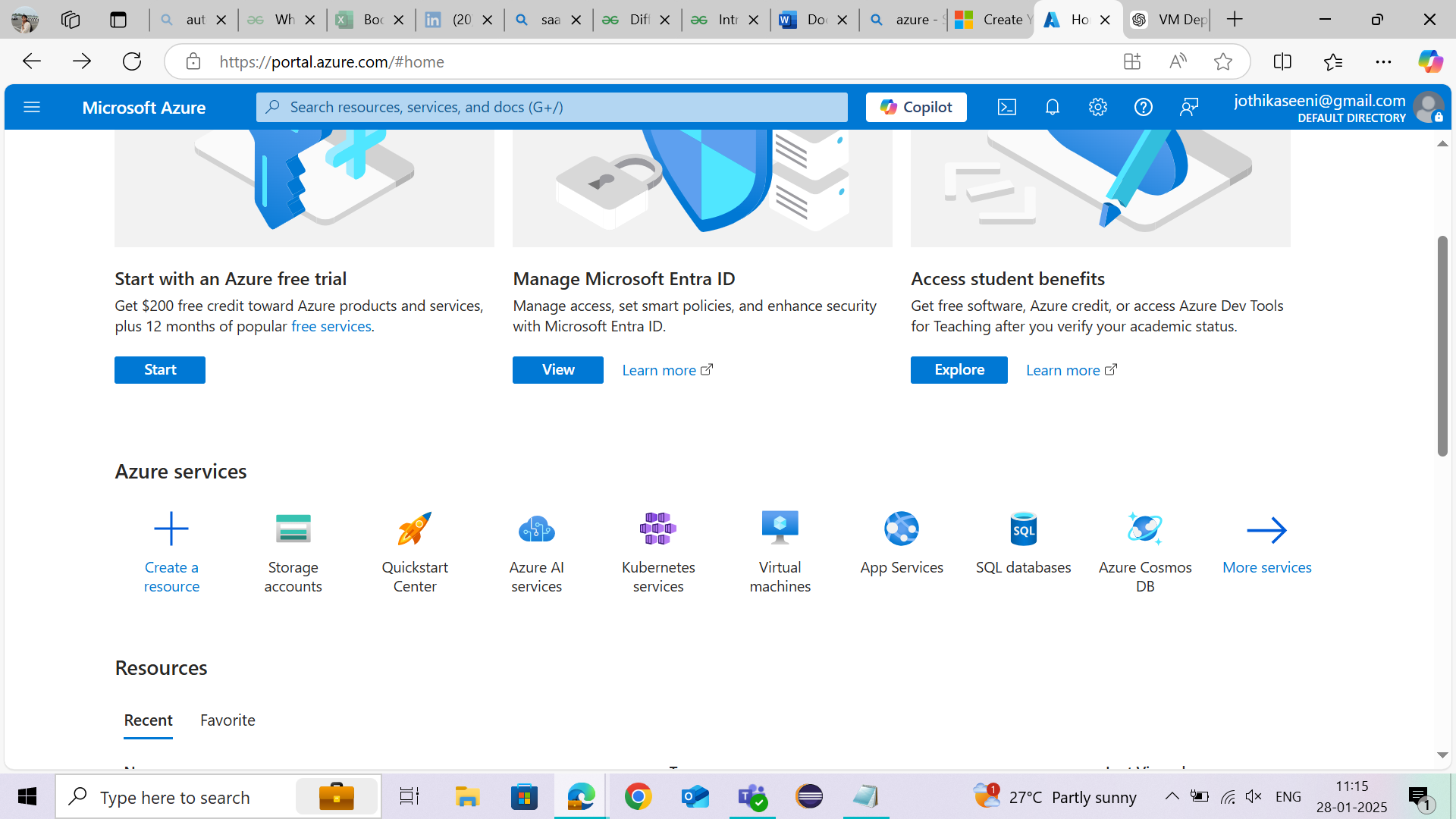
**Name: Jothika S**  
**Date: 27-01-2025**  
**Domain: Testing**

