AZ-204 Q&As

Developing Solutions for Microsoft Azure

Pass Microsoft AZ-204 Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

https://www.pass4itsure.com/az-204.html

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by Microsoft Official Exam Center

- **Instant Download** After Purchase
- **100% Money Back** Guarantee
- **365 Days** Free Update
- **800,000+** Satisfied Customers

QUESTION 1

HOTSPOT

You are working for Contoso, Ltd.

You define an API Policy object by using the following XML markup:

```
<set-variable name= "bodySize" value="@(context.Request.Headers["Content-Length"] [0])"/>
<choose>
 <when condition= "@(int.Parse(context.Variables.GetValueOrDefault<string> ("bodySize"))<512000)">
</when>
<otherwise>
     <rewrite-uri template= "/put"/>
     <set-backend-service base-url= "http://contoso.com/api/9.1/"/>
</otherwise>
</choose>
```

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

Hot Area:

Answer Area

Statement	Yes	No
The XML segment belongs in the <inbound> section of the policy.	O	O
If the body size is >256k, an error will occur.	O	O
If the request is http://contoso.com/api/9.2/, the policy will retain the higher version.	O	O

Correct Answer:

Answer Area

Statement	Yes	No
The XML segment belongs in the <inbound> section of the policy.	○	○
If the body size is >256k, an error will occur.	○	○
If the request is http://contoso.com/api/9.2/, the policy will retain the higher version.	○	○

Box 1: Yes

Use the set-backend-service policy to redirect an incoming request to a different backend than the one specified in the API settings for that operation. Syntax:

Box 2: No

The condition is on 512k, not on 256k.

Box 3: No

The set-backend-service policy changes the backend service base URL of the incoming request to the one specified in the policy.

Reference:

https://docs.microsoft.com/en-us/azure/api-management/api-management-transformation-policies

QUESTION 2

You are developing a software solution for an autonomous transportation system. The solution uses large data sets and Azure Batch processing to simulate navigation sets for entire fleets of vehicles.

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

What should you do?

A. In the Azure portal, add a Job to a Batch account.

B. In a .NET method, call the method: BatchClient.PoolOperations.CreateJob

C. In Python, implement the class: JobAddParameter

D. In Azure CLI, run the command: az batch pool create

E. In a .NET method, call the method: BatchClient.PoolOperations.CreatePool

F. In Python, implement the class: TaskAddParameter

G. In the Azure CLI, run the command: az batch account create

Correct Answer: B

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

Note:

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

your pool.

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

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

Incorrect Answers:

C, F: To create a Batch pool in Python, the app uses the PoolAddParameter class to set the number of nodes, VM size, and a pool configuration.

E: BatchClient.PoolOperations does not have a CreateJob method.

References: https://docs.microsoft.com/en-us/azure/batch/quick-run-dotnet https://docs.microsoft.com/en-us/azure/batch/quick-run-python

QUESTION 3

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

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

You need to deploy the website.

What should you do?

A. Deploy the website to a virtual machine. Configure the virtual machine to automatically scale when the CPU load is high.

B. Deploy the website to an App Service that uses the Shared service tier. Configure the App service plan to automatically scale when the CPU load is high.

C. Deploy the website to an App Service that uses the Standard service tier. Configure the App service plan to automatically scale when the CPU load is high.

D. Deploy the website to a virtual machine. Configure a Scale Set to increase the virtual machine instance count when the CPU load is high.

Correct Answer: C

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

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

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

Incorrect Answers:

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

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

QUESTION 4

HOTSPOT

You have an app that stores player scores for an online game. The app stores data in Azure tables using a class named PlayerScore as the table entity. The table is populated with 100,000 records.

You are reviewing the following section of code that is intended to retrieve 20 records where the player score exceeds 15,000. (Line numbers are included for reference only.)

```
1 public void GetScore(string.playerId, int score, string gameName)
2 {
3    Table Query<DynamicTableEntity> query – new TableQuery<DynamicTableEntity>().Select(new.string[.] { "Score" })
        .Where(TableQuery.GenerateFilterConditionForInt("Score", QueryComparisons.GretaerThanOrEqual, 15000)).Take
(20);
4    EntityResolver<KeyValuePair<string, int?>> resolver =
      (partitionKey, rowKey, ts, props, etag) => new KeyValuePair<string, int?>(rowKey, props["Score"].Int32Value);
5    foreach (var scoreItem in scoreTable.ExecuteQuery (query, resolver, null, null))
6    {
      Console.Writeline($"{scoreItem.Key} {scoreItem.Value}");
7    }
8   }

9  public class PlayerScore : TableEntity
10 {
11   public PlayerScore(string gameId, string playerId, int score, long timePlayed)
12   {
13     PartitionKey = gameId;
14     RowKey = playerId;
15     Score = score;
16     TimePlayed = timePlayed;
17   }
18   public int Score { get; set; }
19   public long TimePlayed { get; set; }
20   }
```


You have the following code. (Line numbers are included for reference only.)

```
01 public void SaveScore(string gameId, string playerId, int score, long timePlayed)
02 {
03   CloudStorageAccount storageAccount = CloudStorageAccount.Parse(connectionString);
04   CloudTableClient tableClient = storageAccount.CreateCloudTableClinet();
05   CloudTable table = tableClient.GetTableReference("scoreTable");
06   table.CreateIfNotExists();
07   var scoreRecord = new PlayerScore(gameId, playerId, score, timePlayed);
08   TableOperation insertOperation = TableOperation.Insert(scoreRecord);
09   table.Execute(insertOperation);
10  }
11  public class PlayerScore : TableEntity
12 {
13   public PlayerScore(string gameId, string playerId, int score, long timePlayed)
14   {
15     this.PartitionKey = gameId;
16     this.RowKey = playerId;
17   Score = score;
18   TimePlayed = timePlayed;
19   }
20   public int Score { get; set; }
21   public long TimePlayed { get; set; }
22 }
```

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

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

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

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

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

Hot Area:

	Yes	No
The code returns every Record where the surname equals Smith.	○	○
The table endpoint https://<mytableendpoint>/People (PartitionKey='Smith',RowKey='Steve')returns the same results as the code.	○	○

Correct Answer:

	Yes	No
The code returns every Record where the surname equals Smith.	○	◯
The table endpoint https://<mytableendpoint>/People (PartitionKey='Smith',RowKey='Steve')returns the same results as the code.	◯	○

QUESTION 5

HOTSPOT

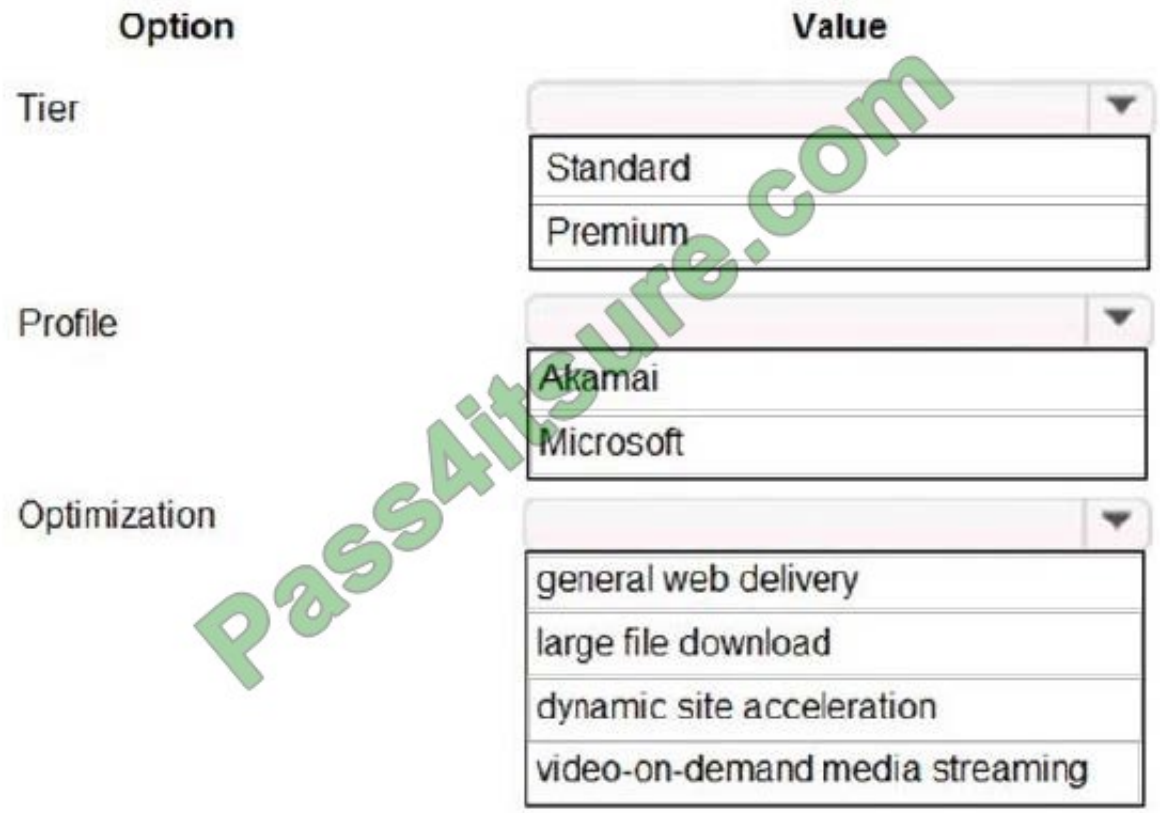
You need to configure Azure CDN for the Shipping web site.

