Course: Cloud and Network Security C3-2024

Student Name: Loise Murugi Murage

Student No: **CS-CNS07-24115**.

Sunday, November 10, 2024

Week 9 Assignment 1

Class Exercise: Key Vault (Implementing Secure Data by setting up Always Encrypted)

Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)

Contents

Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)

Introduction

In this exercise, we will be using Azure SQL Database to encrypting content of columns in database tables by using Always Encrypted and set up automated ARM template deployment, to set up a Virtual Machine with Visual Studio 2019 .NET, and use C# language and SQL Server Management Studio 19.Then we will configure the Key Vault resource with a key and a secret, the keys and the secret will be stored in vault. Key Vault enables Microsoft Azure applications and users to store and use several types of secret/key data which are collectively referred to as objects.

We will be using JSON Azure Resource Manager (ARM) templates to enable us to define the infrastructure requirements.

For this lab, I have used West US region.

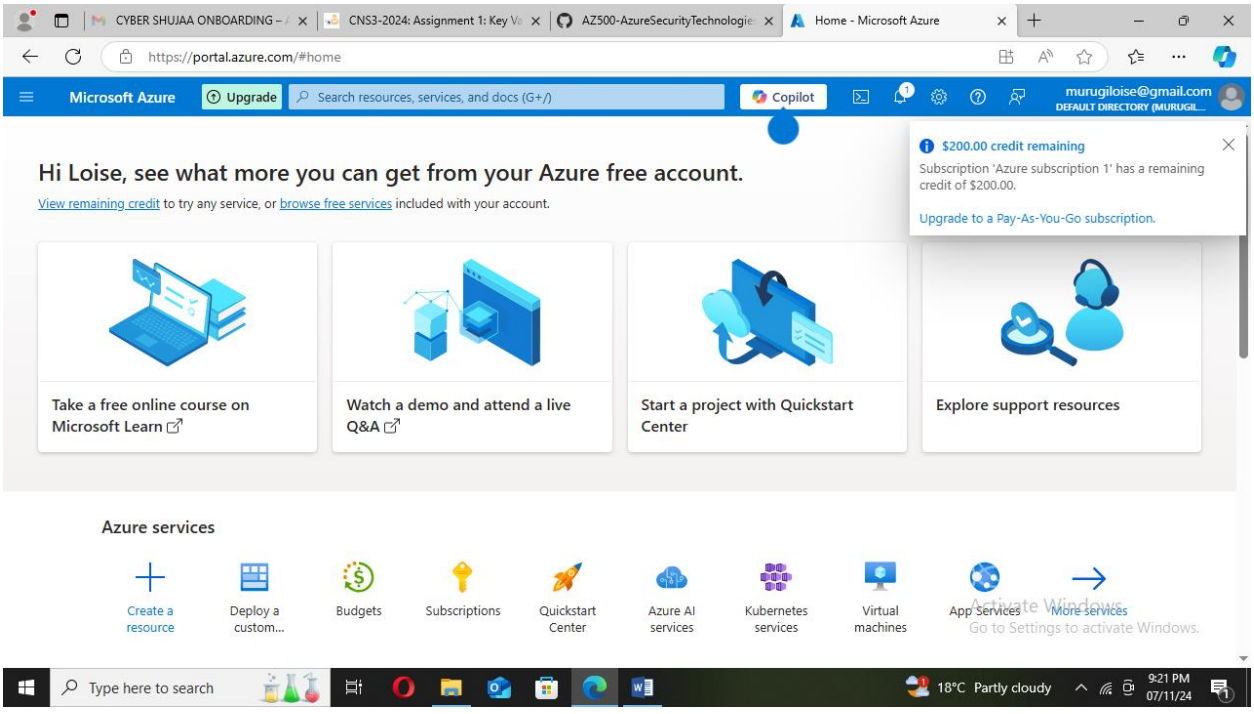
Instructions

Exercise 1: Deploy the base infrastructure from an ARM template

Task 1: Deploy an Azure VM and an Azure SQL database

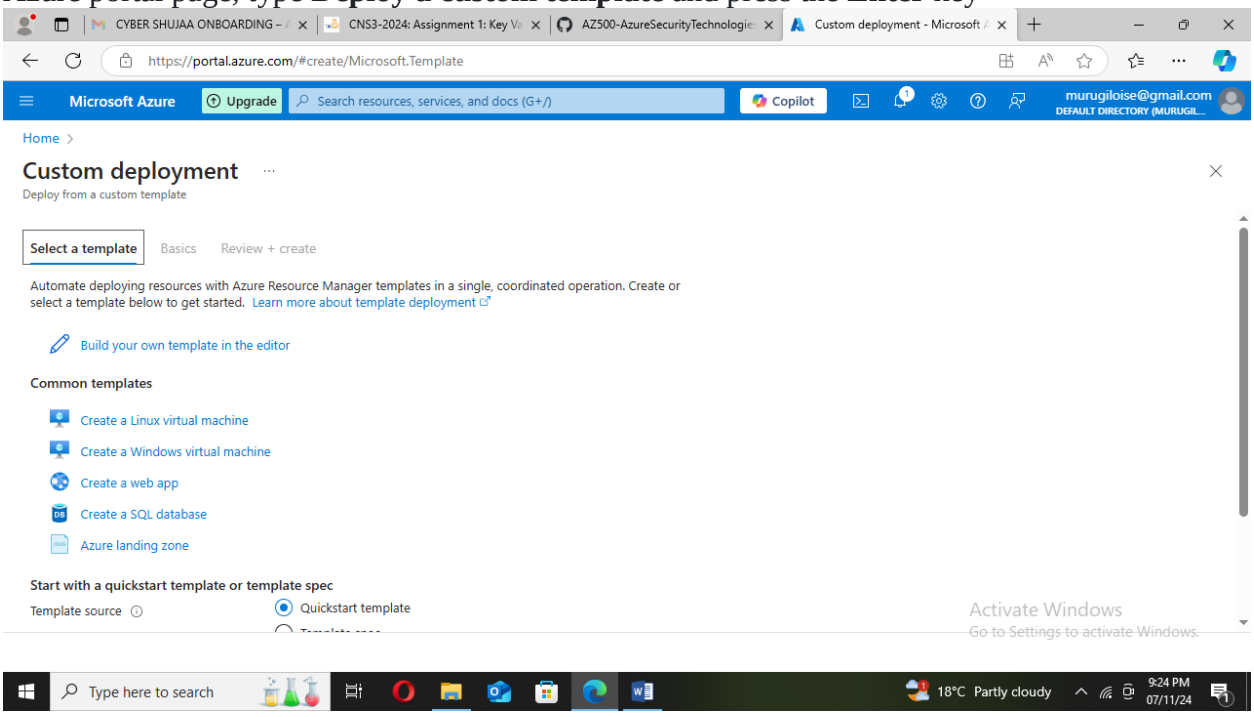
In this task, you will deploy an Azure VM, which will automatically install Visual Studio 2019 and SQL Server Management Studio 19 as part of the deployment.

1. Sign-in to the Azure portal **https://portal.azure.com/**.

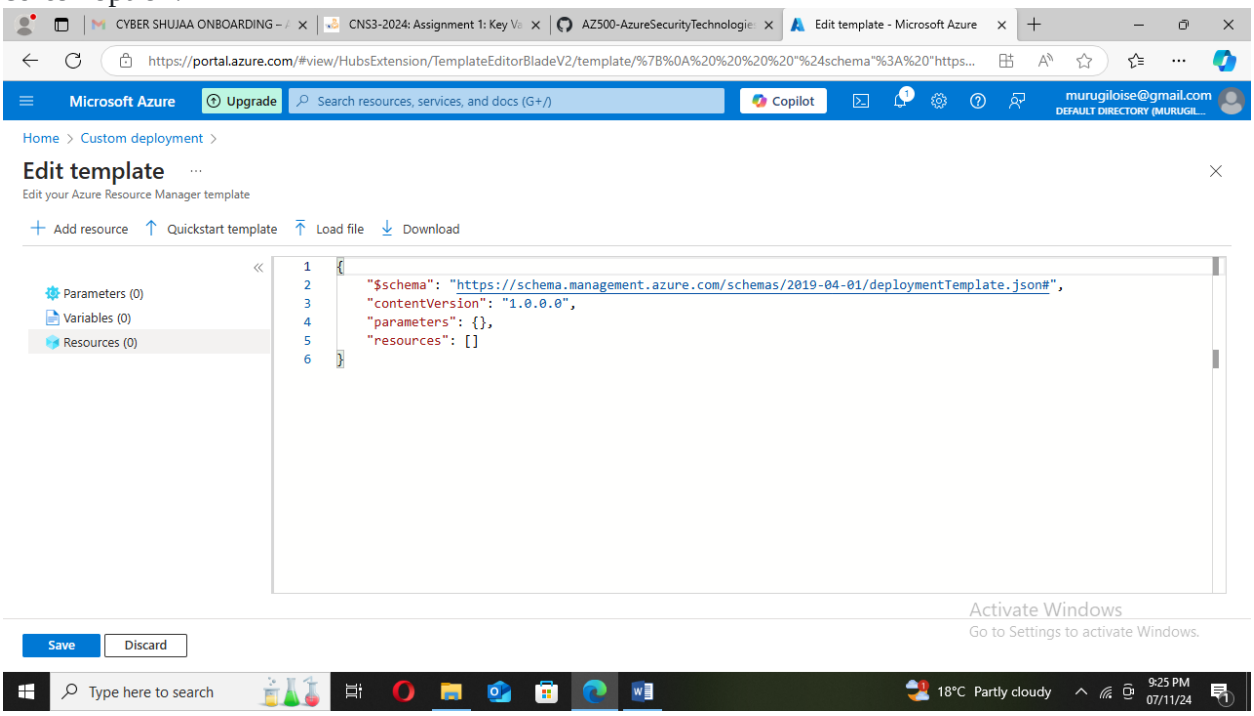


Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)

