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.geekcert.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 have an Azure Batch project that processes and converts files and stores the files in Azure storage. You are developing a function to start the batch job.

You add the following parameters to the function.

Parameter name	Description
fileTasks	a list of tasks to be run
jobId	the identifier that must be assigned to the job
outputContainerSasUrl	a storage SAS URL to store successfully converted files
failedContainerSasUrl	a storage SAS URL to store copies of files that failed to convert.

You must ensure that converted files are placed in the container referenced by the outputContainerSasUrl parameter. Files which fail to convert are placed in the container referenced by the failedContainerSasUrl parameter.

You need to ensure the files are correctly processed.

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

NOTE: Each correct selection is worth one point.

Hot Area:

Answer Area

```
public list<CloudTasks> StartTasks(List<FileTask> fileTasks, string jobId,
 string outputContainerSasUrl, string failedContainerSasUrl)
{
  BatchSharedKeyCredentials sharedKeyCredentials =
   new BatchSharedKeyCredentials(batchAccountUrl, batchAccountName,
batchAccountKey);
 List<CloudTask> tasks = new List<CloudTask>();
 using (BatchClient batchClient = BatchClient.Open(sharedKeyCredentials))
{
   CloudJob = batchClient.JobOperations. [ ▼ ] ();
                                          GetJob
                                          GetTask
                                          EnableJob
                                          CreateJob
       job.Id = jobId,
       job.PoolInfromation = new PoolInformation { PoolId = poolId };
       job.Commit();
       fileTasks.ForEach((fileTask) =>
       {
         string taskId = $"Task{DateTime.Now.ToFileTimeUtc().ToString()}";
         CloudTask task = new CloudTask (taskId, fileTask.Command);
         List<OutputFile> outputFileList = new List<OutputFile>();
         OutputFileBlobContainerDestination outputContainer =
          new OutputFileBlobContainerDestination(outputContainerSasUrl);
        OutputFileBlobContainerDestination failedContainer =
          new OutputFileBlobContainerDestination (failedContainerSasUrl);
        outputFileList.Add(new OutputFile(fileTask.Output,
          new OutputFileDestination(outputContainer),
    new OutputFileUploadOptions(OutputFileUploadCondition. [ ▼ ] )));
                                                            TaskSuccess
                                                            TaskFailure
                                                            TaskCompletion
    outputFileList.Add(new OutputFile(fileTask.Output,
     new OutputFileDestination(failedContainer),
     new OutputFileUploadOptions(OutputFileUploadCondition, [ ▼ ] )));
                                                            TaskSuccess
                                                            TaskFailure
                                                            TaskCompletion
     task [ ▼ ] =outputFileList;
          OutputFiles
          FilesToStage
          ResourceFiles
          StageFiles
     task.Add(task);
     });
     ]
     return tasks,
}
```

Correct Answer:

Answer Area

```
public list<CloudTasks> StartTasks(List<FileTask> fileTasks, string jobId,
 string outputContainerSasUrl, string failedContainerSasUrl)
{
  BatchSharedKeyCredentials sharedKeyCredentials =
   new BatchSharedKeyCredentials(batchAccountUrl, batchAccountName,
batchAccountKey);
 List<CloudTask> tasks = new List<CloudTask>();
 using (BatchClient batchClient = BatchClient.Open(sharedKeyCredentials))
{
   CloudJob = batchClient.JobOperations. [ ▼ ] ();
                                          GetJob
                                          GetTask
                                          EnableJob
                                          CreateJob
       job.Id = jobId,
       job.PoolInfromation = new PoolInformation { PoolId = poolId };
       job.Commit();
       fileTasks.ForEach((fileTask) =>
       {
         string taskId = $"Task{DateTime.Now.ToFileTimeUtc().ToString()}";
         CloudTask task = new CloudTask (taskId, fileTask.Command);
         List<OutputFile> outputFileList = new List<OutputFile>();
         OutputFileBlobContainerDestination outputContainer =
          new OutputFileBlobContainerDestination(outputContainerSasUrl);
        OutputFileBlobContainerDestination failedContainer =
          new OutputFileBlobContainerDestination (failedContainerSasUrl);
        outputFileList.Add(new OutputFile(fileTask.Output,
          new OutputFileDestination(outputContainer),
    new OutputFileUploadOptions(OutputFileUploadCondition. [ ▼ ] )));
                                                            TaskSuccess
                                                            TaskFailure
                                                            TaskCompletion
    outputFileList.Add(new OutputFile(fileTask.Output,
     new OutputFileDestination(failedContainer),
     new OutputFileUploadOptions(OutputFileUploadCondition, [ ▼ ] )));
                                                            TaskSuccess
                                                            TaskFailure
                                                            TaskCompletion
     task [ ▼ ] =outputFileList;
          OutputFiles
          FilesToStage
          ResourceFiles
          StageFiles
     task.Add(task);
     });
     ]
     return tasks,
}
```

