you define policies that execute before the request is forwarded to the backend service. Placing the XML segment here ensures it handles incoming requests effectively.

2. Statement 2 (If the size of the request body exceeds 256 KB, an error will be triggered): This statement is incorrect. The condition in the policy checks if the request body size exceeds 512 KB, not 256 KB. Therefore, the correct answer is No.

3. Statement 3 (The policy will prioritize the higher version of the API if the incoming request is directed to http://contoso.com/api/9.2/): This statement is incorrect. The policy does not prioritize the API version based on the request URL. Instead, it redirects requests with large payload sizes to a different backend service version. Therefore, the correct answer is No.

Feedback(if wrong):-

Option A is incorrect because placing the XML segment in the <inbound> section of the policy does not ensure effective handling of incoming requests. While the <inbound> section is indeed used to process incoming requests, the effectiveness of handling depends on the specific logic defined within the XML segment, not just its placement within the policy.

Option C is incorrect because the condition in the policy checks whether the request body size exceeds 512 KB, not 256 KB. Therefore, selecting "Yes" for this statement implies a misunderstanding of the condition defined in the policy.

Option D is incorrect because the policy does not prioritize API versions based on the incoming request URL. Instead, it redirects requests with large payload sizes to a different backend service version, as determined by the condition in the policy. Therefore, selecting "Yes" for this statement misinterprets the behavior of the policy.

Skill Mapping:

Skills: Azure Developer Certification (AZ-204)

Subskills: Develop Azure compute solutions

Competencies: Azure API Management, Azure Functions, Azure Logic Apps, Azure Kubernetes Service (AKS)

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

12. You are tasked with configuring Azure Key Vault to securely store sensitive information for an application. Additionally, you need to ensure that a Linux virtual machine (VM) hosted on Azure is encrypted at rest using industry-standard encryption technology. To accomplish this, you plan to use Azure Disk Encryption. Based on the provided scenario, select the correct sequence of PowerShell commands to set up Azure Key Vault and enable disk encryption for the Linux VM. Choose the options that best align with Azure's best practices and ensure the secure storage and encryption of sensitive information.

A) Create the VM, create a Key Vault, create a cryptographic key within the Key Vault, and enable disk encryption for the VM.

B) Create a Key Vault, create a cryptographic key within the Key Vault, create the VM, and enable disk encryption for the VM.

C) Create a cryptographic key within the Key Vault, create the VM, enable disk encryption for the VM, and create a Key Vault.

D) Enable disk encryption for the VM, create a Key Vault, create a cryptographic key within the Key Vault, and create the VM.

Answer: B

Feedback(if correct):-

The correct answer is to first create the Azure Key Vault and enable it for disk encryption, then create a cryptographic key within the Key Vault, create the virtual machine, and finally enable encryption for the VM using the Key Vault and cryptographic key. This sequence ensures that the necessary components for encryption are properly set up before enabling disk encryption on the VM.

Feedback(if wrong):-

- Option A: This option attempts to create the VM before setting up the Key Vault and encryption key, which violates the correct sequence of actions required for Azure Disk Encryption.

- Option B: While not entirely wrong, this option creates the key after enabling encryption with `az vm encryption enable`, which is inefficient. The key should be created beforehand to ensure it's available for the encryption process.

- Option C: Similar to Option A, this option creates the VM before configuring the Key Vault and encryption key, which is incorrect.

- Option D: This option also creates the VM before setting up the Key Vault and encryption key, which is not the recommended sequence of actions for Azure Disk Encryption.

Skill Mapping:

Skills: Azure Developer Certification (AZ-204)

Subskill: Develop Azure compute solutions

Competencies: Azure Compute Solutions

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

13. You are developing a web application that utilizes Azure App Service and Azure SQL Database. The application must securely access sensitive configuration data stored in Azure App Configuration. As part of your development process in Visual Studio Code, you need to take specific actions to enable the web application to retrieve this configuration data.

Instructions:

Which two actions should you perform to allow your web application to securely access the configuration data?

A. Configure Azure App Configuration settings in the application code to retrieve the configuration data.

B. Implement Azure Key Vault to store and manage the sensitive configuration data securely.

C. Integrate the Azure App Configuration SDK into your project's dependencies for seamless access to the configuration data.

D. Set up Azure Active Directory authentication for the web application to ensure secure access to Azure App Configuration.

Skill Mapping:

Skills: Azure Developer Certification (AZ-204)

Subskills: Develop Azure compute solutions

Competencies: Azure API Management, Azure Functions, Azure Logic Apps, Azure Kubernetes Service (AKS)

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

14. You are developing an application that requires seamless integration with Azure Cosmos DB, a globally distributed NoSQL database, using the .NET API. Your task is to create an object that configures and executes requests to the database.

You are tasked with configuring a .NET object to interact with Azure Cosmos DB, a globally distributed NoSQL database. Which code segment should you use?

A) `new Container(EndpointUri, PrimaryKey);`

B) `new Database(Endpoint, PrimaryKey);`

C) `new CosmosClient(EndpointUri, PrimaryKey);`

D) `new DocumentClient(EndpointUri, PrimaryKey);`

Answer: C

Feedback(if correct):-

The correct option creates a CosmosClient object, which is the appropriate class for interacting with Azure Cosmos DB using the .NET API. The CosmosClient class allows configuration and execution of requests to the database, making it the suitable choice for this scenario.

Feedback(if wrong):-

A) `new Container(EndpointUri, PrimaryKey);`: This option is incorrect because Container is not a class used for interacting with Azure Cosmos DB using the .NET API. Containers are used to represent collections within a Cosmos DB database.

B) `new Database(Endpoint, PrimaryKey);`: This option is incorrect because Database is not a class used for interacting with Azure Cosmos DB using the .NET API. Databases represent the logical containers for collections within a Cosmos DB account.

D) `new DocumentClient(EndpointUri, PrimaryKey);`: This option is incorrect because DocumentClient is not used in the latest versions of the .NET SDK for Azure Cosmos DB. Instead, CosmosClient is the preferred class for interacting with the database.

Skill Mapping:

Skills: Azure Developer Certification (AZ-204)

Subskills: Develop Azure compute solutions

Competencies: Choose the appropriate Azure services to meet scalability, availability, and cost requirements for web applications.

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

15. Within the sphere of Azure development, how would you go about connecting to a globally spanned No-SQL database utilizing the .NET API? Illustrate the process by composing the requisite lines of code to initialize the connection.

A) await databaseContainer.CreateItemAsync(item, new PartitionKey("myPartition"));

B) cosmosDbClient = new CosmosClient("<connection_string>", "<database_name>");

C) var container = new Container("<database_name>", "<container_name>");

D) db.CreateDocumentAsync(documentLink, document);

Answer: B

Feedback(if correct):-

B) cosmosDbClient = new CosmosClient("<connection_string>", "<database_name>");

To connect to a globally distributed NoSQL database like Azure Cosmos DB using the .NET API, you typically initialize a CosmosClient object with the connection string and database name. The CosmosClient class is designed to provide functionalities for interacting with Azure Cosmos DB databases. Therefore, option B represents the correct approach for establishing a connection to a Cosmos DB database using the .NET API.

Feedback(if wrong):-

A) await databaseContainer.CreateItemAsync(item, new PartitionKey("myPartition")):

- This line of code is used to create a new item within a container in Cosmos DB, but it does not represent the process of establishing a connection to the database.

C) var container = new Container("<database_name>", "<container_name>"):

- This line of code initializes a Container object within a Cosmos DB database, but it assumes that the connection to the database has already been established. It does not demonstrate the process of connecting to the database.

D) db.CreateDocumentAsync(documentLink, document):

- This line of code is part of the deprecated DocumentClient class, which was used in earlier versions of the Cosmos DB SDK. It is not the recommended approach for connecting to a Cosmos DB database using the .NET API.

Skill Mapping:

Skills: Azure Developer Certification (AZ-204)

Subskills: Develop Azure compute solutions

Competencies: Choose the appropriate Azure services to meet scalability, availability, and cost requirements for web applications.

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

16. Assume the role of a developer faced with the challenge of engaging a globally distributed No-SQL database through the .NET API within the Azure ecosystem. Deliberate on the ensuing course of action and articulate the essential lines of code needed to kickstart the process.

A) var container = new Container(endpointUri, primaryKey, databaseName, containerName);

B) var database = new Database(endpoint, primaryKey, databaseName);

C) var cosmosClient = new CosmosClient(endpointUri, primaryKey);

D) var collection = database.GetCollection(collectionLink);

Answer: C

Feedback(if correct):-

Correct answer: C) var cosmosClient = new CosmosClient(endpointUri, primaryKey);

Option C correctly initializes a CosmosClient object, which is the appropriate way to connect to a globally distributed No-SQL database like Azure Cosmos DB using the .NET API.

Feedback(if wrong):-

A) var container = new Container(endpointUri, primaryKey, databaseName, containerName);

This option attempts to create a Container object directly, but in Azure Cosmos DB, you first need to initialize a CosmosClient object before creating containers.

B) var database = new Database(endpoint, primaryKey, databaseName);

This option tries to create a Database object, but in Azure Cosmos DB, you work with containers, not databases, directly through the .NET API.

D) var collection = database.GetCollection(collectionLink);

This option is incorrect because it references a method GetCollection that does not exist in the Azure Cosmos DB .NET API. Additionally, it assumes a database-centric approach, whereas Azure Cosmos DB operates primarily with containers.

Skill Mapping:

Skills: Azure Developer Certification (AZ-204)

Subskills: Develop Azure compute solutions

Competencies: Choose the appropriate Azure services to meet scalability, availability, and cost requirements for web applications.

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

17. You are developing an Azure Cosmos DB solution for a real-time analytics application. The application requires storing data about user interactions, such as clicks, views, and purchases, in a Cosmos DB container named "UserInteractions." Each document in the container represents a single user interaction event.

You need to implement a query to retrieve the latest 50 user interaction events for a specific user, identified by their unique user ID. The query should efficiently return the desired data while minimizing resource consumption.

Which approach should you take to implement the query?

A. Use the SQL API and execute a SQL query with the WHERE clause filtering by the user ID, then use the ORDER BY clause to sort the results by timestamp in descending order, and finally use the TOP keyword to limit the results to 50.

B. Use the SQL API and execute a SQL query with the WHERE clause filtering by the user ID, then use the ORDER BY clause to sort the results by timestamp in ascending order, and finally use the TOP keyword to limit the results to 50.

C. Use the MongoDB API and execute a find operation with a filter on the user ID, then use the sort operation to order the results by timestamp in descending order, and finally use the limit operation to restrict the results to 50.

D. Use the Table API and execute a query with a filter on the user ID, then use the order by operation to sort the results by timestamp in descending order, and finally use the top operation to limit the results to 50.

Answer: A

Feedback(if correct):-

The correct approach is A. Use the SQL API and execute a SQL query with the WHERE clause filtering by the user ID, then use the ORDER BY clause to sort the results by timestamp in descending order, and finally use the TOP keyword to limit the results to 50.

breakdown of the approach:

- Using the SQL API allows you to execute SQL queries against the Cosmos DB container.

- The WHERE clause filters the results based on the user ID, ensuring that only events for the specific user are retrieved.

- The ORDER BY clause sorts the results by timestamp in descending order, ensuring that the latest events appear first.

- The TOP keyword limits the results to 50, efficiently retrieving only the desired number of events.

Feedback(if wrong):-

B. Use the SQL API and execute a SQL query with the WHERE clause filtering by the user ID, then use the ORDER BY clause to sort the results by timestamp in ascending order, and finally use the TOP keyword to limit the results to 50.

This option is incorrect because it sorts the results by timestamp in ascending order, which means the oldest user interaction events will appear first. However, the requirement is to retrieve the latest 50 user interaction events. Therefore, sorting in ascending order will not fulfill the requirement.

C. Use the MongoDB API and execute a find operation with a filter on the user ID, then use the sort operation to order the results by timestamp in descending order, and finally use the limit operation to restrict the results to 50.

This option is incorrect because it suggests using the MongoDB API instead of the SQL API, which is not the appropriate API for Azure Cosmos DB in this scenario. The SQL API should be used for executing queries against Azure Cosmos DB.

D. Use the Table API and execute a query with a filter on the user ID, then use the order by operation to sort the results by timestamp in descending order, and finally use the top operation to limit the results to 50.

This option is incorrect because it suggests using the Table API, which is not suitable for querying Azure Cosmos DB containers. The Table API is used for working with Azure Table storage, not Azure Cosmos DB.

Therefore, the correct approach is A. Use the SQL API and execute a SQL query with the WHERE clause filtering by the user ID, then use the ORDER BY clause to sort the results by timestamp in descending order, and finally use the TOP keyword to limit the results to 50.

Skill: Azure Developer Certification (AZ-204)

Subskill: Develop for Azure storage.

Competency: Cosmos DB, Querying data

Difficulty Level: Intermediate

Blooms Taxonomy Level: Application

18. You are developing a cloud-based application that requires secure and scalable storage for various types of data, including files, images, and documents. The application needs to ensure that access to the stored data is restricted to authorized users and that access activities are logged for auditing purposes. Which Azure service should you use to meet these requirements?

A) Azure Functions

B) Azure Key Vault

C) Azure Active Directory

D) Azure Blob Storage

Answer: D

Feedback(if correct):-

To meet the requirements of secure and scalable storage with access control and auditing, the most suitable Azure service is Azure Blob Storage. Azure Blob Storage provides secure and reliable storage for large amounts of unstructured data. It allows you to control access to the stored data through shared access signatures (SAS) and access policies, ensuring that only authorized users can access the data. Additionally, Azure Blob Storage offers logging and monitoring capabilities to track access activities for auditing purposes.

Azure Blob Storage is a cloud-based object storage solution that provides secure and scalable storage for unstructured data such as files, images, and documents. It offers various access tiers to optimize costs and provides access control mechanisms to restrict access to authorized users. Additionally, Blob Storage integrates with Azure Active Directory for authentication and authorization, and it provides auditing capabilities to track access activities for compliance and security purposes.

Feedback(if wrong):-

Azure Functions is a serverless compute service that allows you to run event-driven code in response to various triggers, and it is not suitable for storing data.

Azure Key Vault is a cloud-based service that provides secure storage of cryptographic keys and secrets, and it is not designed for storing unstructured data.

Azure Active Directory is a cloud-based identity and access management service that provides authentication and authorization capabilities, and it is not designed for storing data.

Skill: Azure Developer Certification (AZ-204)

Subskill: Develop for Azure storage.

Competency: Understanding different Azure storage services, configuring access control and logging for Azure storage, and Integrating Azure storage with other Azure services.

Difficulty Level: Intermediate

Blooms Taxonomy Level: Application

19. A travel and bookings management service plans to incorporate restaurant bookings into its platform. As a developer, you need to ensure that users can search for restaurants by name, description, location, and cuisine while also providing options to filter the results by location, cuisine, rating, and family-friendliness. Which attribute annotations should you use in the restaurant class to achieve these search and filtering functionalities?

A) IsFilterable, IsSortable, IsSearchable

B) IsSearchable, IsFilterable, IsFacetable

C) IsFilterable, IsFacetable, IsSortable

D) IsSearchable, IsFacetable, Required

Answer: B

Sure, here's the feedback for the question:

Feedback(if correct):

The correct answer is B) IsSearchable, IsFilterable, IsFacetable. These attribute annotations enable the restaurant class properties to be searchable by name, description, location, and cuisine, filterable by location, cuisine, rating, and family-friendliness and allow for faceted navigation to refine search results based on specific categories or facets.

Feedback(if wrong):

Option A) IsFilterable, IsSortable, IsSearchable: This option is incorrect because it includes IsSortable, which is not necessary for the described functionality of searching and filtering restaurants by various criteria.

Option C) IsFilterable, IsFacetable, IsSortable: This option is incorrect because it includes IsSortable, which is not necessary for the described functionality of searching and filtering restaurants by various criteria.

Option D) IsSearchable, IsFacetable, Required: This option is incorrect because it includes Required, which is not relevant to specifying the search and filtering functionality of the restaurant class. Additionally, it lacks the IsFilterable attribute, which is necessary for filtering search results.

Skill: Azure Developer Certification (AZ-204)

Subskill: Develop for Azure storage.

Competency: Querying data

Difficulty Level: Intermediate

Blooms Taxonomy Level: Application

20. Your development team is enhancing a travel and bookings management service to include restaurant bookings. To implement the search functionality for restaurants, you need to ensure that all words in restaurant descriptions are included in searches and provide filtering options based on various attributes such as location, cuisine, rating, and family-friendliness. Which property attributes should you apply to the description property in the restaurant class to meet these requirements?

A) IsSearchable, IsFilterable, IsSortable

B) IsSortable, IsFacetable, Required

C) IsFilterable, IsFacetable, IsSortable

D) IsSearchable, Required

Answer: A

Feedback(if correct):

The correct answer is A) IsSearchable, IsFilterable, IsSortable. These attributes are essential for implementing the search functionality for restaurants, allowing users to search by description, filter by various attributes, and sort the search results if needed.

