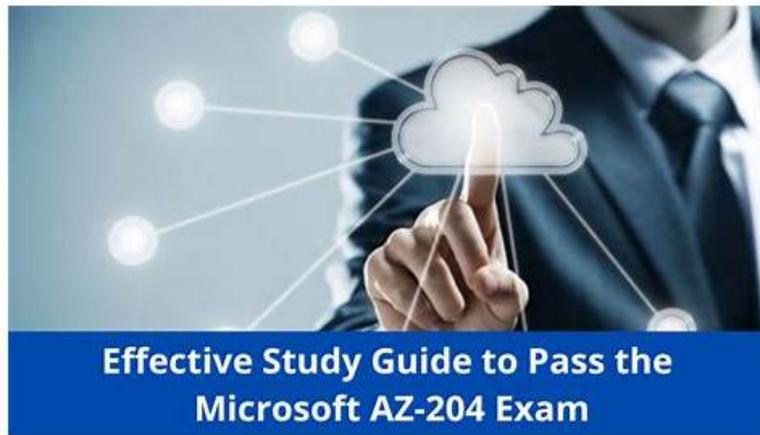


# Pass4sure AZ-204 Pass Guide & AZ-204 Premium Files



BTW, DOWNLOAD part of Braindumpsqa AZ-204 dumps from Cloud Storage: [https://drive.google.com/open?id=1y41ois\\_5Lh6dyv0ngVSKYggZ6YdKzXTw](https://drive.google.com/open?id=1y41ois_5Lh6dyv0ngVSKYggZ6YdKzXTw)

If you are lack of skills in the preparation of getting the certification, our AZ-204 study materials are the best choice for you. Many people have successfully realized economic freedom after getting the AZ-204 certificate and changing a high salary job. So you need to act from now, come to join us and struggle together. Our AZ-204 Study Materials will help you change into social elite and you will never feel disappointed.

As the saying goes, knowledge has no limits. You may be old but the spirit of endless learning won't be old. If you attend the test of AZ-204 certification you will update your stocks of knowledge and improve your actual abilities, buying our AZ-204 exam practice materials can help you pass the test smoothly. There are no threshold limits to attend the AZ-204 test such as the age, sexuality, education background and your job conditions, and anybody who wishes to improve their volume of knowledge and actual abilities can attend the AZ-204 test.

>> Pass4sure AZ-204 Pass Guide <<

## AZ-204 Quiz Torrent: Developing Solutions for Microsoft Azure - AZ-204 Quiz Braindumps & AZ-204 Study Guide

Knowledge is defined as intangible asset that can offer valuable reward in future, so never give up on it and our AZ-204 exam preparation can offer enough knowledge to cope with the exam effectively. To satisfy the needs of exam candidates, our experts wrote our AZ-204 practice materials with perfect arrangement and scientific compilation of messages, so you do not need to study other numerous AZ-204 study guide to find the perfect one anymore.

## Microsoft Developing Solutions for Microsoft Azure Sample Questions (Q449-Q454):

### NEW QUESTION # 449

You need to configure Azure App Service to support the REST API requirements.

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

NOTE: Each correct selection is worth one point.

| Setting        | Value   |
|----------------|---|
| Plan           | <input type="text"/> <ul style="list-style-type: none"> <li>Basic</li> <li>Standard</li> <li>Premium</li> <li>Isolated</li> </ul> |
| Instance Count | <input type="text"/> <ul style="list-style-type: none"> <li>1</li> <li>10</li> <li>20</li> <li>100</li> </ul>                     |

**Answer:**

Explanation:

| Setting        | Value  |
|----------------|--|
| Plan           | <input type="text"/> <ul style="list-style-type: none"> <li>Basic</li> <li><b>Standard</b></li> <li>Premium</li> <li>Isolated</li> </ul> |
| Instance Count | <input type="text"/> <ul style="list-style-type: none"> <li>1</li> <li><b>10</b></li> <li>20</li> <li>100</li> </ul>                     |

Reference:

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

**NEW QUESTION # 450**

You are developing an application to securely transfer data between on-premises file systems and Azure Blob storage. The application stores keys, secrets, and certificates in Azure Key Vault. The application uses the Azure Key Vault APIs. The Application must allow recovery of an accidental deletion of the key vault or key vault objects. Key Vault objects must be retained for 90 days after deletion.

You need to protect the key vault and key vault object.

Which Azure key Vault feature should you use?



**Answer:**

Explanation:



Explanation  
Soft delete  
Soft delete

### NEW QUESTION # 451

You need to ensure that PolicyLib requirements are met.

How should you complete the code segment? To answer, drag the appropriate code segments to the correct locations. Each code segment may be used once, more than once, or not at all. You may need to drag the split bar between panes or scroll to view content.