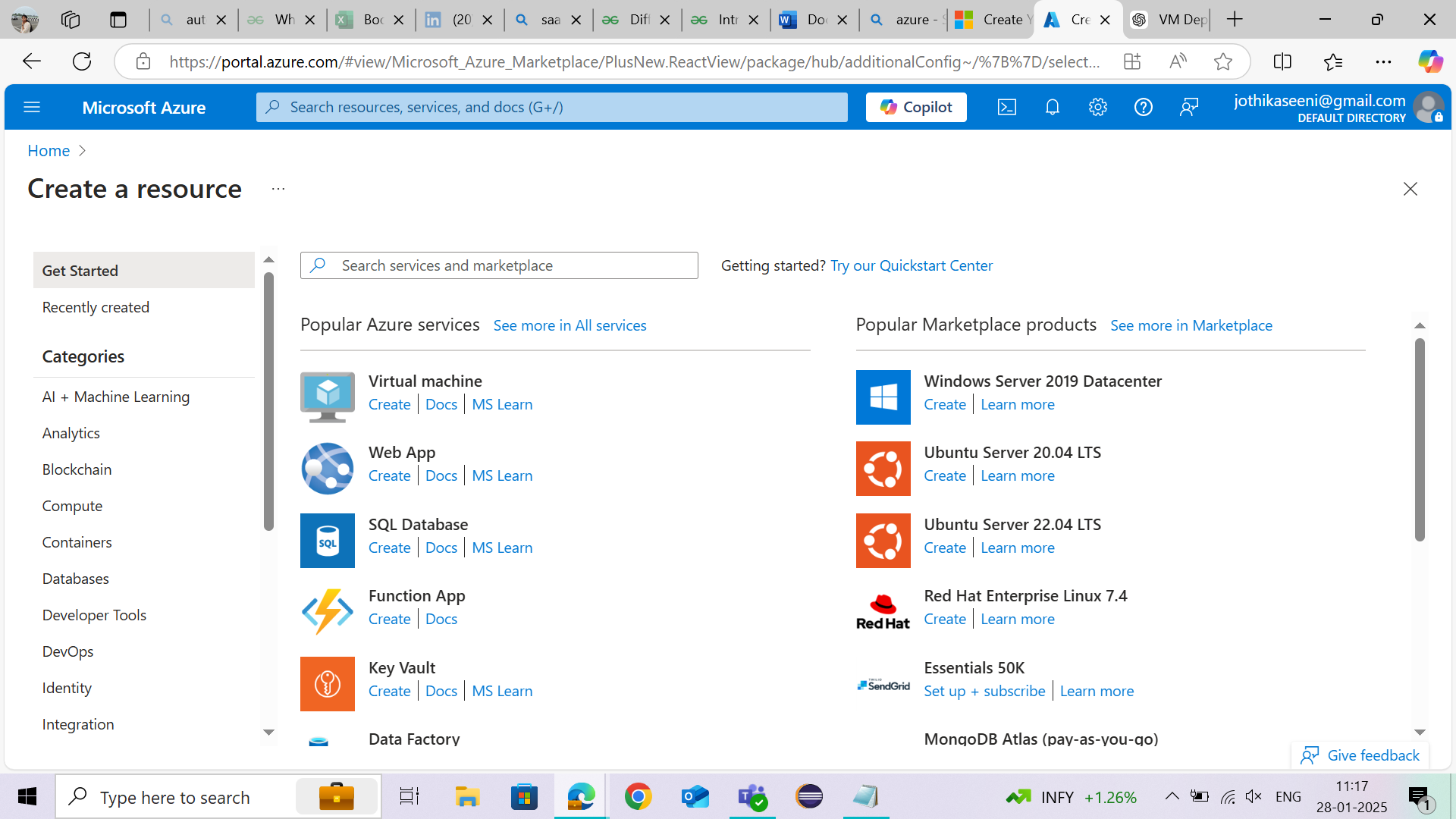
**AZURE TASK**

**Scenario 1:** Your team needs to deploy a virtual machine in azure portal to test a new software application. Here, the team has requested both windows and Linux virtual machine.

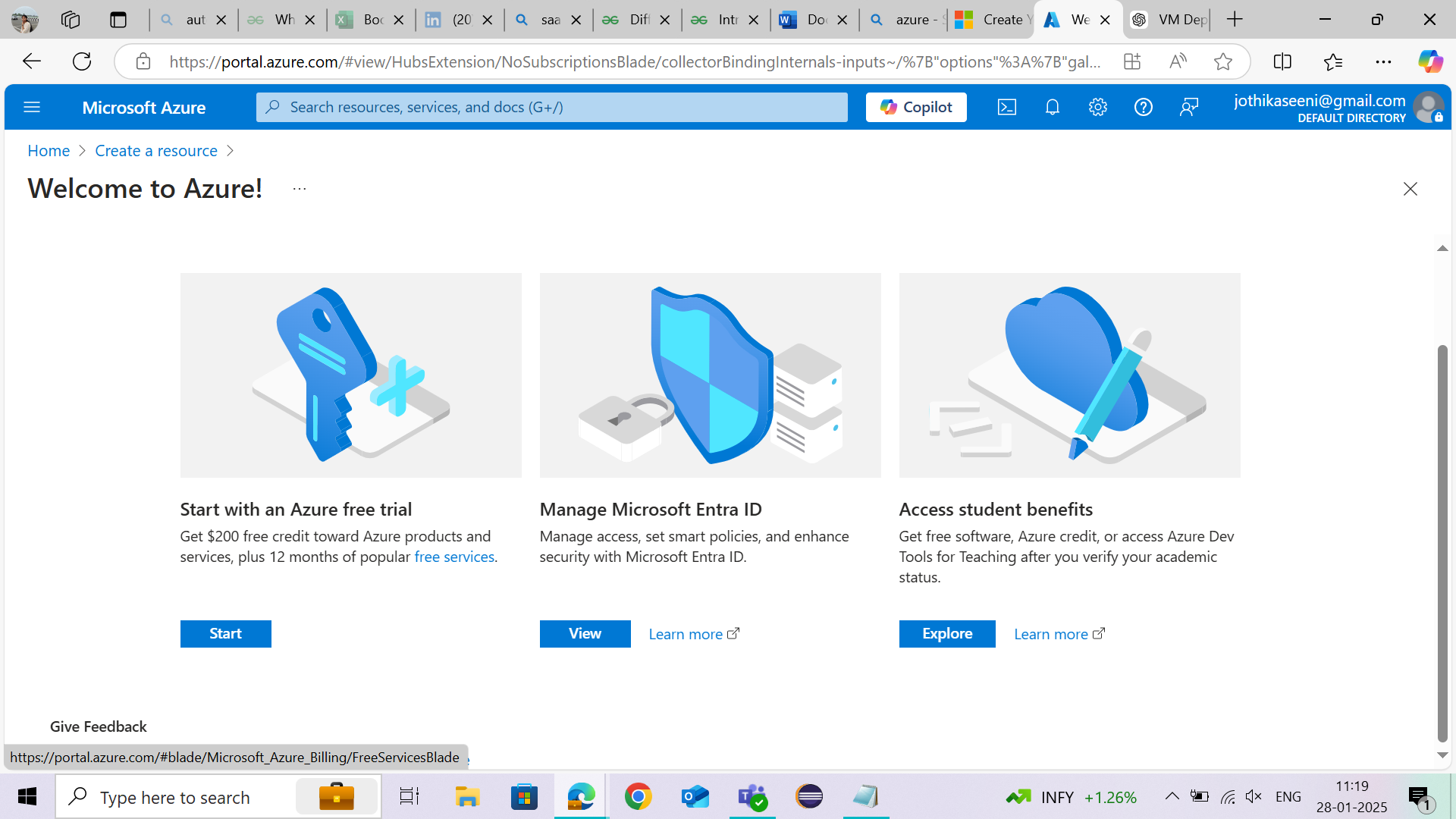
**Q1**: How could you setup these virtual machines? and what consideration need for prizing and OS licensing?