Feedback(if wrong):

Options B, C, and D are incorrect.

- Option B: IsSortable and IsFacetable are not sufficient for implementing the search functionality and filtering options described in the question.

- Option C: IsFilterable and IsFacetable are not enough to ensure that all words in restaurant descriptions are included in searches.

- Option D: IsSearchable alone is not sufficient as it does not include the ability to filter and sort the search results.

Skill: Azure Developer Certification (AZ-204)

Subskill: Develop for Azure storage.

Competency: Cosmos DB, Querying data

Difficulty Level: Intermediate

Blooms Taxonomy Level: Application

21. As part of enhancing a travel and bookings management service to include restaurant bookings, you need to develop a solution that enables users to search for restaurants and filter the results by various criteria. Which Azure service should you incorporate into your solution to facilitate efficient search and filtering capabilities for restaurants?

A) Azure Search

B) Azure Cosmos DB

C) Azure Storage

D) Azure SQL Database

Answer: A

Feedback(if correct):

The correct answer is A) Azure Search. Azure Search is specifically designed to facilitate efficient search capabilities for various types of data, including text-based searches like restaurant descriptions, and supports filtering and faceting features, making it an ideal choice for implementing search and filtering capabilities for restaurants in a travel and bookings management service.

Feedback(if wrong):

Options B, C, and D are incorrect.

- Option B: Azure Cosmos DB is a NoSQL database service and is not specifically designed for efficient search and filtering capabilities like Azure Search.

- Option C: Azure Storage is a scalable storage solution for structured and unstructured data but does not provide built-in search capabilities like Azure Search.

- Option D: Azure SQL Database is a relational database service and while it supports querying and filtering data, it may not be as efficient for search functionality compared to Azure Search.

Skill: Azure Developer Certification (AZ-204)

Subskill: Develop for Azure storage.

Competency: Querying data

Difficulty Level: Intermediate

Blooms Taxonomy Level: Application

22. You need to move an IoT sensor dataset from an on-premises Cassandra database to Azure. What should you use?

A) Azure Stream Analytics

B) Azure Data Factory

C) Azure Databricks

D) Azure Time Series Insight

Answer: B

Feedback (if correct):

You've got it right! The Azure Database Migration Service is the go-to solution for migrating on-premises MongoDB data to Azure Cosmos DB while ensuring minimal downtime and reliable migration. Great pick!

Feedback (if wrong):

A) Azure AD: Wrong choice as Azure AD serves as an identity and access management platform, having nothing to do with data migration.

C) Azure Files Explorer: Being a file manager utility, Azure Files Explorer does not concern itself with data migration tasks.

D) Azure MS DB Data Migration tool: Another poor fit, this tool is used for migrating data from earlier Microsoft SQL Server or Oracle databases to later versions.

Skills: Azure Developer Certification (AZ 204)

Subskills: Implement Azure storage

Competency: Blob storage

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

23. You are developing a web application that allows users to upload images to an Azure Blob Storage account named "MediaStorage." The application consists of two main features: image uploading and image processing. After images are uploaded, they need to be automatically resized and moved to a separate container named "ProcessedImages." Additionally, the original images must remain in the "MediaStorage" container. What is the most suitable approach to automatically resize and move the uploaded images to the "ProcessedImages" container while keeping the original images in the "MediaStorage" container?

A) Utilize the Azure Storage REST API to resize the images on the client-side and directly upload them to the "ProcessedImages" container.

B) Implement a virtual machine to monitor the "MediaStorage" container and manually resize and move the images to the "ProcessedImages" container.

C) Use Azure Functions triggered by blob storage events to resize and move the images to the "ProcessedImages" container.

D) Utilize the AzCopy tool with the Sync option to continuously synchronize the contents of the "MediaStorage" container with the "ProcessedImages" container.

Answer: C

Feedback(if correct):-

Leveraging Azure Functions triggered by blob storage events offers several advantages in this scenario. First, it enables serverless computing, meaning that you pay solely for the compute time consumed during execution. Second, it integrates seamlessly with Azure Blob Storage, allowing you to process and manipulate stored objects easily. Lastly, using Azure Functions reduces manual intervention and potential errors compared to periodically checking and moving files on a VM.

Feedback(if wrong):-

B) Implementing a virtual machine to monitor the MediaStorage container and manually resizing and moving images isn't practical due to the hands-on involvement required. Over time, this method becomes cumbersome, prone to human mistakes, and less cost-effective.

A) Using the Azure Storage REST API to resize the images on the client side introduces additional complexity and dependency on client-side technologies. Furthermore, it compromises bandwidth efficiency since larger, non-resized images would still require transmission to the server before resizing.

D) Utilizing the AzCopy tool with the Sync option seems appealing initially, but it overlooks the fact that processed images need to be saved separately from original media files. Merely syncing containers wouldn't guarantee the distinct treatment of resized versus original pictures. Thus, this solution falls short of addressing the primary objective outlined in the scenario.

Skill Mapping:
Skills: Azure Developer Certification (AZ 204)

Subskill: Develop for Azure storage

Competency: Azure Storage

Specific Competency: Blob storage

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

24. You are tasked with configuring custom roles in Azure to meet specific requirements for your organization's migration to the cloud. The IT department needs to grant internal developers communication access with Microsoft support while restricting service agents from viewing resources and creating support tickets only. Which actions should you take to create the custom role?

A. Run the PowerShell command `Get-AzureRmRoleDefinition -Name '*Reader' | ConvertTo-Json | Out-File C:\SupportRole.json` to retrieve the existing Reader role definition.

B. Run the PowerShell command `Get-AzureRmRoleDefinition -Name '*Operator' | ConvertTo-Json | Out-File C:\SupportRole.json` to retrieve the existing Operator role definition.

C. Execute the PowerShell command `Set-AzureRmRoleDefinition -Name 'Reader' -InputFile C:\SupportRole.json` to update the Reader role with the required permissions.

D. Execute the PowerShell command `Set-AzureRmRoleDefinition -InputFile C:\SupportRole.json` to update the custom role with the specified permissions.

Answer: A

Feedback(if correct):-

A. Run the PowerShell command `Get-AzureRmRoleDefinition -Name '*Reader' | ConvertTo-Json | Out-File C:\SupportRole.json` to retrieve the existing Reader role definition.

This option retrieves the existing Reader role definition, which can serve as a starting point for creating the custom role. It ensures that the custom role inherits some of the necessary permissions.

Feedback(if wrong):-

B. Run the PowerShell command `Get-AzureRmRoleDefinition -Name '*Operator' | ConvertTo-Json | Out-File C:\SupportRole.json` to retrieve the existing Operator role definition.

- This option attempts to retrieve the existing Operator role definition, which might not be relevant to the requirements specified in the scenario. There's no indication that the Operator role is suitable for customizing to meet the organization's needs.

C. Execute the PowerShell command `Set-AzureRmRoleDefinition -Name 'Reader' -InputFile C:\SupportRole.json` to update the Reader role with the required permissions.

- This option tries to update the existing Reader role directly, which may not align with the requirement to create a custom role tailored specifically for the organization's needs. Modifying built-in roles is generally not recommended.

D. Execute the PowerShell command `Set-AzureRmRoleDefinition -InputFile C:\SupportRole.json` to update the custom role with the specified permissions.

- This option attempts to update a custom role with permissions specified in a JSON file. However, there's no indication that a custom role has already been created or defined. Creating a custom role typically involves defining permissions from scratch rather than updating an existing role.

Remember, granting access based on identities (internal developers, service agents) instead of roles is generally recommended for better security and control.

Skill Mapping:

Skill: Azure Developer Certification (AZ-204)

Subskill: Implement Azure security

Competency: Azure Security: Authentication and authorization, Azure Active Directory

Bloom's Taxonomy Level: Application

Difficulty Level: Intermediate

25. You are developing an ASP.NET Core API app that allows users to authenticate using Twitter and Azure Active Directory (Azure AD). Users must be authenticated before calling API methods, and the user's name must be logged for each method call. Which configurations should you use?

A) Attribute: AllowAnonymous, Request Header: X-MS-CLIENT-PRINCIPAL-ID

B) Attribute: Authorize, Request Header: X-MS-CLIENT-PRINCIPAL-NAME

C) Attribute: Authorize, Request Header: X-MS-CLIENT-PRINCIPAL-ID

D) Attribute: AllowAnonymous, Request Header: X-MS-CLIENT-PRINCIPAL-NAME

Answer: B

Feedback(if correct):-

Using the "Authorize" attribute ensures that users must be authenticated before calling API methods.

he request header "X-MS-CLIENT-PRINCIPAL-NAME" allows logging the user's name for each method call, which is a requirement mentioned in the question.