NOTE: Each correct selection is worth one point.

| Code segments                                     | Answer Area  |
|---|--|
| Process   | <pre> public class IncludeEventId : {     public void         (ITelemetry telemetry)     {         .Properties["EventId"] =         ;     } } </pre> |
| Initialize  |  |
| telemetry.Sequence                                |  |
| ITelemetryProcessor                               |  |
| ITelemetryInitializer                             |  |
| telemetry.Context                                 |  |
| EventGridController.EventId.Value                 |  |
| ((EventTelemetry)telemetry).Properties["EventId"] |  |

Answer:

Explanation:

| Code segments                                     | Answer Area  |
|---|--|
| Process   | <pre> public class IncludeEventId : {     public void         (ITelemetry telemetry)     {         .Properties["EventId"] =         ;     } } </pre> |
| Initialize  |  |
| telemetry.Sequence                                |  |
| ITelemetryProcessor                               |  |
| ITelemetryInitializer                             |  |
| telemetry.Context                                 |  |
| EventGridController.EventId.Value                 |  |
| ((EventTelemetry)telemetry).Properties["EventId"] |  |

Explanation

```

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

```

Scenario: You have a shared library named PolicyLib that contains functionality common to all ASP.NET Core web services and applications. The PolicyLib library must:

- \* Exclude non-user actions from Application Insights telemetry.
- \* Provide methods that allow a web service to scale itself.
- \* Ensure that scaling actions do not disrupt application usage.

Box 1: ITelemetryInitializer

Use telemetry initializers to define global properties that are sent with all telemetry, and to override selected behavior of the standard telemetry modules.

Box 2: Initialize

Box 3: Telemetry.Context

Box 4: ((EventTelemetry)telemetry).Properties["EventID"]

Reference:

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

Topic 5, Litware Inc

Case study

This is a case study. Case studies are not timed separately. You can use as much exam time as you would like to complete each case. However, there may be additional case studies and sections on this exam. You must manage your time to ensure that you are able to complete all questions included on this exam in the time provided.

To answer the questions included in a case study, you will need to reference information that is provided in the case study. Case studies might contain exhibits and other resources that provide more information about the scenario that is described in the case study. Each question is independent of the other questions in this case study.

At the end of this case study, a review screen will appear. This screen allows you to review your answers and to make changes before you move to the next section of the exam. After you begin a new section, you cannot return to this section.

To start the case study

To display the first question in this case study, click the Next button. Use the buttons in the left pane to explore the content of the case study before you answer the questions. Clicking these buttons displays information such as business requirements, existing environment, and problem statements. When you are ready to answer a question, click the Question button to return to the question.

Background

You are a developer for Litware Inc., a SaaS company that provides a solution for managing employee expenses. The solution consists of an ASP.NET Core Web API project that is deployed as an Azure Web App.

Overall architecture

Employees upload receipts for the system to process. When processing is complete, the employee receives a summary report email that details the processing results. Employees then use a web application to manage their receipts and perform any additional tasks needed for reimbursement.

Receipt processing

Employees may upload receipts in two ways:

- \* Uploading using an Azure Files mounted folder

- \* Uploading using the web application

Data Storage

Receipt and employee information is stored in an Azure SQL database.

Documentation

Employees are provided with a getting started document when they first use the solution. The documentation includes details on supported operating systems for Azure File upload, and instructions on how to configure the mounted folder.

Solution details

Users table

| Column         | Description  |
|----------------|--|
| UserId         | unique identifier for an employee                            |
| ExpenseAccount | employees expense account number in the format 1234-123-1234 |
| AllowedAmount  | limit of allowed expenses before approval is needed          |
| SupervisorId   | unique identifier for employee's supervisor                  |
| SecurityPin    | value used to validate user identity                         |

Web Application

You enable MSI for the Web App and configure the Web App to use the security principal name WebAppIdentity.

Processing

Processing is performed by an Azure Function that uses version 2 of the Azure Function runtime. Once processing is completed, results are stored in Azure Blob Storage and an Azure SQL database. Then, an email summary is sent to the user with a link to the processing report. The link to the report must remain valid if the email is forwarded to another user.

Logging

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

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

Requirements

Receipt processing

Concurrent processing of a receipt must be prevented.

Disaster recovery

Regional outage must not impact application availability. All DR operations must not be dependent on application running and must ensure that data in the DR region is up to date.

Security

- \* User's SecurityPin must be stored in such a way that access to the database does not allow the viewing of SecurityPins. The web application is the only system that should have access to SecurityPins.

- \* All certificates and secrets used to secure data must be stored in Azure Key Vault.

- \* You must adhere to the principle of least privilege and provide privileges which are essential to perform the intended function.

- \* All access to Azure Storage and Azure SQL database must use the application's Managed Service Identity (MSI).
- \* Receipt data must always be encrypted at rest.
- \* All data must be protected in transit.
- \* User's expense account number must be visible only to logged in users. All other views of the expense account number should include only the last segment, with the remaining parts obscured.
- \* In the case of a security breach, access to all summary reports must be revoked without impacting other parts of the system.

#### Issues

##### Upload format issue

Employees occasionally report an issue with uploading a receipt using the web application. They report that when they upload a receipt using the Azure File Share, the receipt does not appear in their profile. When this occurs, they delete the file in the file share and use the web application, which returns a 500 Internal Server error page.

##### Capacity issue

During busy periods, employees report long delays between the time they upload the receipt and when it appears in the web application.