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

NOTE: Each correct selection is worth one point.

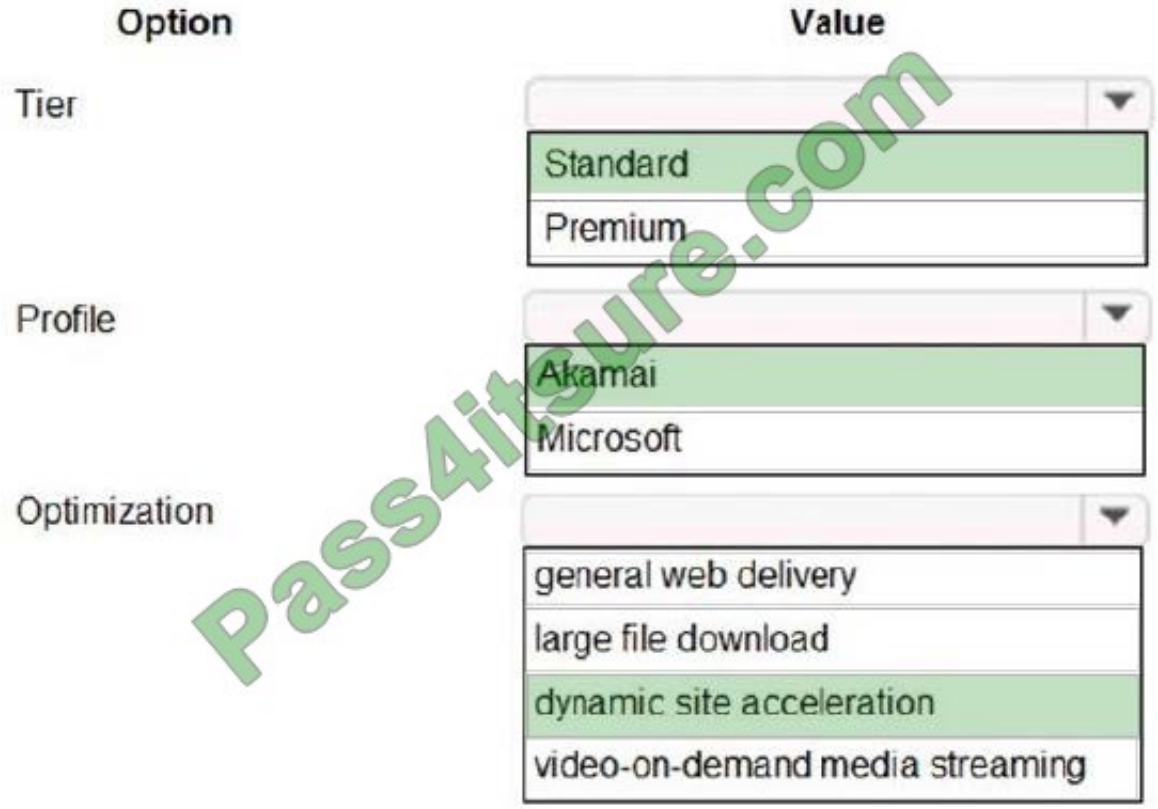
Hot Area:

Answer Area



Correct Answer:

Answer Area



Scenario: Shipping website Use Azure Content Delivery Network (CDN) and ensure maximum performance for dynamic content while minimizing latency and costs. Tier: Standard Profile: Akamai Optimization: Dynamic site acceleration Dynamic site acceleration (DSA) is available for Azure CDN Standard from Akamai, Azure CDN Standard from Verizon, and Azure CDN Premium from Verizon profiles. DSA includes various techniques that benefit the latency and performance of dynamic content. Techniques include route and network optimization, TCP optimization, and more. You can use this optimization to accelerate a web app that includes numerous responses that aren\\'t cacheable. Examples are search results, checkout transactions, or real-time data. You can continue to use core Azure CDN caching capabilities for static data.

References: https://docs.microsoft.com/en-us/azure/cdn/cdn-optimization-overview