2. In the Azure portal, in the **Search resources, services, and docs** text box at the top of the Azure portal page, type **Deploy a custom template** and press the **Enter** key

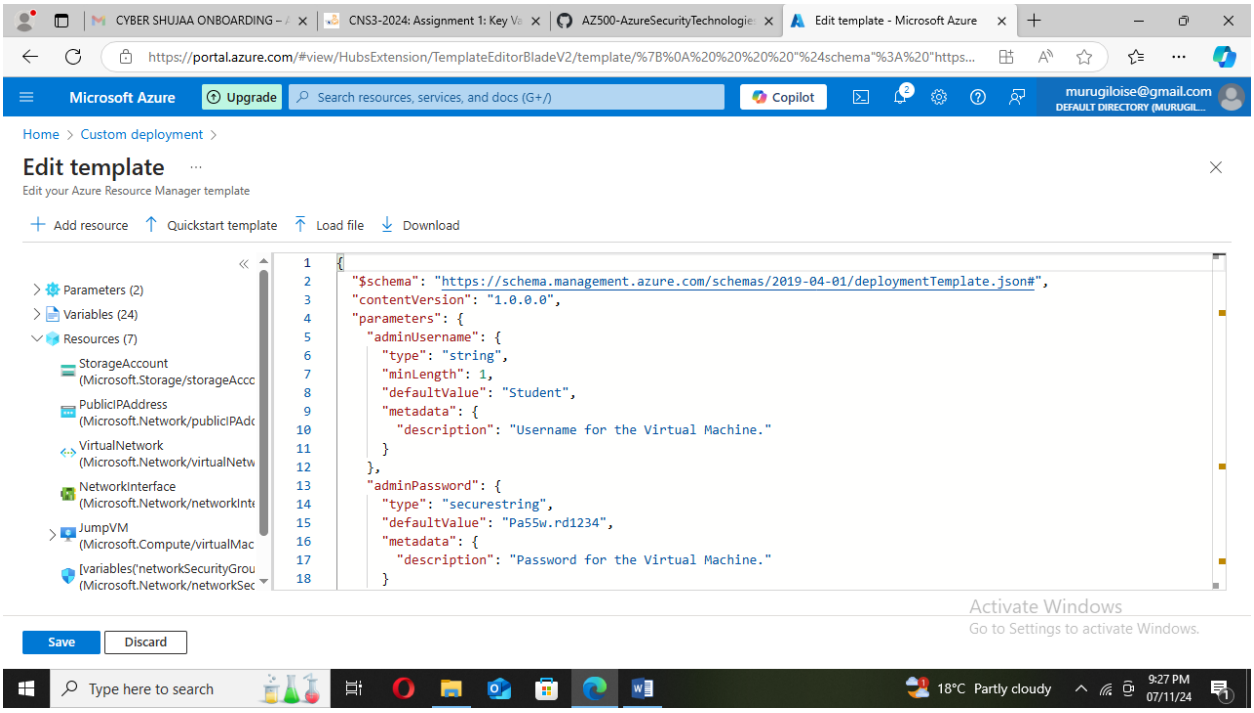


3. On the **Custom deployment** blade, click the **Build your own template in the editor** option.

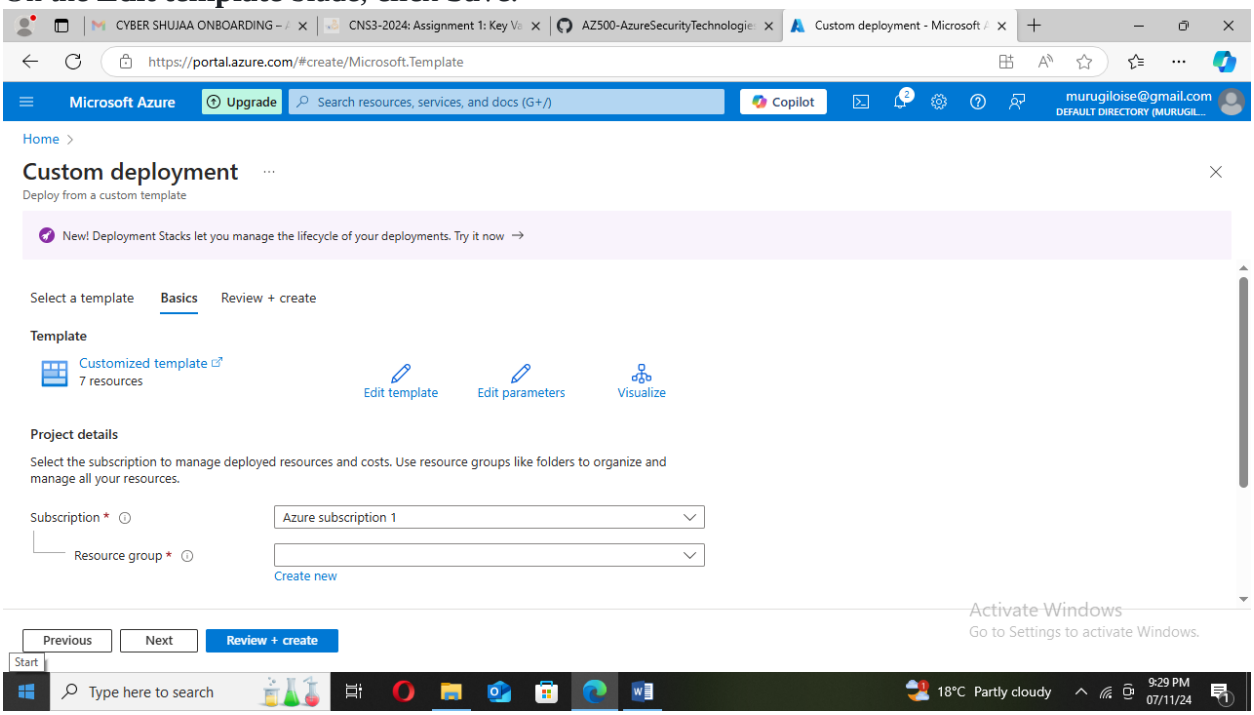


4. On the **Edit template** blade, click **Load file**, locate the **\Allfiles\Labs\10\az-500-10_azuredeploy.json** file and click **Open**.

Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)



5. On the **Edit template** blade, click **Save**.



6. On the **Custom deployment** blade, under **Deployment Scope** ensure that the following settings are configured (leave any others with their default values):

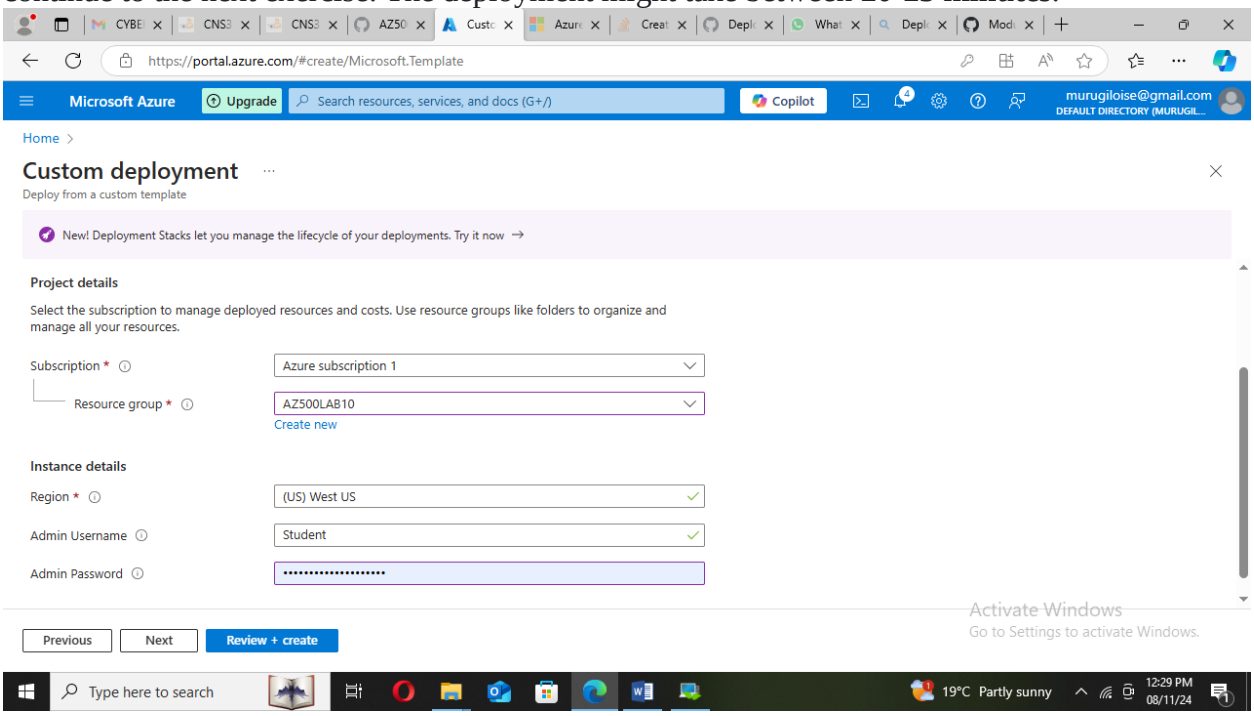
Setting	**Value**
Subscription	the name of the Azure subscription you will be using in this lab

Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)

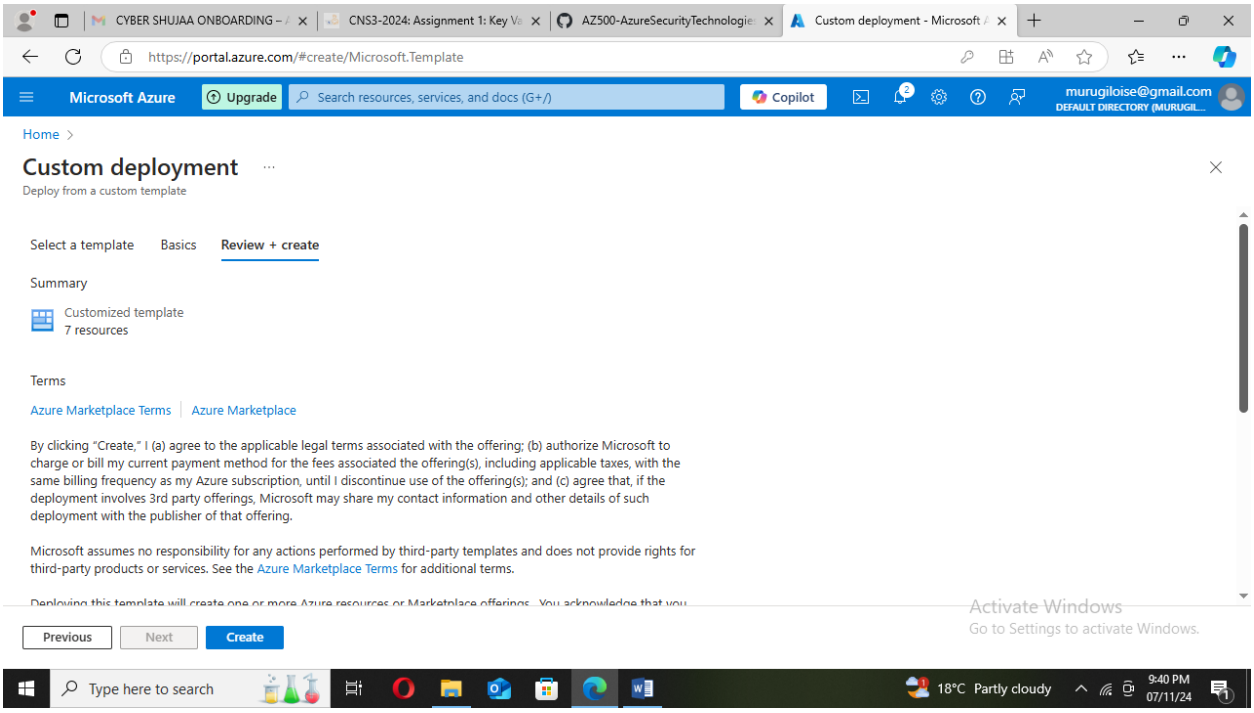
Setting	Value
Resource group	click **Create new** and type the name **AZ500LAB10**
Location	**East US**
Username	**Student**
Password	**Please use your personal password created in Lab 02 > Exercise 2 > Task 1 > Step 3.**

1. **Note**: While you can change the administrative credentials used for logging on to the Virtual Machine, you don't have to.7. Click the **Review and Create** button, and confirm the deployment by clicking the **Create** button.

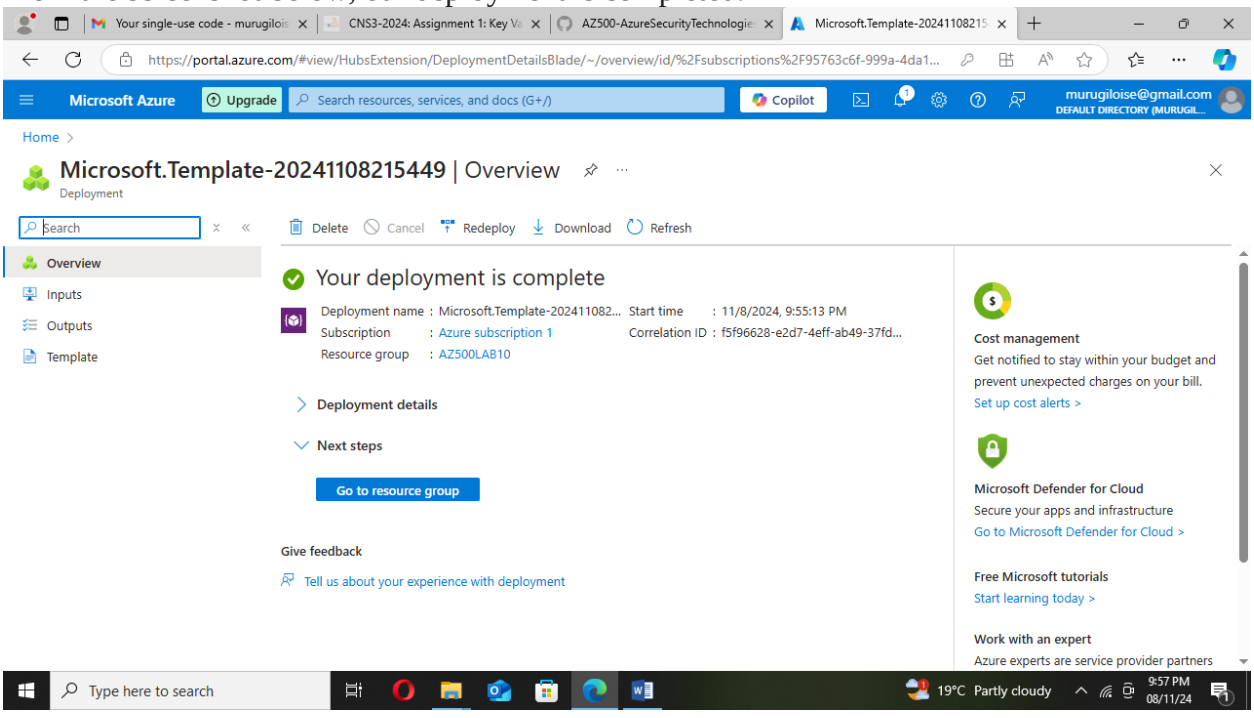
   Note: This initiates the deployment of the Azure VM and Azure SQL Database required for this lab.
   Note: Do not wait for the ARM template deployment to be completed, but instead continue to the next exercise. The deployment might take between **20-25 minutes**.



Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)



At this point we have initiated a deployment of the Azure VM and Azure SQL Database. From the screenshot below, our deployment is completed.



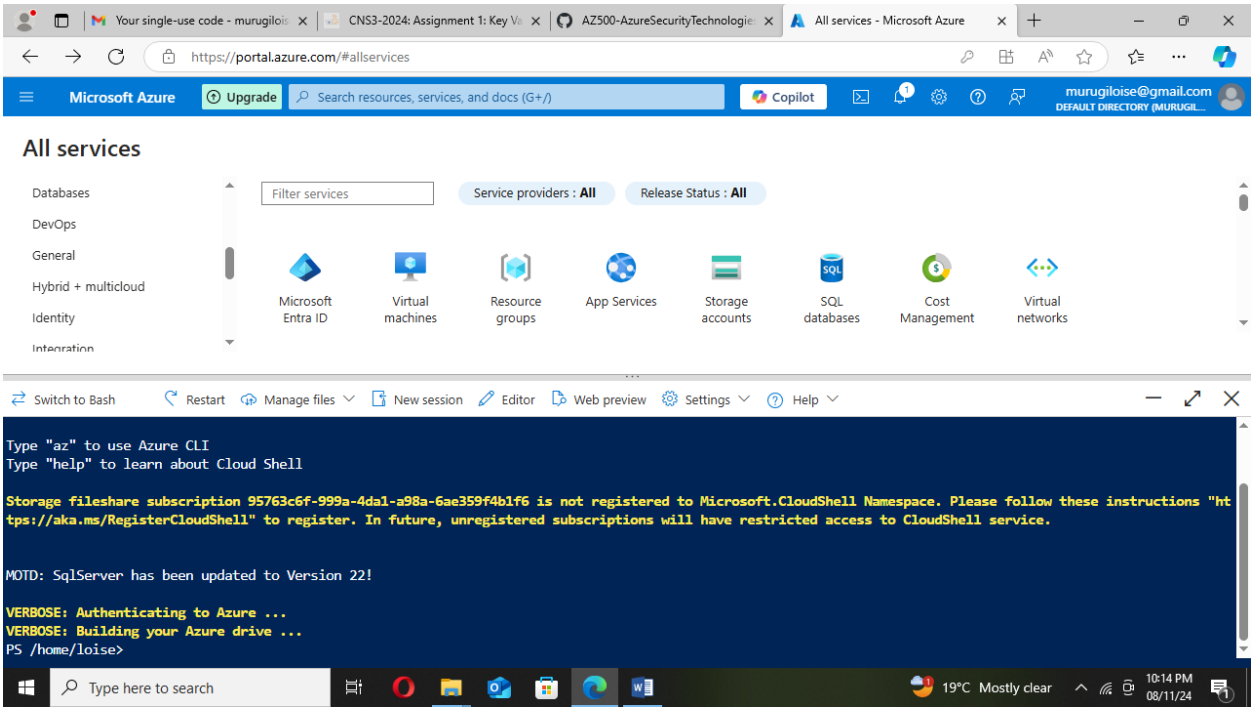
Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)