##### Log capacity issue

Developers report that the number of log messages in the trace output for the processor is too high, resulting in lost log messages.

##### Application code

##### Processing.cs

```

PC01 public static class Processing
PC02 {
PC03     public static class Function
PC04     {
PC05         [FunctionName("IssueWork")]
PC06         public static async Task Run([TimerTrigger("0 */5 * * * *")] TimerInfo timer, ILogger
log)
PC07         {
PC08             var container = await GetCloudBlobContainer();
PC09             foreach (var fileItem in await ListFiles())
PC10             {
PC11                 var file = new CloudFile(fileItem.StorageUri.PrimaryUri);
PC12                 var ms = new MemoryStream();
PC13                 await file.DownloadToStreamAsync(ms);
PC14                 var blob = container.GetBlockBlobReference(fileItem.Uri.ToString());
PC15                 await blob.UploadFromStreamAsync(ms);
PC16             }
PC17         }
PC18     }
PC19     private static CloudBlockBlob GetDRBlob(CloudBlockBlob sourceBlob)
PC20     {
PC21         . . .
PC22     }
PC23     private static async Task<CloudBlobContainer> GetCloudBlobContainer()
PC24     {
PC25         var cloudBlobClient = new CloudBlobClient(new Uri(". . ."), await GetCredentials());
PC26
PC27         await cloudBlobClient.GetRootContainerReference().CreateIfNotExistsAsync();
PC28         return cloudBlobClient.GetRootContainerReference();
PC29     }
PC30     private static async Task<StorageCredentials> GetCredentials()
PC31     {
PC32         . . .
PC33     }
PC34     private static async Task<List<IListFileItem>> ListFiles()
PC35     {
PC36         . . .
PC37     }
PC37     private KeyVaultClient _keyVaultClient = new KeyVaultClient(". . .");
PC38 }
PC39 }
Database.cs

```

```

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

```

ReceiptUploader.cs

```

RU01 public class ReceiptUploader
RU02 {
RU03     public async Task UploadFile(string file, byte[] binary)
RU04     {
RU05         var httpClient = new HttpClient();
RU06         var response = await httpClient.PutAsync("-", new ByteArrayContent(binary));
RU07         while (ShouldRetry(response))
RU08         {
RU09             response = await httpClient.PutAsync("-", new ByteArrayContent(binary));
RU10         }
RU11     }
RU12     private bool ShouldRetry(HttpResponseMessage response)
RU13     {
RU14
RU15     }
RU16 }

```

ConfigureSSE.ps1

```

CS01 $storageAccount = Get-AzureRmStorageAccount -ResourceGroupName "..." -AccountName "..."
CS02 $keyVault = Get-AzureRmKeyVault -VaultName "..."
CS03 $key = Get-AzureKeyVaultKey -VaultName $keyVault.VaultName -Name "..."
CS04 Set-AzureRmKeyVaultAccessPolicy `
CS05     -VaultName $keyVault.VaultName `
CS06     -ObjectId $storageAccount.Identity.PrincipalId `
CS07
CS08
CS09 Set-AzureRmStorageAccount `
CS10     -ResourceGroupName $storageAccount.ResourceGroupName `
CS11     -AccountName $storageAccount.StorageAccountName `
CS12     -EnableEncryptionService File `
CS13     -KeyvaultEncryption `
CS14     -KeyName $key.Name
CS15     -KeyVersion $key.Version `
CS16     -KeyVaultUri $keyVault.VaultUri

```

#### NEW QUESTION # 452

An organization plans to deploy Azure storage services.

You need to configure shared access signature (SAS) for granting access to Azure Storage.

Which SAS types should you use? To answer, drag the appropriate SAS types to the correct requirements. Each SAS type may be

used once, more than once, or not at all. You may need to drag the split bar between panes or scroll to view content.  
 NOTE: Each correct selection is worth one point.

**SAS types**

- Account-level
- Service-level
- User delegation

**Answer Area**

- Requirement**
- Delegate access to resources in one or more of the storage services
  - Delegate access to a resource in a single storage service
  - Secure a resource by using Azure AD credentials

**SAS type**

- 
- 
- 

**Answer:**

Explanation:

**SAS types**

- Account-level
- Service-level
- User delegation

**Answer Area**

- Requirement**
- Delegate access to resources in one or more of the storage services
  - Delegate access to a resource in a single storage service
  - Secure a resource by using Azure AD credentials

**SAS type**

- Account-level
- Service-level
- User delegation

Reference:

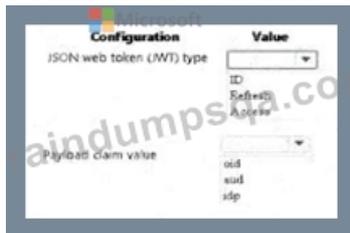
<https://docs.microsoft.com/en-us/azure/storage/common/storage-sas-overview>

**NEW QUESTION # 453**

YOU need to reliably identify the delivery driver profile information.

How should you configure the system? To answer, select the appropriate options in the answer area.

NOTE Each correct selection is worth one point.



**Answer:**

Explanation:



Explanation:





