NAME: GOWTHAM RAM

DEPARTMENT: TESTING

DATE: 27/01/2002

------------------------------------------------------------------------------------------------------------------------------

**AZURE SCENARIO BASED QUESTIONS**

**Scenario 1**: Your team needs to deploy a virtual machine in Azure portal or CLI to test a new software application. The team has requested both Windows and Linux machines.

**Q1**: How could you set up these virtual machines? What considerations are needed for pricing and OS licensing?

**WINDOWS:**

STEPS:

1. Login with valid credentials in Azure portal.
2. Under the azure service we can find “Create a resource”.
3. Click “Create a resource”.
4. Then we can find Popular Azure services, under that we can find “Virtual machine”.
5. Then click “Create”.
6. Choose Azure subscription.
7. Configure the basics: Resource group, virtual machine name, region, VM size, Set authentication (username and password).
8. Networking set up.
9. Review the VM and then click create.

**LINUX:**

For Linux we need to follow the same steps as we follow in windows.

**Pricing:**

1. Pricing depends on the size of the virtual machine. If we choose the larger size, then it costs more.
2. It also differs based on the region.
3. Auto shutdowns save the cost when VM is not in use.
4. Use existing Windows Server or SQL Server licenses to save on Windows VM costs.

**OS licensing:**

1. Windows: Azure includes the cost of the Windows license in the VM pricing. You don't need to bring your own license unless specified.
2. Linux: Most Linux distributions are free; however, some might have additional licensing fees.

------------------------------------------------------------------------------------------------------------------------------

**Scenario 2**: The IT security team has requested that sensitive data stored in Azure storage account be encrypted to meet compliance requirements.

**Q1**: How could you ensure the data stored in Azure storage is encrypted, and what encryption types are available?

\*\*\*How you can ensure the data stored in an **Azure Storage Account** is encrypted\*\*\*

To ensure the data is stored in an azure storage account, we can rely on the default Azure built in service (Storage Service Encryption). When we try to encrypt the data in storage account it (SSE) will automatically be enabled by default.

**\*\*\***Types**\*\*\***

* Storage Service Encryption (SSE) with Microsoft-managed keys
* Client-side Encryption: Data is encrypted before uploading to Azure
* Customer-managed Keys: we can use our keys in Azure Key Vault for additional control

-------------------------------------------------------------------------------------------------------------------------------

**Scenario 3**: You are responsible for setting up a DevOps pipeline in Azure DevOps for your application. The pipeline must deploy code to an Azure app service and notify the team if the deployment fails.

**Q1**: How could you configure this pipeline to meet these requirements?

**Steps:**

1. First, we need to upload our code in a repository
2. Login to Azure DevOps account
3. Select the project
4. Navigate to pipeline
5. Click new pipeline and create
6. Select source repo
7. Choose the pipeline and configure its type (Eg: YAML)
8. Write or update an yml file in a repo
9. Save the file into a root directory
10. Goto project settings and then Create service connection
11. New service connection then selects or add new Azure resource manager
12. Choose service principle (Manual or automatic)
13. Save the pipeline and run it.
14. Monitor the progress

**Notify the team when pipeline fails:**

Example: Email notification

1. Goto project settings
2. Click Notification
3. Create new subscription
4. Choose build pipeline failure
5. Add team member's email
6. Save it

------------------------------------------------------------------------------------------------------------------------------

**Scenario 4:** Your organization is moving its premises SQL database to Azure. The database must remain accessible during migration with minimal downtime.

**Q1:** Which Azure service could you use, and how could you perform the migration?

**Steps:**

To migrate the SQL Database on-premises to Azure with minimal downtime, we can use Azure Database Migration Service. This service ensures that the availability of databases continuously through the migration process.

1. Use Azure data migration assistant to check compatibility.
2. Choose the azure SQL deployment option
3. Configure the Migration: In DMS, create a migration project, specify the source and target databases and install the DMS agent on the on-premises server.
4. Select Premium SKU for online migration and configure networking and security settings
5. Configure Source and Databases: Enter on-premises SQL server details and give Azure SQL database connection details
6. For Online minimal downtime: use DMS to replicates data continuously from the on-premises database to azure
7. Cut over to Azure: Stop changes in on-premises database and start the final data sync process to ensure all are migrated
8. Monitoring the Azure Databases

For Minimal downtime:

To migrate an on-premises SQL database to Azure, use Azure Database Migration Service (DMS) with online migration. By using this service, we can continuously replicate the data, but we need to ensure the on-premises database is active. Then we need to perform final data sync after that switch to Azure database and monitor it.