**Ans**:  
Step1: Login to the Azure portal.  
Step 2: click “create resource” in Azure services.



Step 3: Select “virtual Machine” in the resource.  
  


Step 4: select **Start with an Azure free trial**



Step 5: Setup basic configuration for windows Virtual Machine – Create a Resource Group, Virtual Machine name, Region, Image - Choose a **Windows Server** version / choose Linux distribution, Size, Choose the Authentication Type as SSH public key or Password.  
Step 6: Review and create.  
Step 7: Review all the setup we configure and click create.

**Pricing and OS licensing:**  
Windows VMs: Higher cost due to included OS licensing; you can use Azure Hybrid Benefit to reduce costs if you have existing licenses.  
Linux VMs: Generally cheaper because most distributions are free; paid Linux distributions like RHEL and SUSE may have additional charges for premium support.  
  
**Scenario 2:** The IT security team has requested that sensitive data has stored in Azure storage account be encrypted to meet compliance requirement.

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

Option 1: **Microsoft-managed keys**

By default, SSE automatically encrypts data at rest using Microsoft-managed keys without any configuration needed on all Azure Storage accounts.  
Option 2: **Use Customer-managed keys**

* Go to the **Azure Portal** and navigate to your **Storage Account**.
* In the Storage Account settings, select **Encryption**.
* Choose **Customer-managed keys** and select the encryption key from **Azure Key Vault** that you created.
* Save the changes.

At last check in Storage Account, under **Encryption** settings, ensure that encryption is enabled either with Microsoft-managed keys or Customer-managed keys.

Check Azure Security Center or Azure Monitor to keep track of encryption status and get

alerts on any issues.

**Available types of encryptions in azure:**

* SSE with Microsoft-managed keys
* SSE with Customer-managed keys
* Encryption in transit (TLS/SSL).

**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.

**Question**: How could you configure this pipeline to meet this requirement?

Step 1: Go to Azure DevOps and sign in   
Step 2: Click New Project and name it  
Step 3: Select Private/Public repo, Version Control and Work Item.   
Step 4: Click Create.  
Step 5: Navigate to your project, Import the repo and push the code   
Step 6: Go to Pipelines and Click New Pipeline.   
Step 7: Select the repository   
Step 8: Choose “Starter Pipeline”  
Step 9: Navigate to Azure DevOps and select Project Settings.  
Step 10: Click on New Service Connection and Select Azure Resource Manager.  
Step 11: Choose Service Principle.   
Step 12: Select your Subscription and App Service   
Step 13: Click Save   
Step 14: Again, go to Project Settings and navigate to Notifications   
Step 15: Click New Subscription   
Step 16: Select Build Completed

Step 17: Set the condition to trigger only on failures   
Step 18: Add team’s email addresses   
Step 19: Click Save   
Step 20: Go to Pipelines and select the pipeline   
Step 21: Click Run & Check logs   
Step 22: If a failure occurs, email notification will be sent.

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

**Q1**: Which azure would you use, how could you perform the migration?

**Answer:**   
Set up an Azure SQL Database or Azure SQL Managed Instance in the Azure portal. Ensure your on-premises SQL Server is up-to-date.

Set up Azure Database Migration Service (DMS) in the Azure portal. Install the DMS migration agent on your on-premises SQL Server.

In DMS, start migrating your data from the on-premises SQL Server to Azure. The database remains accessible during this step.

Set up continuous data sync between the on-premises SQL Server and Azure SQL to keep both databases in sync.

When the data is fully synced, pause the on-premises database to ensure no new changes. Complete the migration by switching to the Azure database.

Update your applications to connect to the new Azure SQL database.