Feedback(if wrong):-

Option A is incorrect because allowing anonymous access with the "AllowAnonymous" attribute contradicts the requirement that users must be authenticated before calling API methods. Additionally, logging the "X-MS-CLIENT-PRINCIPAL-ID" header is not mentioned as a requirement for logging the user's name.

Option C is incorrect because using the "X-MS-CLIENT-PRINCIPAL-ID" header would log the user's ID, not their name, which does not meet the requirement specified in the question.

Option D is incorrect because allowing anonymous access with the "AllowAnonymous" attribute contradicts the requirement that users must be authenticated before calling API methods. Additionally, logging the "X-MS-CLIENT-PRINCIPAL-NAME" header is necessary for recording the user's name, as per the question.

Skill Mapping:

Skill: Azure Developer Certification (AZ-204)

Subskill: Implement Azure security

Competency: Azure Security: Authentication and authorization

Bloom's Taxonomy Level: Application

Difficulty Level: Intermediate

26. **You are tasked with enhancing the security of an HR intranet site that handles confidential staff information. The site uses Azure Active Directory (AAD) for authentication. To meet compliance standards, you need to enforce multi-factor authentication** which **requires protection for the HR intranet. Which Essential steps should you take? Choose the correct options.**

A. Set up Multi-Factor Authentication in Azure AD.

B. Establish Conditional Access Policies in Azure AD.

C. Designate Multi-Factor Authentication coverage for specific users or groups.

D Update Azure AD subscription tier to Premium edition.

Answer: A, B

Feedback(if correct):-

Option A (Set up Multi-Factor Authentication in Azure AD) is essential as it sets up multi-factor authentication, while option B (Establish Conditional Access Policies in Azure AD to enforce multi-factor authentication specifically for the HR intranet) ensures that multi-factor authentication is specifically enforced for the HR intranet, aligning with the security requirements.

Feedback(if wrong):-

C. Designate Multi-Factor Authentication coverage for specific users or groups: While designating multi-factor authentication for specific users or groups may be necessary in some scenarios, it does not specifically address the requirement to enforce multi-factor authentication for the HR intranet.

D. Update Azure AD subscription tier to Premium edition: Updating the Azure AD subscription tier to Premium edition may offer additional features, but it is not specifically required to enforce multi-factor authentication for the HR intranet.

Skill Mapping:

Skill: Azure Developer Certification (AZ-204)

Subskill: Implement Azure security

Competency: Azure Security: Authentication and authorization, Azure Active Directory

Bloom's Taxonomy Level: Application

Difficulty Level: Intermediate

27. You are developing an ASP.NET Core API application that integrates with Azure Active Directory (Azure AD) and Twitter for authentication. Authentication is required before users can access API methods, and their names must be logged for auditing purposes. Which configuration options should you choose to achieve this?

A) Marking the API methods with the "Authorize" attribute and accessing the user's name from the "X-MS-CLIENT-PRINCIPAL-NAME" request header.

B) Enabling the "AllowAnonymous" attribute and retrieving the user's name from the "Proxy-Authorization" request header.

C) Adding the "AutoValidateAntiforgeryToken" attribute and extracting the user's name from the "X-Forwarded-For" request header.

D) Configuring the API methods with the "Authorize" attribute and obtaining the user's name from the "X-MS-CLIENT-PRINCIPAL-ID" request header.

Answer: A

Feedback(if correct):

The correct answer is option A. The "Authorize" attribute ensures that users must be authenticated before accessing API methods, while the "X-MS-CLIENT-PRINCIPAL-NAME" request header allows logging of the user's name for auditing purposes.

Feedback(if wrong):

Option B is incorrect because the "AllowAnonymous" attribute would allow unauthorized access to API methods, contrary to the requirement of user authentication.

Option C is incorrect because the "AutoValidateAntiforgeryToken" attribute is used for preventing cross-site request forgery (CSRF) attacks and is not related to user authentication or logging user names.

Option D is incorrect because the "X-MS-CLIENT-PRINCIPAL-ID" request header does not provide the user's name and is not suitable for logging purposes.

Skill Mapping:

Skill: Azure Developer Certification (AZ-204)

Subskill: Implement Azure security

Competency: Authentication and authorization, Role-based access control (RBAC), Azure Key Vault, Azure Active Directory

Bloom's Taxonomy Level: Application

Difficulty Level: Intermediate

28. You are developing an internal web application to allow employees to access sensitive data securely. The application is authenticated using Azure Active Directory (AAD). To enhance security, you need to implement multi-factor authentication (MFA) for the application. What steps should you take to achieve this?

A) Configure Azure AD to enforce multi-factor authentication for the web application.

B) Implement Azure AD Application Proxy for secure remote access to the web application.

C) Integrate the web application with Azure AD B2C for multi-factor authentication support.

D) Upgrade the Azure AD subscription to include premium features for MFA capabilities.

Answer: A

Feedback(if correct): Explanation: Option A is correct because configuring Azure AD to enforce multi-factor authentication is the appropriate solution for implementing MFA for the web application.

Feedback(if wrong): Option B is incorrect because Azure AD Application Proxy is used for secure remote access to on-premises applications, not for implementing MFA. Option C is incorrect because Azure AD B2C is typically used for customer-facing applications, not internal employee applications. Option D is incorrect because upgrading to Azure AD Premium is not specifically required for implementing MFA; it provides additional features beyond MFA.

Skill Mapping:

Skills: Azure Developer Certification (AZ-204)

Subskills: Implement Azure security

Competencies: Authentication and authorization, Role-based access control (RBAC), Azure Active Directory, Conditional Access Policies

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

29. Your Finance department utilizes a cloud-based workflow automation system powered by Azure Logic Apps. The system handles highly sensitive financial data exchanged among multiple departments, third-party APIs, and an on-premises database. You need to ensure secure connectivity and data transmissions for the entire pipeline. Which solution should you employ?

A) Create an Azure App Service Environment (ASE) hosting the Azure Logic Apps.

B) Utilize Azure AD B2B integration for secure connections to external APIs.

C) Opt for Integration Service Environments (ISE) to isolate and manage secure connectivity.

D) Implement VNet service endpoints to establish private connections to Azure resources.

Answer: C

Feedback(if correct):-

C) Opt for Integration Service Environments (ISE) to isolate and manage secure connectivities.

Integration Service Environments (ISE) provide a dedicated and isolated environment for running Logic Apps and other integration components. This ensures secure connectivity and data transmissions for sensitive workflows, as it allows you to connect to resources within a virtual network and utilize dedicated storage resources with fixed costs.

When dealing with sensitive financial data transmitted through a cloud-based workflow system, it's crucial to ensure secure communications and connections. In this scenario, Integration Service Environments (ISE) come into play as the ideal choice. ISEs enable you to run logic apps and create integration accounts with exclusive access to Azure virtual network resources, ensuring secure data exchange among involved parties. Other options aren't as fitting for this scenario, highlighting the significance of ISEs for secure data processing pipelines.

Feedback(if wrong):-

A) Create an Azure App Service Environment (ASE) hosting the Azure Logic Apps.

This option is incorrect because an Azure App Service Environment (ASE) is primarily used for hosting web apps, not Logic Apps. While it provides an isolated and dedicated environment, it may not be the most suitable solution for securing Logic Apps and handling sensitive financial data.

B) Utilize Azure AD B2B integration for secure connections to external APIs.

This option is incorrect because Azure AD B2B integration is primarily used for allowing external users to access resources within your organization's Azure AD tenant. It is not directly related to securing data transmissions within a workflow automation system like Azure Logic Apps.

D) Implement VNet service endpoints to establish private connections to Azure resources.

This option is incorrect because VNet service endpoints are used to secure Azure services by allowing them to be accessed privately from within a virtual network. While it enhances security, it may not provide the necessary isolation and management capabilities required for securing the entire workflow automation system powered by Azure Logic Apps.

Skill Mapping:

Skills: Azure Developer Certification (AZ-204)

Subskills: Implement Azure security

Competencies: Authentication and authorization, Role-based access control (RBAC), Azure Active Directory, Conditional Access Policies

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

30. Your e-commerce company wants to upgrade its existing on-premises order tracking system to Azure. The system houses critical order details in a MongoDB JSON document database, requiring seamless migration with minimal disruption. Moreover, the budget constraint mandates exploring cost-effective approaches. Which Azure solution should you embrace to tackle these challenges?