Box 1: CreateJob

Box 2: TaskSuccess

TaskSuccess: Upload the file(s) only after the task process exits with an exit code of 0.

Incorrect: TaskCompletion: Upload the file(s) after the task process exits, no matter what the exit code was.

Box 3: TaskFailure

TaskFailure:Upload the file(s) only after the task process exits with a nonzero exit code.

Box 4: OutputFiles

To specify output files for a task, create a collection of OutputFile objects and assign it to the CloudTask.OutputFiles property when you create the task.

References:

https://docs.microsoft.com/en-us/dotnet/api/microsoft.azure.batch.protocol.models.outputfileuploadcondition

https://docs.microsoft.com/en-us/azure/batch/batch-task-output-files

QUESTION 2

DRAG DROP

You develop a web app that uses tier D1 app service plan by using the Web Apps feature of Microsoft Azure App Service.

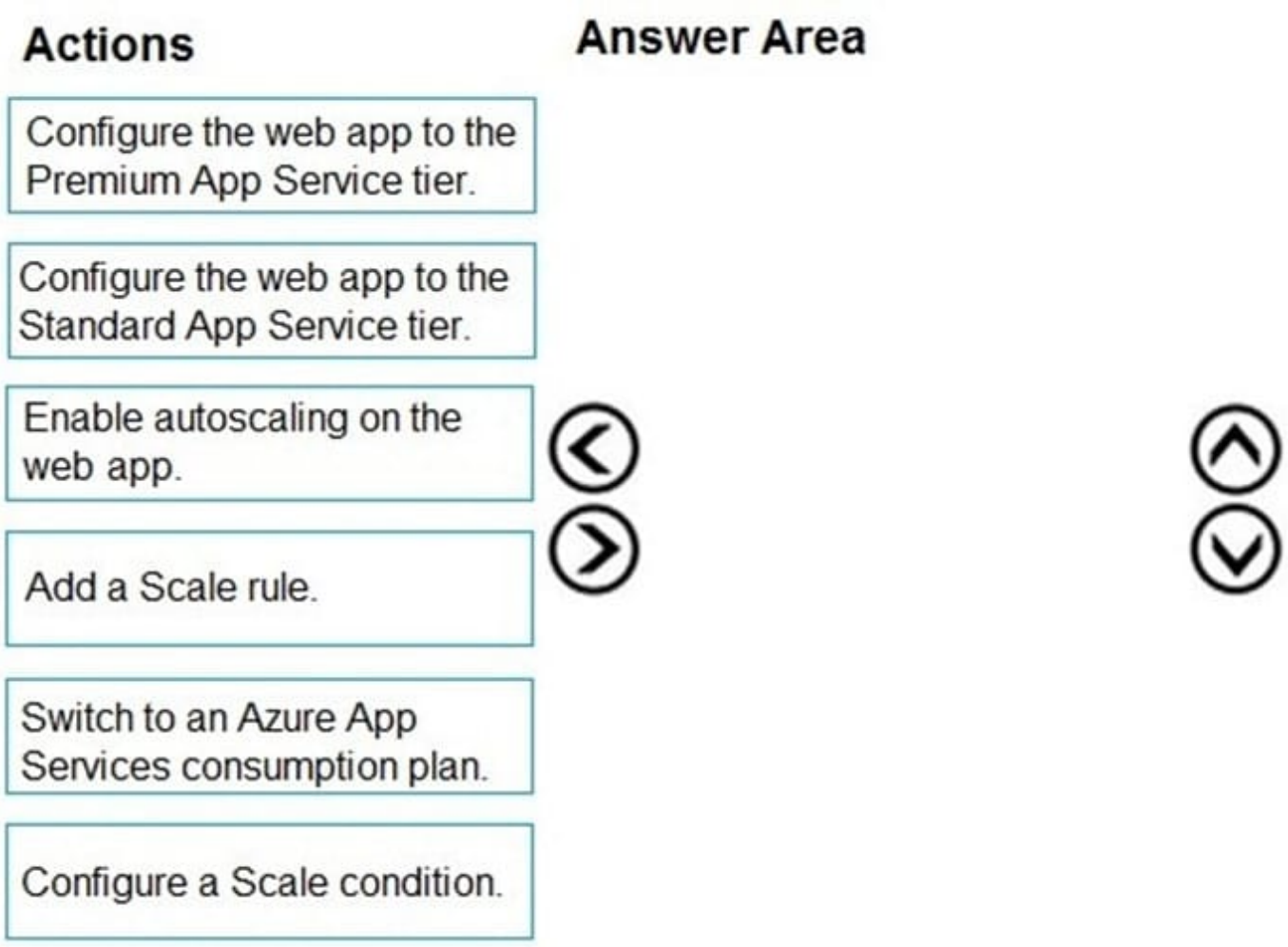
Spikes in traffic have caused increases in page load times.