Exercise 2: Configure the Key Vault resource with a key and a secret

Task 1: Create and configure a Key Vault

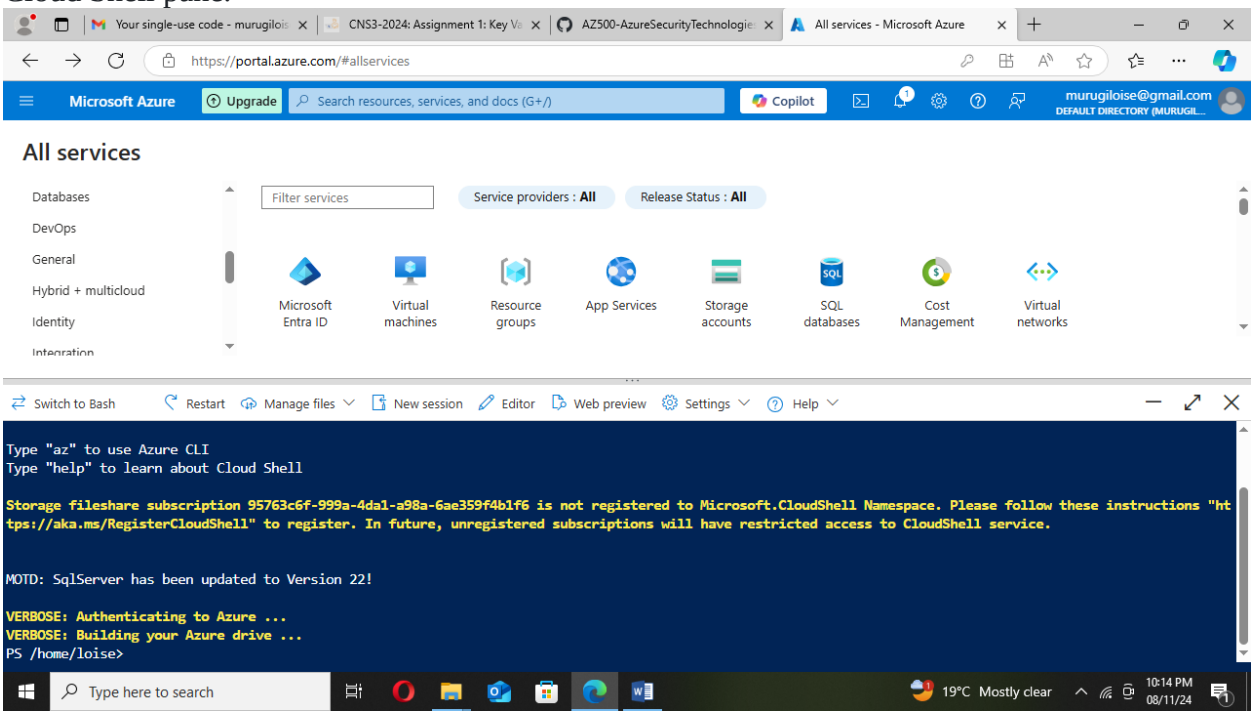
In this task, you will create an Azure Key Vault resource. You will also configure the Azure Key Vault permissions.

1. Open the Cloud Shell by clicking the first icon (next to the search bar) at the top right of the Azure portal. If prompted, select **PowerShell** and **Create storage**.



Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)

2. Ensure **PowerShell** is selected in the drop-down menu in the upper-left corner of the Cloud Shell pane.

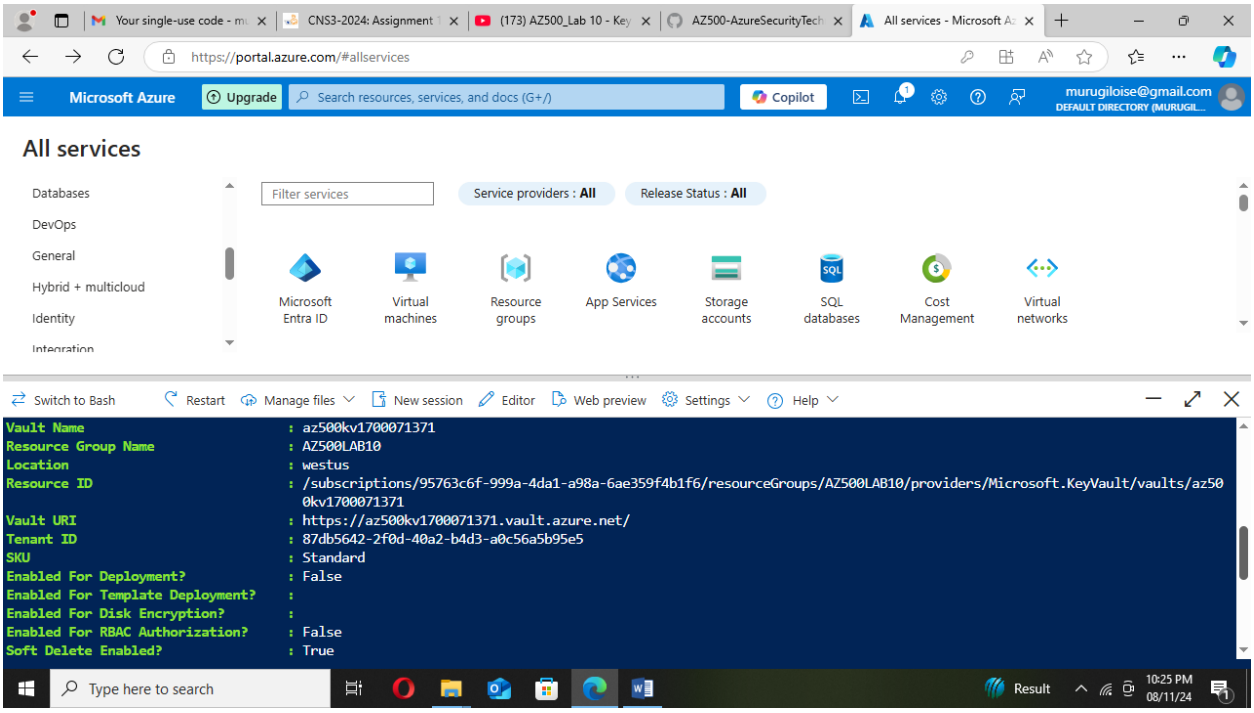


3. In the PowerShell session within the Cloud Shell pane, run the following to create an Azure Key Vault in the resource group **AZ500LAB10**. (If you chose another name for this lab's Resource Group out of Task 1, use that name for this task as well). The Key Vault name must be unique. Remember the name you have chosen. You will need it throughout this lab.

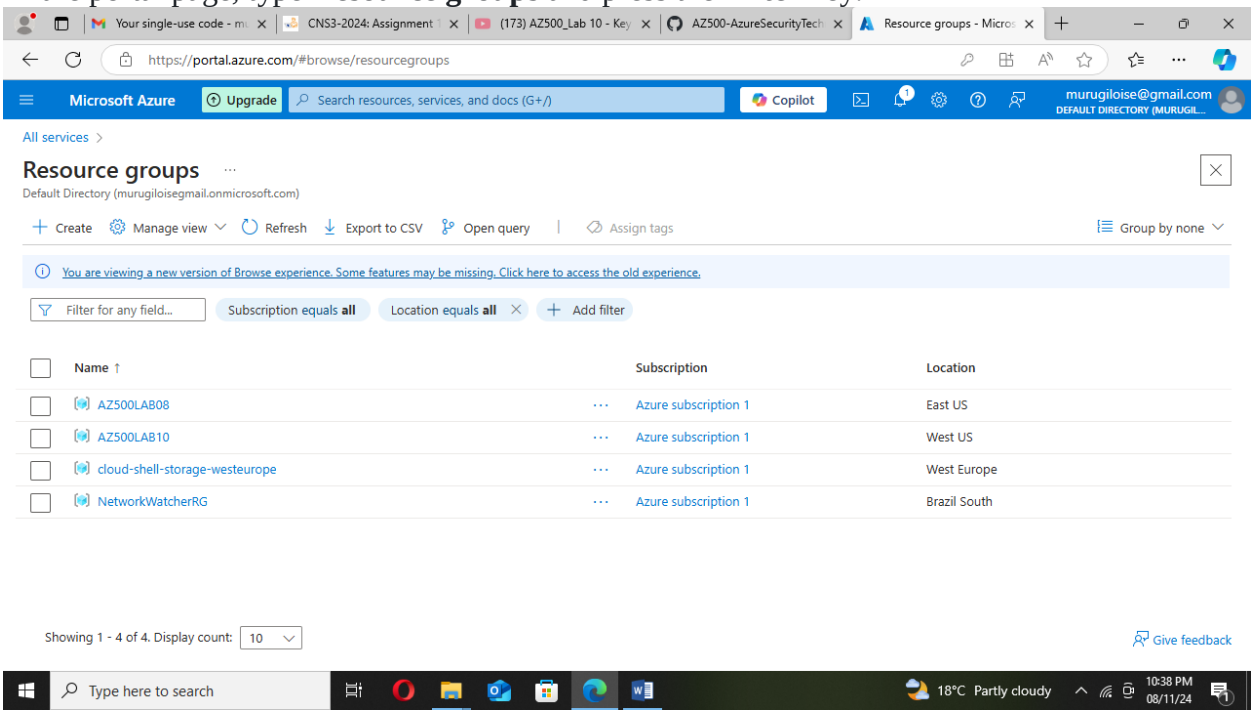
```
$kvName = 'az500kv' + $(Get-Random)

$location = (Get-AzResourceGroup -ResourceGroupName 'AZ500LAB10').Location

New-AzKeyVault -VaultName $kvName -ResourceGroupName 'AZ500LAB10' -Location $location –DisableRbacAuthorization
```

From the screenshot below, our key vault is Vault Name : az500kv1700071371

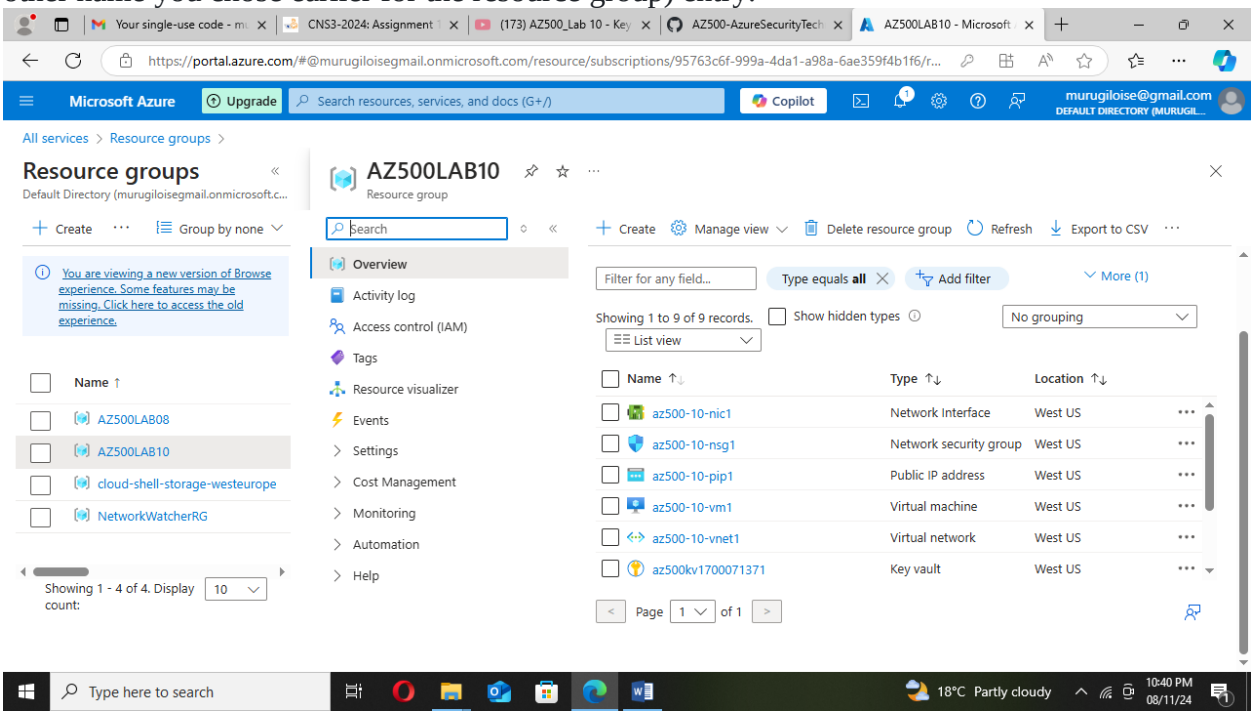
Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)



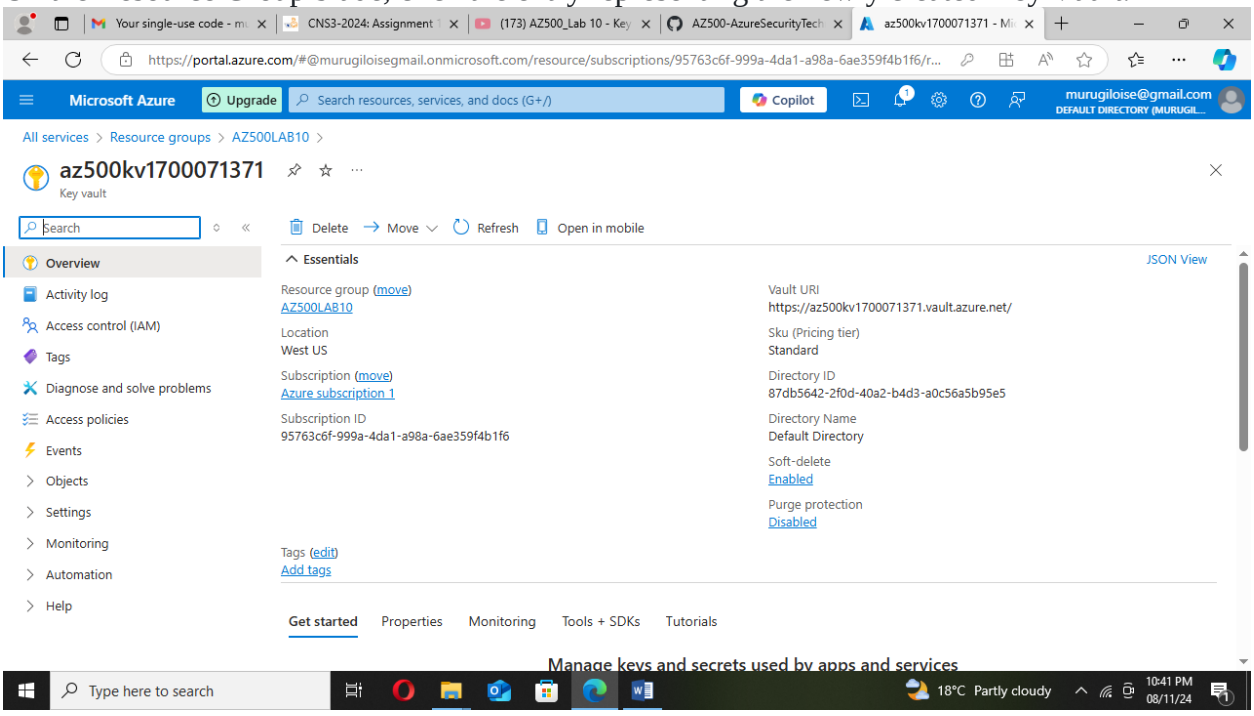
4. Close the Cloud Shell pane. Vault URI : https://az500kv1700071371.vault.azure.net/ name az500kv1700071371
5. In the Azure portal, in the **Search resources, services, and docs** text box at the top of the Azure portal page, type **Resource groups** and press the **Enter** key.



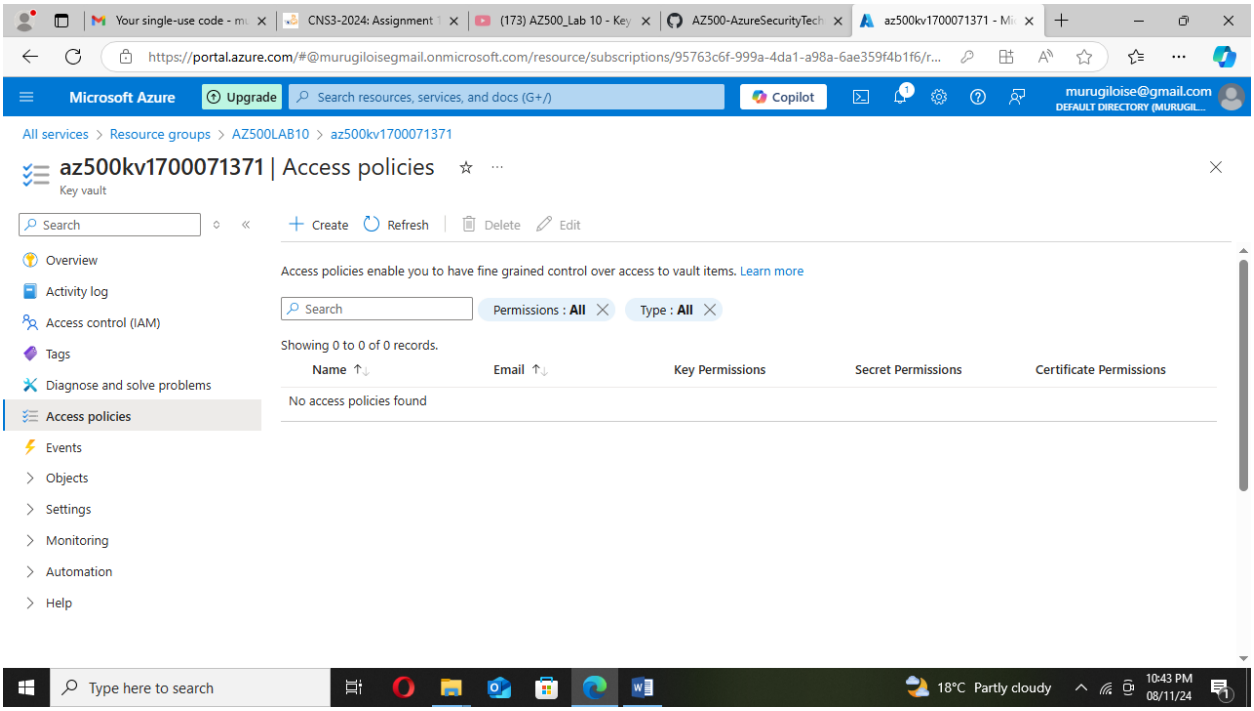
6. On the **Resource groups** blade, in the list of resource group, click the **AZ500LAB10** (or other name you chose earlier for the resource group) entry.

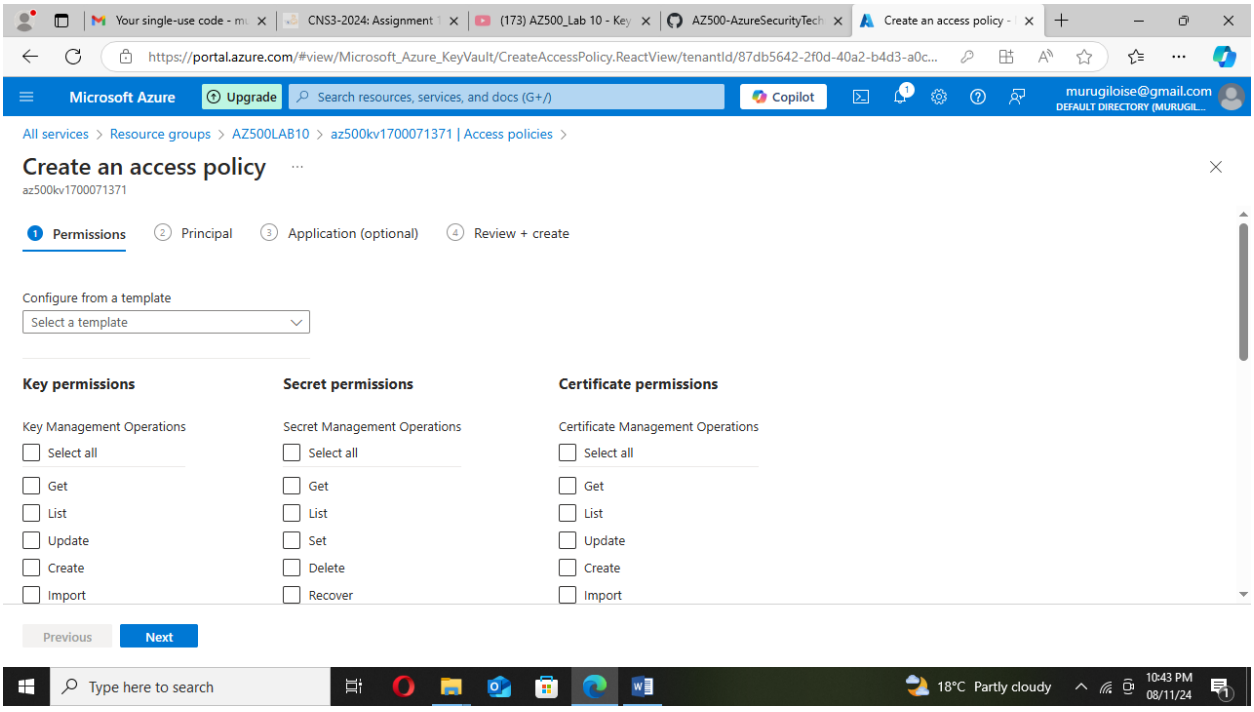


7. On the Resource Group blade, click the entry representing the newly created Key Vault.



8. On the Key Vault blade, in the **Overview** section, click **Access Policies** and then click **+ Create**.

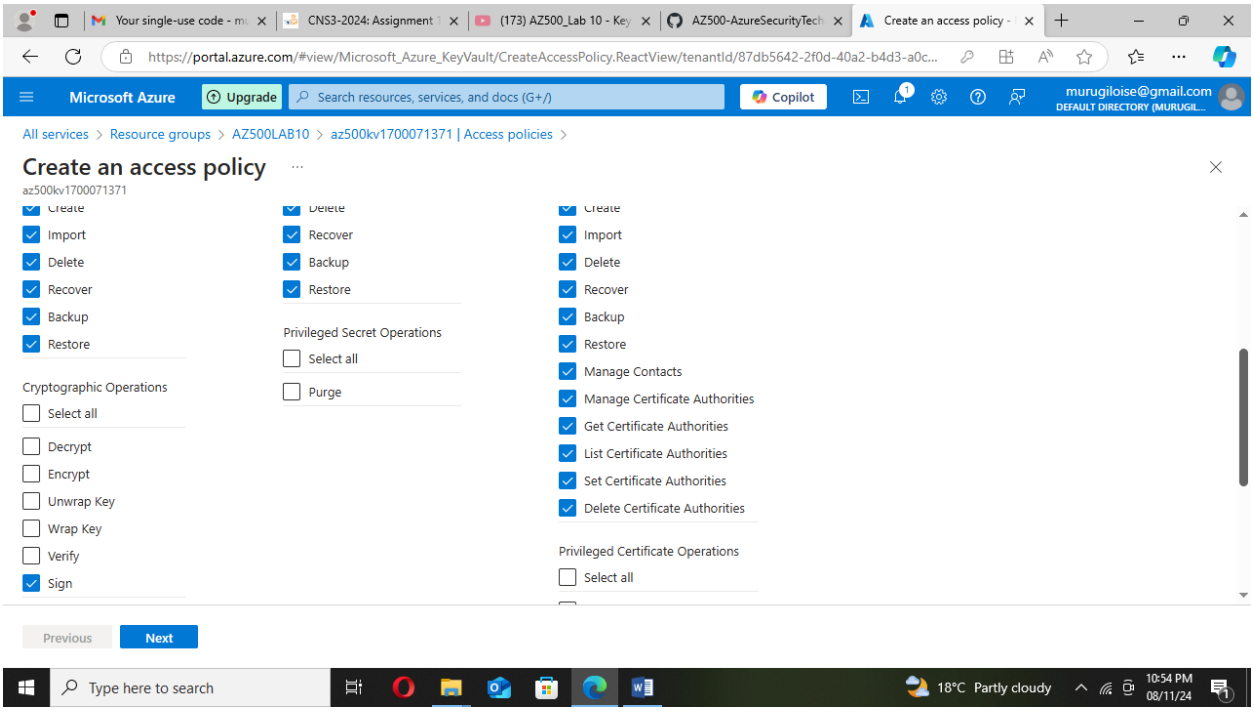




9. On the **Create an access policy** blade, specify the following settings (leave all others with their default values):
Some of the key permissions for the vault are Get, Update, Create, Import, Delete, Backup, Recover, List, Restore.

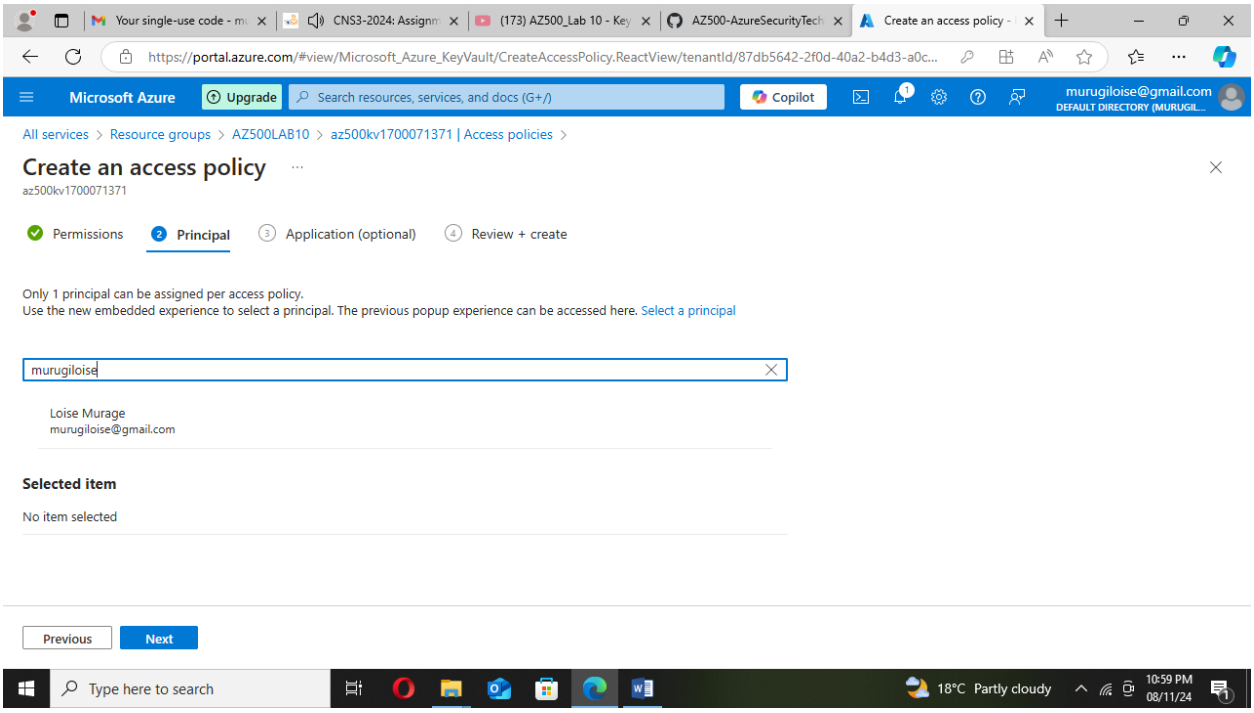
Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)

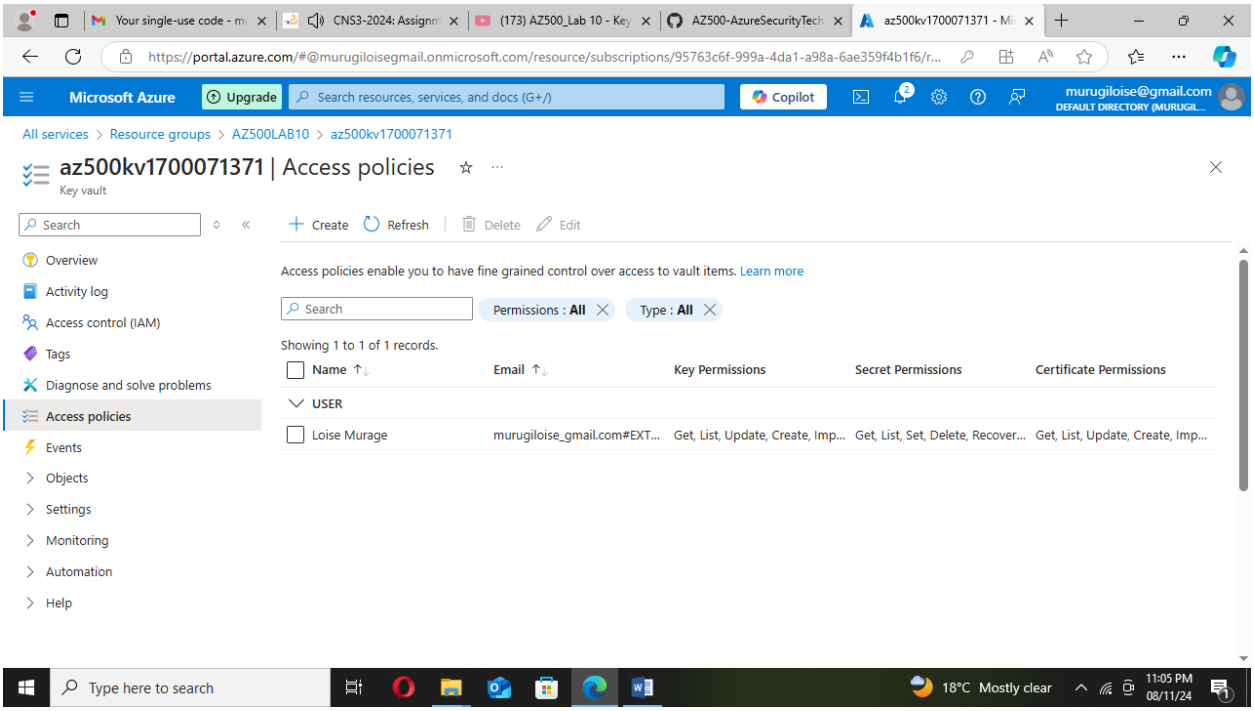
Setting	Value
Configure from template (optional)	**Key, Secret, & Certificate Management**
Key permissions	click **Select all** resulting in total of **9 selected** permissions
Key permissions/Cryptographic Operations	click **Sign** resulting in total of **1 selected** permissions
Secret permissions	click **Select all** resulting in total of **7 selected** permissions
Certification permissions	click **Select all** resulting in total of **15 selected** permissions
Select principal	click **None selected**, on the **Principal** blade, select your user account, and click **Next**
Application (optional)	click **Next**
Review + create	click **Create**



The screenshot below shows my user account as the principal assigned the permissions to.

Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)





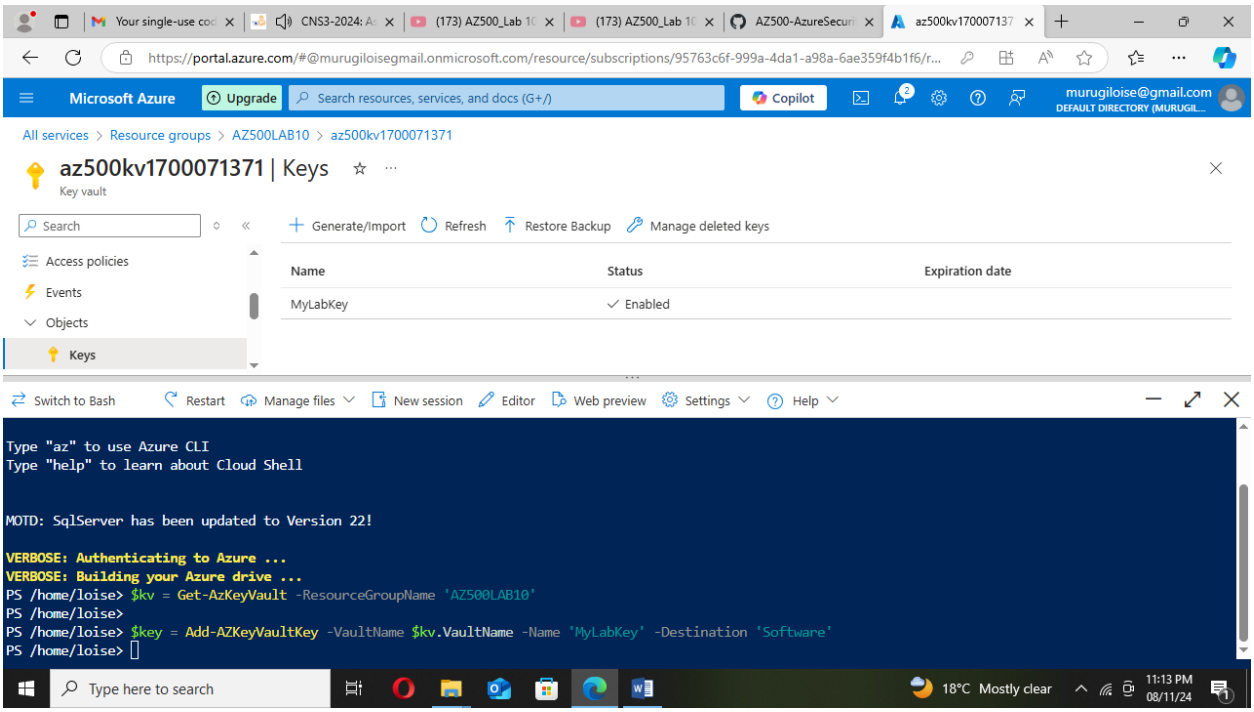
Task 2: Add a key to Key Vault

1. In the Azure portal, open a PowerShell session in the Cloud Shell pane.
2. In this task, you will add a key to the Key Vault and view information about the key. Ensure **PowerShell** is selected in the upper-left drop-down menu of the Cloud Shell pane.

3. In the PowerShell session within the Cloud Shell pane, run the following to add a software-protected key to the Key Vault:

   $kv = Get-AzKeyVault -ResourceGroupName 'AZ500LAB10'

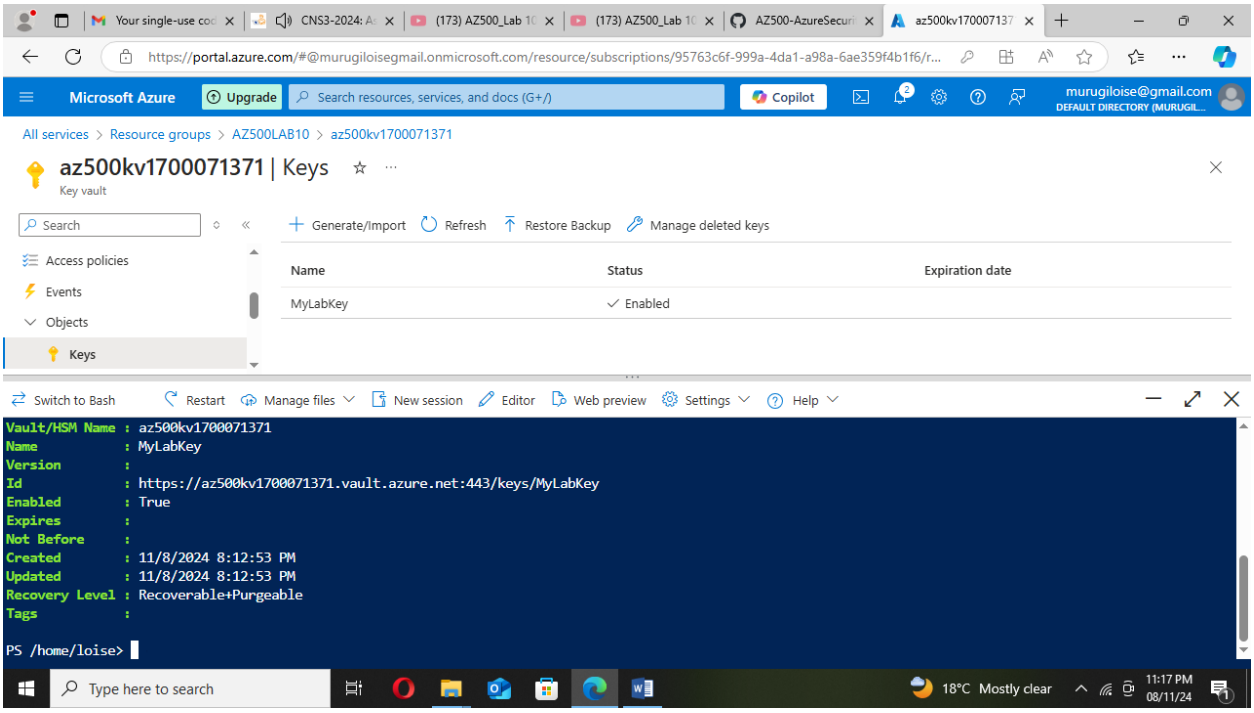
   $key = Add-AZKeyVaultKey -VaultName $kv.VaultName -Name 'MyLabKey' -Destination 'Software'
   Note: The name of the key is **MyLabKey**