A) Azure Active Directory (Azure AD)

B) Azure Files Explorer

C) Azure Database Migration Service

D) Azure MS DB Data Migration Tool

Answer: C

Feedback (if correct):

You should use Azure Database Migration Service for migrating on-premises shipping data to Azure Cosmos DB with minimal downtime and high reliability. This service is specially designed for migrating MongoDB implementations to Azure Cosmos DB, making it the perfect fit for this scenario.

Feedback (if wrong):

A) Azure AD: Although Azure Active Directory plays a vital role in Azure identity and access management, it is not designed for data migration from on-premises systems to Azure.

B) Azure Files Explorer: While Azure Files Explorer helps browse Azure File Share contents, it cannot be employed for migrating data from on-premises MongoDB to Azure Cosmos DB.

D) Azure MS DB Data Migration Tool: This tool is primarily used for migrating data from various versions of Microsoft SQL Server or Oracle databases to newer ones. Thus, it is not suitable for migrating MongoDB data to Azure Cosmos DB.

Skills: Azure Developer Certification (AZ 204)

Subskill: Implement Azure security

Competency: Azure Security, Authentication and authorization, Role-based access control (RBAC), Azure Key Vault, Azure Active Directory

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

31. When deploying a global event management platform on Azure, which service offers advanced layer 7 load balancing functionalities like cookie-based session affinity and redirection within the same region?

A) Azure Virtual Machine Scale Set

B) Azure Front Door

C) Azure CDN

D) Azure Application Gateway

Answer: D

Feedback(if correct):-

Azure Application Gateway provides advanced layer 7 load balancing functionalities, including cookie-based session affinity and URL-based routing, making it suitable for managing traffic within the same region with features like redirection and session persistence.

Feedback(if wrong):- A) Azure Virtual Machine Scale Set is a service for deploying and managing a set of identical VMs, primarily used for scalability and high availability but doesn't offer layer 7 load balancing features like session affinity and redirection.

B) Azure Front Door is a global CDN service that provides content acceleration and security features but doesn't focus on layer 7 load balancing functionalities like session affinity and redirection within a region.

C) Azure CDN (Content Delivery Network) is designed for caching and delivering content to users globally with low latency, but it doesn't offer advanced layer 7 load balancing features like session affinity and redirection within the same region.

Skill Mapping:

Skills: Azure Developer Certification (AZ 204)

Subskill: Implement Azure security

Competency: Azure Security, Authentication, and authorization.

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

32. A company is migrating its on-premises infrastructure to Azure and plans to use the Azure SQL Database to store sensitive customer data for an application. The company wants to ensure that only specific users have access to view the sensitive information. Which two actions should you take to implement this security measure? Select the correct answers.

A) Include the managers' group.

B) Exclude the managers' group.

C) Exclude the administrators' group.

D) Utilize Azure Key Vault to encrypt and manage the sensitive data.

Answer: B, D

Feedback(if correct):-

B) Excluding the managers' group ensures that only users who are not part of this group will have access to view the sensitive information, thereby restricting access to specific users.

D) Utilizing Azure Key Vault to encrypt and manage sensitive data adds an extra layer of security by safeguarding the data at rest and in transit, thus ensuring its confidentiality and integrity.

Feedback(if wrong):-

A) Including the managers' group would grant access to sensitive information to members of this group, which contradicts the requirement to restrict access to specific users.

C) Excluding the administrators' group might hinder necessary administrative access to the Azure SQL Database, but it does not address the specific requirement of restricting access to sensitive information to specific users.

Skill Mapping:

Skills: Azure Developer Certification (AZ 204)

Subskill: Implement Azure security

Competency: Azure Security, Authentication, and authorization.

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

33. You're developing a mission-critical Azure application that stores highly sensitive customer data in Azure SQL Database. The company prioritizes tight control over who can access this data. Your task is to implement security measures that enforce this requirement.

Which two of the following security measures would most effectively ensure strict access control and protect sensitive customer data in the Azure SQL Database? Select the two best options.

A) Implement Azure Active Directory (AD) integration with Azure SQL Database and assign granular access roles to users based on their job functions and data sensitivity levels.

B) Create individual Azure SQL Logins with complex passwords for each user who needs access, and periodically rotate those passwords.

C) Configure Azure SQL Database firewall rules to restrict access to only specific IP addresses used by authorized applications.

D) Utilize Azure Key Vault to manage and rotate encryption keys for data at rest and in transit, and configure Azure SQL Database Transparent Data Encryption (TDE).

Answer: A, D

Feedback(if correct):-

A) Implement Azure Active Directory (AD) integration with Azure SQL Database and assign granular access roles to users based on their job functions and data sensitivity levels - Integrating Azure AD with Azure SQL Database allows for centralized authentication and authorization management, enabling granular control over user access based on roles and data sensitivity levels.

D) Utilize Azure Key Vault to manage and rotate encryption keys for data at rest and in transit, and configure Azure SQL Database Transparent Data Encryption (TDE) - Azure Key Vault ensures secure storage and management of encryption keys, while TDE encrypts sensitive data at rest, providing an additional layer of protection.

Feedback(if wrong):-

B) Creating individual Azure SQL Logins with complex passwords for each user and periodically rotating those passwords introduces management complexity and may not provide the granular access control required for the scenario.

C) Configuring Azure SQL Database firewall rules to restrict access based on specific IP addresses limits access to only authorized applications, but it does not address the need for granular access control based on user roles and data sensitivity levels.

Skill Mapping:

Skills: Azure Developer Certification (AZ 204)

Subskill: Implement Azure security

Competency: Azure Security, Authentication, and authorization.

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

34. You are developing a cloud-native application on Azure that requires background processing tasks to be executed regularly. Additionally, certain tasks need to be executed immediately upon the occurrence of specific events. You also need the ability to debug these background tasks remotely for troubleshooting purposes. You are tasked with recommending the appropriate WebJob type for different scenarios in your Azure application development. Which WebJob type should you recommend for each scenario?

Scenario 3:

You require the ability to perform remote debugging for troubleshooting purposes while executing background tasks.

Select the appropriate WebJob type for Scenario 3 from the options provided.

A. Continuous
B. Triggered
C. Incessant
D. Activated

Answer: A

Feedback(if correct):-

Continuous WebJobs are suitable for executing background tasks on all instances that the web app runs on, and they support remote debugging. This makes them the appropriate choice for Scenario 3, where the ability to perform remote debugging while executing background tasks is required.

Feedback(if wrong):-

Option B) Triggered is not recommended because Triggered WebJobs are designed to execute tasks immediately upon the occurrence of specific events, rather than regularly or supporting remote debugging.

Option C) Incessant is not a valid WebJob type in Azure App Service.

Option D) Activated is not a valid WebJob type in Azure App Service.

Skill Mapping:

Skills: Azure Developer Certification (AZ-204)

Subskills: Monitor, optimize, and troubleshoot Azure solutions

Competencies: Optimizing Azure Solutions, Troubleshooting Azure Solutions

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

35. You are developing a mobile application and need to implement Application Insights instrumentation capabilities using the Azure Mobile Apps SDK to analyze user interactions effectively. Which three data values should you capture to enable the Usage Analytics feature of Application Insights? Choose the correct options from the following:

A. Trace

B. Session Id

C. User Id

D. Events

Answer: A, C, D

Feedback(if correct):

To implement the Usage Analytics feature of Application Insights, you need to capture traces (A), user IDs (C), and events (D) to analyze user interactions and application usage effectively.

Feedback(if wrong):

Option B is incorrect. Application Insights automatically manages the Session Id, so capturing it manually is not necessary for implementing the Usage Analytics feature. Instead, capturing traces, user IDs, and events provide the necessary data for usage analysis.

Skill Mapping:

Skills: Azure Developer Certification (AZ-204)

Subskills: Monitor, optimize, and troubleshoot Azure solutions

Competencies: Implementing and configuring monitoring solutions

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

36. You are tasked with optimizing and monitoring an Azure solution to ensure its efficient performance. Discuss the steps you would take to implement and configure monitoring solutions for an Azure application, highlighting the key metrics and tools that can be utilized for effective monitoring.

A) Implement Azure Monitor to collect and analyze telemetry data, utilizing metrics such as CPU usage, memory utilization, and request rates.

B) Configure Azure Security Center to monitor the application's security posture and identify potential vulnerabilities.

C) Utilize Azure Advisor to provide recommendations for optimizing the application's performance and cost-efficiency.

D) Implement Azure Application Insights to gain deep insights into the application's performance, usage patterns, and exceptions.

Answer: A

Feedback(if correct):- Option A is correct: Implement Azure Monitor to collect and analyze telemetry data, utilizing metrics such as CPU usage, memory utilization, and request rates.