You need to ensure that the web app automatically scales when CPU load is about 85 percent and minimize costs.

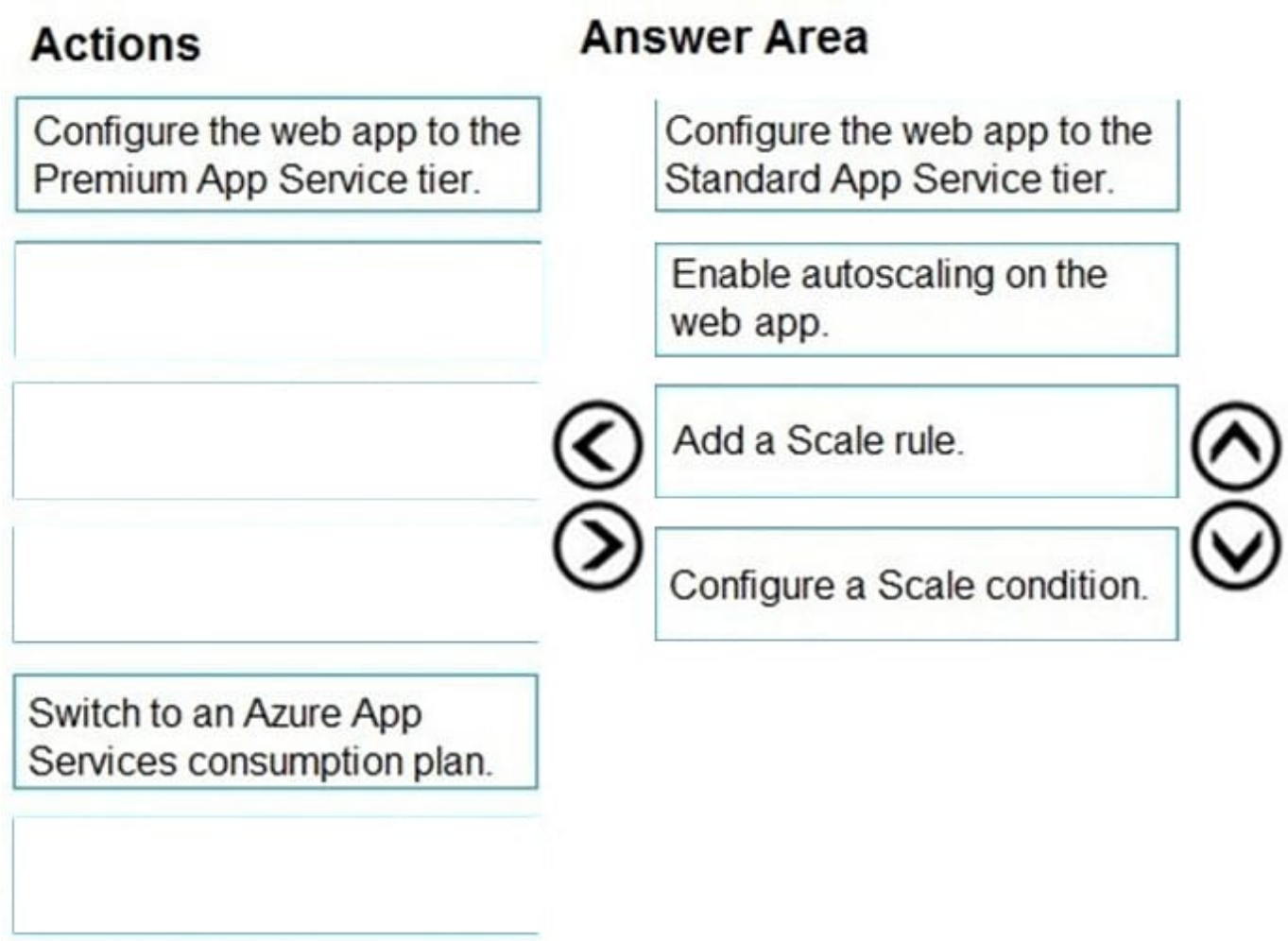
Which four actions should you perform in sequence? To answer, move the appropriate actions from the list of actions to the answer area and arrange them in the correct order.

NOTE: More than one order of answer choices is correct. You will receive credit for any of the correct orders you select.

Select and Place:



Correct Answer:



Step 1: Configure the web app to the Standard App Service Tier

The Standard tier supports auto-scaling, and we should minimize the cost.

Step 2: Enable autoscaling on the web app

First enable autoscale

Step 3: Add a scale rule

Step 4: Add a Scale condition

https://docs.microsoft.com/en-us/azure/monitoring-and-diagnostics/monitoringautoscale-get-started

https://azure.microsoft.com/en-us/pricing/details/app-service/plans/

QUESTION 3

HOTSPOT

You plan to deploy a new application to a Linux virtual machine (VM) that is hosted in Azure.

The entire VM must be secured at rest by using industry-standard encryption technology to address organizational security and compliance requirements.

You need to configure Azure Disk Encryption for the VM.

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

NOTE: Each correct selection is worth one point.

Hot Area:

Answer Area

```
az provider register -n Microsoft.KeyVault
resourcegroup= "myResourceGroup"
az group create - -name $resourcegroup - -location westus
keyvault name=myvaultname$RANDOM
az [ ▼ ] create\
```

- vm
- keyvault
- keyvault key
- vm encryption

```
- -name $keyvault_name \
- -resource -group $resourcegroup\
- -locstion eastus \
- -enabled for-disk-encryption True

az [ ▼ ] create\
```

- vm
- keyvault
- keyvault key
- vm encryption

```
 - -vault-name $keyvault_name\
 - -name Name1 \
 - -protection software
az [ ▼ ] create\
```