Azure Monitor is a comprehensive monitoring solution that allows you to collect and analyze telemetry data from Azure resources, applications, and infrastructure.

-ey metrics such as CPU usage, memory utilization, and request rates are essential for monitoring the performance and health of Azure applications.

Azure Monitor provides insights into resource utilization, application performance, and availability, enabling efficient optimization and troubleshooting.

Feedback(if wrong):-

Option B focuses on security monitoring using Azure Security Center, which is essential but not directly related to optimizing and monitoring application performance.

Option C mentions Azure Advisor, which provides recommendations for optimizing costs and performance, but it does not offer detailed telemetry data or performance metrics.

Option D suggests using Azure Application Insights, which is a powerful tool for monitoring application performance and usage patterns, but it does not cover broader Azure resource monitoring or provide metrics such as CPU and memory utilization.

Skill Mapping:

Skills: Azure Developer Certification (AZ-204)

Subskills: Monitor, optimize, and troubleshoot Azure solutions

Competencies: Implementing and configuring monitoring solutions

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

37. You are tasked with optimizing and monitoring an Azure solution to ensure its efficient performance. The application you are working on is a real-time messaging service that enables users to exchange messages securely. You need to implement monitoring solutions for the Azure application. Which of the following steps should you take, and what metrics and tools should you utilize for effective monitoring?

A) Configure Azure Security Center to monitor the application's security posture and identify potential vulnerabilities.

B) Implement Azure Monitor to collect and analyze telemetry data, utilizing metrics such as message delivery latency, message throughput, and error rates.

C) Utilize Azure Advisor to provide recommendations for optimizing the application's performance and cost-efficiency.

D) Implement Azure Application Insights to gain deep insights into the application's performance, usage patterns, and exceptions.

Answer: B

Feedback(if correct):-

B) Implement Azure Monitor to collect and analyze telemetry data, utilizing metrics such as message delivery latency, message throughput, and error rates.

Effective monitoring entails measuring key performance metrics, and Azure Monitor specializes in collecting and analyzing telemetry data. Metrics such as message delivery latency, message throughput, and error rates aid in identifying bottlenecks, slowdowns, and other performance concerns. Tools such as Application Insights and Security Center serve specific niches, while Azure Advisor targets recommendations for improvement. Hence, Azure Monitor proves to be the most versatile and important choice for monitoring the Azure application's performance.

Feedback(if wrong):

A) Configure Azure Security Center to monitor the application's security posture and identify potential vulnerabilities.

While monitoring security is undoubtedly important, especially in a real-time messaging service, this answer choice does not cover the broader spectrum of monitoring required for efficient performance, namely monitoring the application's performance and usage patterns.

C) Utilize Azure Advisor to provide recommendations for optimizing the application's performance and cost-efficiency.

Similar to the previous explanation, although receiving recommendations for performance optimization and cost reduction holds merits, it fails to fully address the holistic monitoring scope covering the application's performance and usage patterns. Besides, Azure Advisor serves as a consultative tool, while the immediate priority lies in acquiring concrete performance metrics and gaining deeper insights into the application.

D) Implement Azure Application Insights to gain deep insights into the application's performance, usage patterns, and exceptions.

Though Azure Application Insights certainly excels in delivering invaluable insights into the application's performance, usage patterns, and exceptions, the answer neglects to mention the importance of telemetry data analysis, which plays a significant role in monitoring and diagnosing performance issues. Without highlighting the significance of gathering and scrutinizing telemetry data, the answer loses relevance as a standalone solution for monitoring the Azure application.

Skill mapping :

Skills: Azure Developer Certification (AZ-204)

Subskills: Monitor, optimize, and troubleshoot Azure solutions

Competencies: Monitoring, Optimization, and Troubleshooting, Azure Monitor, Log Analytics, Application Insights, Troubleshooting common issues

Difficulty Level: Intermediate or Expert

Bloom's Taxonomy Level: Application

38. You are tasked with optimizing the scalability of a web application deployed on Azure Web App service. The current service plan is D1, and you need to ensure that the application infrastructure can automatically scale when the CPU load reaches 85 percent while minimizing costs. Which of the following steps would you implement to achieve the requirements?

A) Implement autoscaling for the web application.

B) Define a scale condition based on CPU load.

C) Migrate the web application to use the Premium App Service Plan.

D) Configure the web application to use the Free App Service Plan.

Answer: A, B

Feedback(if correct):-

A) Implement autoscaling for the web application.

B) Define a scale condition based on CPU load.

Enabling autoscaling (Option A) allows the infrastructure to automatically adjust resources based on demand, meeting scalability requirements efficiently. Defining a scale condition based on CPU load (Option B) ensures that the application scales up or down as needed to maintain optimal performance without manual intervention.

Feedback(if wrong):-

Option C is incorrect because migrating the web application to a higher-tier service plan, such as the Premium App Service Plan, may not be necessary to achieve autoscaling based on CPU load. Premium plans are typically associated with higher costs and may not align to minimize expenses while achieving scalability.

Option D is incorrect because configuring the web application to use the Free App Service Plan does not support autoscaling. Additionally, the Free plan offers limited resources, which would not meet the requirement of automatically scaling when CPU load reaches 85 percent.

Skill Mapping:

Skills: Azure Developer Certification (AZ-204)

Subskills: Monitor, optimize, and troubleshoot Azure solutions

Competencies: Monitoring, Optimization, and Troubleshooting, Azure Monitor, Log Analytics, Application Insights, Troubleshooting common issues

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

39. You are tasked with ensuring that the "KnowledhutApp" application can dynamically scale to meet demand. Which Azure Application Insights data model should you utilize to achieve this?

A) An Application Insights metric

B) An Application Insights dependency

C) An Application Insights trace

D) An Application Insights event

Answer: A

Feedback(if correct):

Option A is correct. Using Application Insights metrics allows you to monitor various performance metrics of the application, including resource utilization and request rates, which can be used to trigger autoscaling based on demand.

Feedback(if wrong):-

Option B: An Application Insights dependency tracks the dependencies of your application, such as external services or databases it relies on. While this data is useful for troubleshooting and understanding the application's dependencies, it is not directly related to scaling the application dynamically based on demand.

Option C: An Application Insights trace captures detailed information about individual requests or operations within the application. While tracing is essential for debugging and understanding the flow of requests, it does not provide the aggregated metrics necessary for autoscaling based on demand.

Option D: An Application Insights event records specific occurrences or custom events within the application. While events are valuable for tracking specific actions or milestones within the application, they do not provide the overall performance metrics needed for autoscaling based on demand.

Skill Mapping:

- Skill: Azure Developer Certification (AZ-204)

- Subskill: Monitor, optimize, and troubleshoot Azure solutions

- Competencies: Monitoring, Optimization, and Troubleshooting, Azure Monitor, Log Analytics, Application Insights, Troubleshooting common issues

- Difficulty Level: Intermediate

- Bloom's Taxonomy Level: Application

40. As an Azure developer, you are responsible for deploying a microservices solution on an Azure Kubernetes Service (AKS) cluster. Security best practices mandate that sensitive information such as API keys and database credentials should be managed separately from container images. You need to determine the Kubernetes element that should be utilized to simplify the management of this sensitive information.

Which Kubernetes element should you configure to ensure that newly deployed container instances in your AKS environment are ready to receive traffic before exposing them?

A) Utilize HPA to automatically scale based on CPU utilization.

B) Implement ConfigMaps for managing sensitive configurations.

C) Configure Liveness Probes for health monitoring.

D) Employ Readiness Probes for assessing container readiness.

Answer: D

Feedback(if correct):-

Utilizing Readiness Probes in Kubernetes allows you to assess container readiness, ensuring that containers are ready to receive traffic before being exposed. This ensures that sensitive information is managed separately from container images, aligning with security best practices in an AKS environment. This skillfully applies the knowledge of implementing security best practices within an AKS environment.

Feedback(if wrong):-

A) Utilize HPA to automatically scale based on CPU utilization. This option is incorrect because it pertains to automatic scaling based on CPU utilization and does not address the management of sensitive information separately from container images or the readiness of container instances.

B) Implement ConfigMaps for managing sensitive configurations. This option is incorrect because while ConfigMaps can be used for managing configurations, it does not specifically address the management of sensitive information separately from container images.

C) Configure Liveness Probes for health monitoring. This option is incorrect because Liveness Probes are used to determine if a container is running, not necessarily if it is ready to receive traffic, and it does not address the management of sensitive information separately from container images.