- vm
- keyvault
- keyvault key
- vm encryption

```
- -resource -group $resourcegroup \
- -name Name2
- -image Canonical:UbuntuServer:16.04=LTS:latest \
- -admin-username azureuser \
- -generate-ssh-keys \
- -data-disk-sizes-gb 5

az [ ▼ ] create\
```

- vm
- keyvault
- keyvault key
- vm encryption

```
 - -resource-group $resourcegroup \
 - -name Name2 \
 - -disk-encryption-keyvault $keyVault_name \
 - -key-encryption-key Name1 \
 - -volume-type
```

[▼]

- all
- data
- os

Correct Answer:

Answer Area

```
az provider register -n Microsoft.KeyVault
resourcegroup= "myResourceGroup"
az group create - -name $resourcegroup - -location westus
keyvault name=myvaultname$RANDOM
az [ ▼ ] create\
```

- vm
- keyvault
- keyvault key
- vm encryption

```
- -name $keyvault_name \
- -resource -group $resourcegroup\
- -locstion eastus \
- -enabled for-disk-encryption True

az [ ▼ ] create\
```

- vm
- keyvault
- keyvault key
- vm encryption

```
 - -vault-name $keyvault_name\
 - -name Name1 \
 - -protection software

az [ ▼ ] create\
```

- vm
- keyvault
- keyvault key
- vm encryption

```
- -resource -group $resourcegroup \
- -name Name2
- -image Canonical:UbuntuServer:16.04=LTS:latest \
- -admin-username azureuser \
- -generate-ssh-keys \
- -data-disk-sizes-gb 5

az [ ▼ ] create\
```

- vm
- keyvault
- keyvault key
- vm encryption

```
 - -resource-group $resourcegroup \
 - -name Name2 \
 - -disk-encryption-keyvault $keyVault_name \
 - -key-encryption-key Name1 \
 - -volume-type
```

[▼]

- all
- data
- os

Box 1: keyvault

Create an Azure Key Vault with az keyvault create and enable the Key Vault for use with disk encryption. Specify a unique Key Vault name for keyvault_name as follows:

keyvault_name=myvaultname$RANDOM

az keyvault create \

--name $keyvault_name \

--resource-group $resourcegroup \

--location eastus \

--enabled-for-disk-encryption True

Box 2: keyvault key

The Azure platform needs to be granted access to request the cryptographic keys when the VM boots to decrypt the virtual disks. Create a cryptographic key in your Key Vault with az keyvault key create. The following example creates a key

named myKey:

az keyvault key create \

--vault-name $keyvault_name \

--name myKey \

--protection software

Box 3: vm

Create a VM with az vm create. Only certain marketplace images support disk encryption. The following example creates a VM named myVM using an Ubuntu 16.04 LTS image:

az vm create \

--resource-group $resourcegroup \

--name myVM \

--image Canonical:UbuntuServer:16.04-LTS:latest \

--admin-username azureuser \

--generate-ssh-keys \

Box 4: vm encryption

Encrypt your VM with az vm encryption enable:

az vm encryption enable \

--resource-group $resourcegroup \

--name myVM \

--disk-encryption-keyvault $keyvault_name \

--key-encryption-key myKey \

--volume-type all

Note: seems to an error in the question. Should have enable instead of create.

Box 5: all

Encrypt both data and operating system.

References:

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

---

QUESTION 4

You are developing an e-Commerce Web App.

You want to use Azure Key Vault to ensure that sign-ins to the e-Commerce Web App are secured by using Azure App Service authentication and Azure Active Directory (AAD).

What should you do on the e-Commerce Web App?

A. Run the az keyvault secret command.

B. Enable Azure AD Connect.

C. Enable Managed Service Identity (MSI).

D. Create an Azure AD service principal.

Correct Answer: C

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

Reference:

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

https://docs.microsoft.com/en-us/samples/azure-samples/app-service-msi-keyvault-dotnet/keyvault-msi- appservice-sample/

QUESTION 5

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.	○	○
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.	○	○

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