Skill Mapping:

Skills: Azure Developer Certification (AZ-204)

Subskills: Connect to and consume Azure services and third-party services

Competencies: Azure Services Integration, Azure Kubernetes Service (AKS), Ingress Controllers

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

41. As part of your role as an Azure developer, you are deploying a microservices solution on an Azure Kubernetes Service (AKS) cluster. Security best practices mandate that sensitive information such as API keys and database credentials should be managed separately from container images. Which Kubernetes element should you utilize to simplify the management of this sensitive information?

A) Ingress Controller

B) Liveness Probe

C) ConfigMap

D) HPA

Answer: C

Feedback(if correct):-

ConfigMap in Kubernetes is designed to store non-sensitive configuration data in key-value pairs. It is commonly used to separate sensitive information like API keys, database credentials, and configuration settings from container images, allowing for easier management and updating without redeploying containers.

Feedback(if wrong):-

A) Ingress Controller - An Ingress Controller is responsible for managing inbound traffic to Kubernetes clusters, not for storing sensitive information separately from container images.

B) Liveness Probe - Liveness Probe is used to determine if a container is healthy or not, but it does not handle the management of sensitive information.

D) HPA (Horizontal Pod Autoscaler) - HPA automatically scales the number of pods in a deployment based on observed CPU utilization, but it is not involved in managing sensitive information.

Skill Mapping:

Skills: Azure Developer Certification (AZ-204)

Subskills: Connect to and consume Azure services and third-party services

Competencies: Azure Services Integration, Azure Kubernetes Service (AKS), Ingress Controllers

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

42. As a DevOps engineer, you manage a complex microservices architecture hosted on an Azure Kubernetes Service (AKS) cluster. Recently, stakeholders requested implementing a single, accessible entry point for both internal teams and external clients. Consequently, you decide to leverage an Ingress Controller that handles reverse proxy functions, configurable traffic routing, and TLS termination. Question: Among the provided collection of command-line tools, identify the optimal instrument for setting up an Ingress Controller capable of managing a solitary, publicly exposed IP endpoint that directs traffic towards numerous microservices.

A) CoreDNS

B) KubeCtl

C) Brigade

D) Helm

Answer :D

Feedback(if correct):

Correct Answer (D) - Helm is the correct choice because it is a package manager for Kubernetes that simplifies the deployment and management of applications. It can be used to install and manage the nginx-ingress controller, which provides the functionality required for reverse proxy, traffic routing, and TLS termination as specified in the scenario.

Feedback(if wrong):

A) CoreDNS - CoreDNS is a DNS server that provides service discovery within Kubernetes clusters and is not directly related to setting up an Ingress Controller.

B) KubeCtl - Kubectl is a command-line tool used for interacting with Kubernetes clusters, but it is not primarily used for installing or managing Ingress Controllers.

C) Brigade - Brigade is a tool for scripting Kubernetes workflows and pipelines and is not typically used for setting up Ingress Controllers or managing networking aspects in Kubernetes clusters.

Skill Mapping:

Skills: Azure Developer Certification (AZ-204)

Subskills: Connect to and consume Azure services and third-party services

Competencies: Azure Services Integration, Azure Kubernetes Service (AKS), Ingress Controllers

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

43. You are developing an ASP.NET Core Web API web service that uses Azure Application Insights for performance monitoring and event tracking. You need to ensure that log messages can be correlated with events tracked by Application Insights. Which step should you take?

A) Configure the ApplicationServices to add Application Insights logging support in the Configure method of the Startup.cs file: `loggerFactory.AddApplicationInsights(app.ApplicationServices, LogLevel.Trace);`

B) Add the ApplicationInsightsLoggerOptions to enable logging options, such as including EventId and EventName properties, by using the following code in the ConfigureServices method: `services.AddOptions<ApplicationInsightsLoggerOptions>().AddMvc();`

C) Include the EventId and EventName properties in the ConfigureServices method by adding the following code: `services.AddOptions<ApplicationInsightsLoggerOptions>().Configure(o => o.IncludeEventId = true);`

D) Implement a custom logging middleware in the ASP.NET Core pipeline to intercept log messages and correlate them with Application Insights events.

Answer: A

Feedback(if correct):-

The correct step to ensure that log messages can be correlated with events tracked by Application Insights is:

A) Configure the ApplicationServices to add Application Insights logging support in the Configure method of the Startup.cs file: loggerFactory.AddApplicationInsights(app.ApplicationServices, LogLevel.Trace);

This step configures the ApplicationServices to add Application Insights logging support, enabling the correlation of log messages with events tracked by Application Insights. By adding this configuration in the Startup.cs file, the log messages will be seamlessly integrated with Application Insights for performance monitoring and event tracking.

Feedback(if wrong):-

B) Configure the ApplicationServices to add Application Insights logging support in the Configure method of the Startup.cs file: `loggerFactory.AddApplicationInsights(app.ApplicationServices, LogLevel.Trace);`

Option B is incorrect because the `AddApplicationInsights` method is not used to enable logging and ensure correlation with Application Insights events. This method is used to add Application Insights telemetry to the logging system, but it does not directly correlate log messages with events tracked by Application Insights.

C) Add the ApplicationInsightsLoggerOptions to enable logging options, such as including EventId and EventName properties, by using the following code in the ConfigureServices method: `services.AddOptions<ApplicationInsightsLoggerOptions>().AddMvc();`

Option C is incorrect because it does not enable logging and correlation with Application Insights events. The `AddOptions<ApplicationInsightsLoggerOptions>()` method is used to configure options for the Application Insights logger, but it does not directly enable logging or correlation with events tracked by Application Insights.

Option D is incorrect because while implementing a custom logging middleware could potentially intercept log messages, it is not the standard or recommended approach for correlating log messages with events tracked by Application Insights in an ASP.NET Core Web API. The use of a custom logging middleware is not necessary when Application Insights provides built-in support for logging and correlation.

Skill Mapping:

Skills: Azure Developer Certification (AZ-204)

Subskills: Connect to and consume Azure services and third-party services

Competencies: Azure Services Integration

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

44. You are setting up logging for an ASP.NET Core Web API that uses Azure Application Insights. Which step is necessary to ensure that log messages are correlated with events tracked by Application Insights?

A) Add the ApplicationInsightsLoggerOptions to enable logging options, such as including EventId and EventName properties, by using the following code in the ConfigureServices method: services.AddOptions<ApplicationInsightsLoggerOptions>().AddMvc();

B) Configure the ApplicationServices to add Application Insights logging support in the Configure method of the Startup.cs file: loggerFactory.AddApplicationInsights(app.ApplicationServices, LogLevel.Trace);

C) Implement a custom logging middleware in the ASP.NET Core pipeline to intercept log messages and correlate them with Application Insights events.

D) Include the EventId and EventName properties in the ConfigureServices method by adding the following code: services.AddOptions<ApplicationInsightsLoggerOptions>().Configure(o => o.IncludeEventId = true);

Answer: B

Feedback(if correct):-

The correct answer is B. Configuring the ApplicationServices in the Configure method of the Startup.cs file to add Application Insights logging support ensures that log messages are correlated with events tracked by Application Insights. This setup enables the ASP.NET Core Web API to properly integrate with Application Insights for comprehensive logging and monitoring.

Feedback(if wrong):-

A) Adding ApplicationInsightsLoggerOptions in the ConfigureServices method does not directly correlate log messages with Application Insights events. This configuration primarily deals with logging options, such as including specific properties in log messages.

C) Implementing a custom logging middleware in the ASP.NET Core pipeline could potentially intercept log messages, but it doesn't inherently correlate them with Application Insights events unless specifically programmed to do so. This option requires additional customization and is not a standard approach for integrating Application Insights logging.

D) Including EventId and EventName properties in the ConfigureServices method does not directly correlate log messages with Application Insights events. This configuration only specifies certain properties to include in the logging options but does not ensure proper correlation with Application Insights events.

Skill Mapping:

Skills: Azure Developer Certification (AZ-204)

Subskills: Connect to and consume Azure services and third-party services

Competencies: Azure Services Integration

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

45. You are developing a mobile delivery service application where drivers receive orders from restaurants in their area. You need to implement an Azure Service Bus solution, considering the dynamic nature of orders and drivers.

Question: With a focus on both scalability and efficiency in order delivery, which Service Bus configuration offers the best overall balance between manageability and flexibility?

Options:

A) Create a single topic for all restaurants and a subscription for each driver, allowing a subscription per restaurant in the future if needed.

B) Create a topic for each restaurant and a subscription for each driver, enabling precise order assignment.

C) Create a single topic for all restaurants and multiple subscriptions per driver, each tailored to specific restaurant groups.

D) Create a topic for each restaurant and a single Service Bus subscription for each restaurant for which a driver can receive orders.

Answer: B

Feedback(if correct): Option B is correct because it provides a balanced approach by creating a separate topic for each restaurant, ensuring efficient organization, and a subscription for each driver, enabling precise order assignment, which strikes a good balance between manageability and flexibility while addressing scalability and efficiency concerns.

Feedback(if wrong):

- Option A lacks flexibility because having a single topic for all restaurants may not efficiently handle scalability, and subscriptions for each driver might not cater to the dynamic nature of orders from different restaurants.

- Option C, although it offers multiple subscriptions per driver, which could be tailored to specific restaurant groups, may introduce complexity in managing and routing orders, potentially impacting efficiency.

- Option D introduces management overhead with a topic for each restaurant and doesn't provide precise order assignments for drivers, impacting efficiency and scalability in a dynamic environment.

Skill Mapping:

Skills: Azure Developer Certification (AZ-204)

Subskills: Connect to and consume Azure services and third-party services

Competencies: Azure Services Integration

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

46. You are developing a file-sharing platform expected to have fluctuating usage patterns throughout the day. The platform must remain responsive and available at all times while minimizing operational costs. What is the best approach to deploying the file-sharing platform on Azure?

A) Deploy the platform to an Azure Virtual Machine and configure it to scale manually based on predicted usage patterns.

B) Utilize Azure Blob Storage to store files and serve them to users, taking advantage of Azure's built-in scalability and high availability.

C) Deploy the platform to an Azure App Service using the Shared service tier, ensuring cost-effective hosting with minimal management overhead.

D) Implement Azure Cosmos DB to store file metadata and user information, allowing for automatic scaling and high availability of data storage.

Answer: A

Feedback(if correct):-

The best approach to deploy the file-sharing platform on Azure while ensuring responsiveness, availability, and cost-effectiveness is:

B) Utilize Azure Blob Storage to store files and serve them to users, taking advantage of Azure's built-in scalability and high availability.

Azure Blob Storage is designed for storing large amounts of unstructured data, such as files, and is highly scalable and reliable.

By leveraging Azure Blob Storage, you can take advantage of its built-in scalability and high availability, ensuring that the platform remains responsive and available at all times, regardless of fluctuating usage patterns.

Additionally, Azure Blob Storage offers cost-effective storage solutions, helping to minimize operational costs compared to hosting on virtual machines or using other services.

Feedback(if wrong):-

Option A may lead to manual management overhead and may not efficiently handle fluctuating usage patterns.

Option C, while cost-effective, may not offer the scalability and high availability required for a file-sharing platform with fluctuating usage patterns.

Option D focuses on data storage but may not be the best fit for serving files directly to users and may introduce unnecessary complexity and cost.

Skill Mapping:

Skills: Azure Developer Certification (AZ-204)

Subskills: Connect to and consume Azure services and third-party services

Competencies: Azure Services Integration

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

47. You are developing an e-commerce platform for a retail company, and you plan to integrate Azure Search for product search functionality. The platform should allow users to search for products based on various criteria and organize search results efficiently. You need to configure the SearchParameters class to meet the requirements. Which properties should you configure for the following scenarios?

Scenario 1:

Allow users to perform advanced searches using regular expressions.

A) QueryType

B) Property

C) OrderBy

D) SearchMode

Answer: A

Feedback(if correct):

The correct answer is option A) QueryType because configuring the QueryType property to "full" enables users to perform advanced searches using regular expressions, as it allows the use of the Lucene query syntax, supporting features like wildcard, fuzzy search, proximity search, and regular expressions.

Feedback(if wrong):

Property typically refers to specifying which fields to search within, OrderBy is used for sorting search results, and SearchMode defines the behavior of the search operation, none of which directly enable the use of regular expressions for advanced searches.

Skill Mapping:

Skills: Azure Developer Certification (AZ-204)

Subskills: Connect to and consume Azure services and third-party services

Competencies: Azure Services Integration

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

48. You are developing an e-commerce platform for a retail company, and you plan to integrate Azure Search for product search functionality. The platform should allow users to search for products based on various criteria and organize search results efficiently. You need to configure the SearchParameters class to meet the requirements. Which properties should you configure for the following scenarios?

Scenario 2:

Organize search results by displaying counts for different product categories.

A) Facets

B) Filter

C) Fuzzy

D) Limit

Answer: C

Feedback(if correct):

The correct answer is option C) Facets because configuring the Facets property allows you to categorize search results and display counts for different values within each category, effectively organizing search results by product categories.

Feedback(if wrong):

Options A) Fuzzy, B) Filter, and D) Limit are incorrect selections because they are not directly related to organizing search results by displaying counts for different product categories. Fuzzy is used for approximate matching, Filter is used to narrow down search results based on specific criteria, and Limit is used to restrict the number of search results returned, none of which address the requirement to organize search results by product categories.

Skill Mapping:

Skills: Azure Developer Certification (AZ-204)

Subskills: Connect to and consume Azure services and third-party services

Competencies: Azure Services Integration

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

49. You are developing a social media application that incorporates Azure Search for user content discovery. The application should enable users to search for posts based on various criteria and organize search results effectively. Configure the SearchParameters class to meet the requirements for the following scenarios.

Scenario 1:

Allow users to search for posts using advanced search options, including wildcards and fuzzy search.

A) Property

B) QueryType

C) OrderBy

D) SearchMode

Feedback(if correct):

The correct answer is option B) QueryType because configuring the QueryType property to "full" enables the use of the Lucene query syntax, which supports advanced search features like wildcards and fuzzy search, allowing users to perform advanced searches effectively.

Feedback(if wrong):

Options A) Property, C) OrderBy, and D) SearchMode are incorrect selections because they are not directly related to enabling users to perform advanced searches using wildcards and fuzzy searches. Property typically refers to specifying which fields to search within, OrderBy is used for sorting search results, and SearchMode defines the behavior of the search operation, none of which directly enable the use of wildcards and fuzzy search options for advanced searches.

Skill Mapping:

Skills: Azure Developer Certification (AZ-204)

Subskills: Connect to and consume Azure services and third-party services

Competencies: Azure Services Integration

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application

50. You are developing a social media application that incorporates Azure Search for user content discovery. The application should enable users to search for posts based on various criteria and organize search results effectively. Configure the SearchParameters class to meet the requirements for the following scenarios.

Scenario 2:

Organize search results by displaying counts for different user interests.

A) Limit

B) Filter

C) Fuzzy

D) Facets

Answer:D

Feedback(if correct):

The correct answer is option D) Facets because configuring the Facets property allows you to categorize search results and display counts for different values within each category, effectively organizing search results by user interests.

Feedback(if wrong):

Options A) Limit, B) Filter, and C) Fuzzy are incorrect selections because they are not directly related to organizing search results by displaying counts for different user interests. Limit is used to restrict the number of search results returned, Filter is used to narrow down search results based on specific criteria, and Fuzzy is used for approximate matching, none of which address the requirement to organize search results by user interests.

Skill Mapping:

Skills: Connect to and consume Azure services and third-party services.

Subskills: 5. Connect to and consume Azure services and third-party services.

Competencies: 5. Azure Services Integration.

Bloom's Taxonomy Level: Application.

51. You are developing an ASP.NET Core website to manage photographs stored in Azure Blob Storage containers. Users authenticate using Azure Active Directory (Azure AD) credentials. RBAC role permissions are implemented on the containers storing photographs, and users are assigned to RBAC roles.

Which setting should you configure in the Azure AD Application to enable the website to utilize user permissions with Azure Blob containers?

A) client_id

B) API

C) Permission

D) Type

Answer: C

Feedback(if correct): Option C) Permission is the correct setting to configure in the Azure AD Application because it specifies the permissions required by the application to access resources like Azure Blob containers based on user roles and RBAC permissions.

Feedback(if wrong):

- Option A) client_id: The client_id is a unique identifier for the application registered in Azure AD but does not directly relate to configuring permissions for accessing resources.

- Option B) API: While APIs are involved in granting permissions, specifying the API itself doesn't directly configure permissions for accessing Azure Blob containers.

- Option D) Type: The "Type" setting is not directly related to configuring permissions for accessing Azure Blob containers in the context of an Azure AD Application.

Skill Mapping:

Skills: Azure Developer Certification (AZ-204)

Subskill: Connect to and consume Azure services and third-party services

Competencies: Azure Services Integration, Azure Blob Storage, Azure Active Directory (Azure AD), Azure AD Application Configuration, Azure RBAC (Role-Based Access Control), Azure API Permissions

Difficulty Level: Intermediate

Bloom's Taxonomy Level: Application