4. In the PowerShell session within the Cloud Shell pane, run the following to verify the key was created: Get-AZKeyVaultKey -VaultName $kv.VaultName

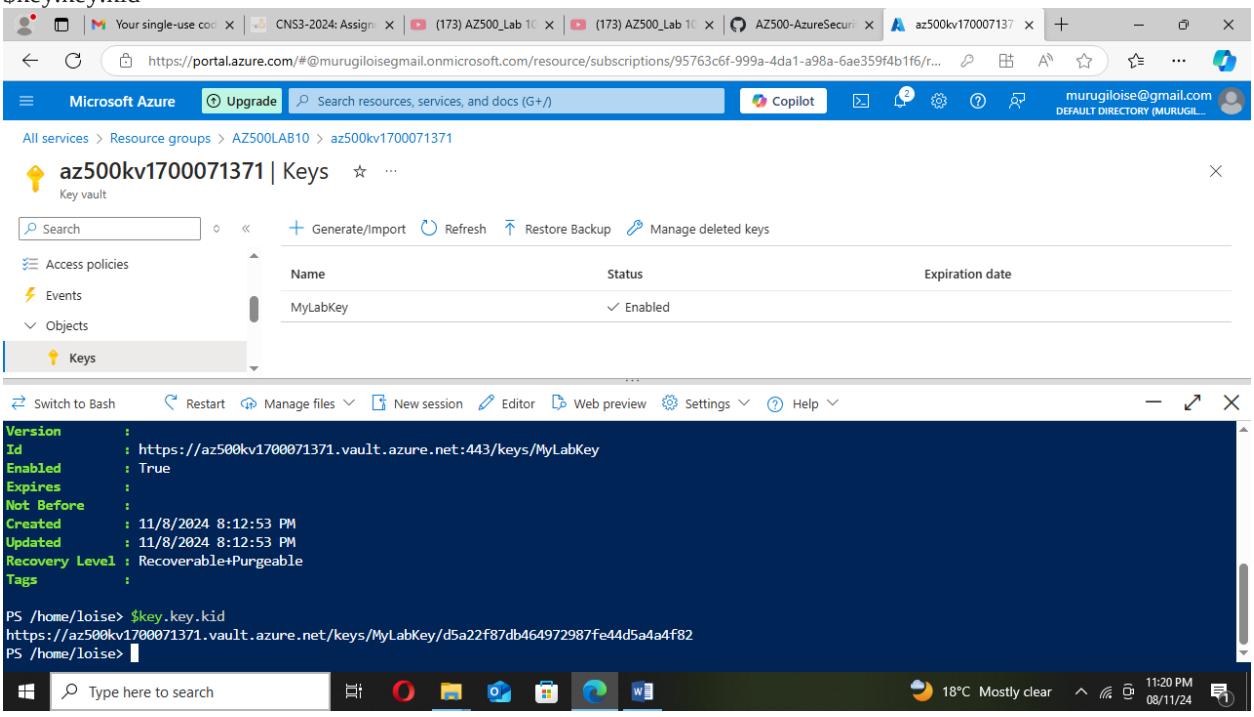
   From the screenshot below MyLabKey was created.

Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)



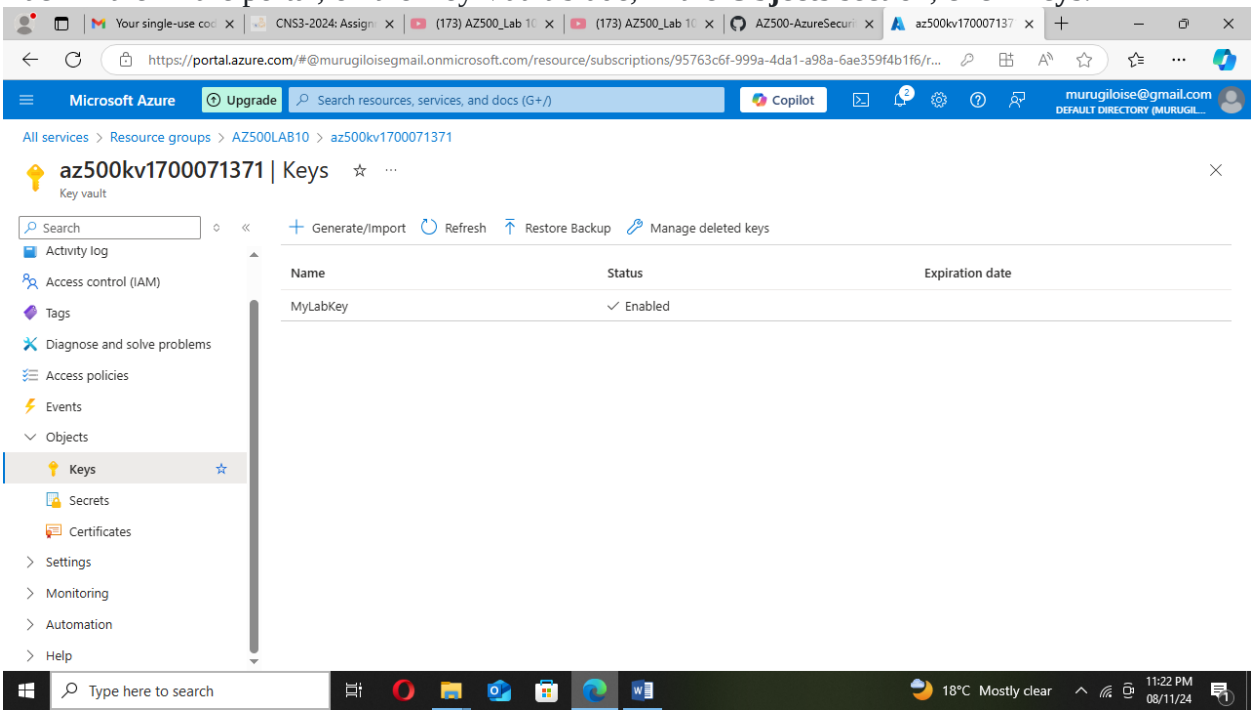
5. In the PowerShell session within the Cloud Shell pane, run the following to display the key identifier:

$key.key.kid



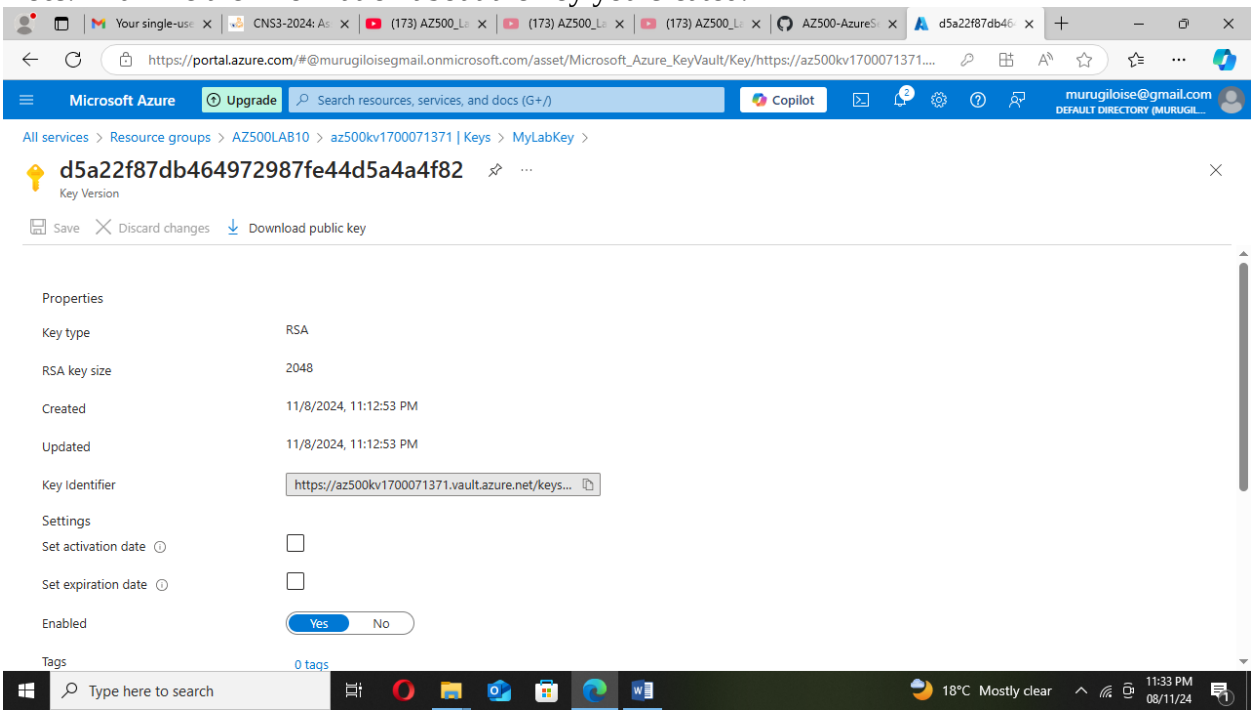
6. Minimize the Cloud Shell pane.

7. Back in the Azure portal, on the Key Vault blade, in the **Objects** section, click **Keys**.



8. In the list of keys, click the **MyLabKey** entry and then, on the **MyLabKey** blade, click the entry representing the current version of the key.

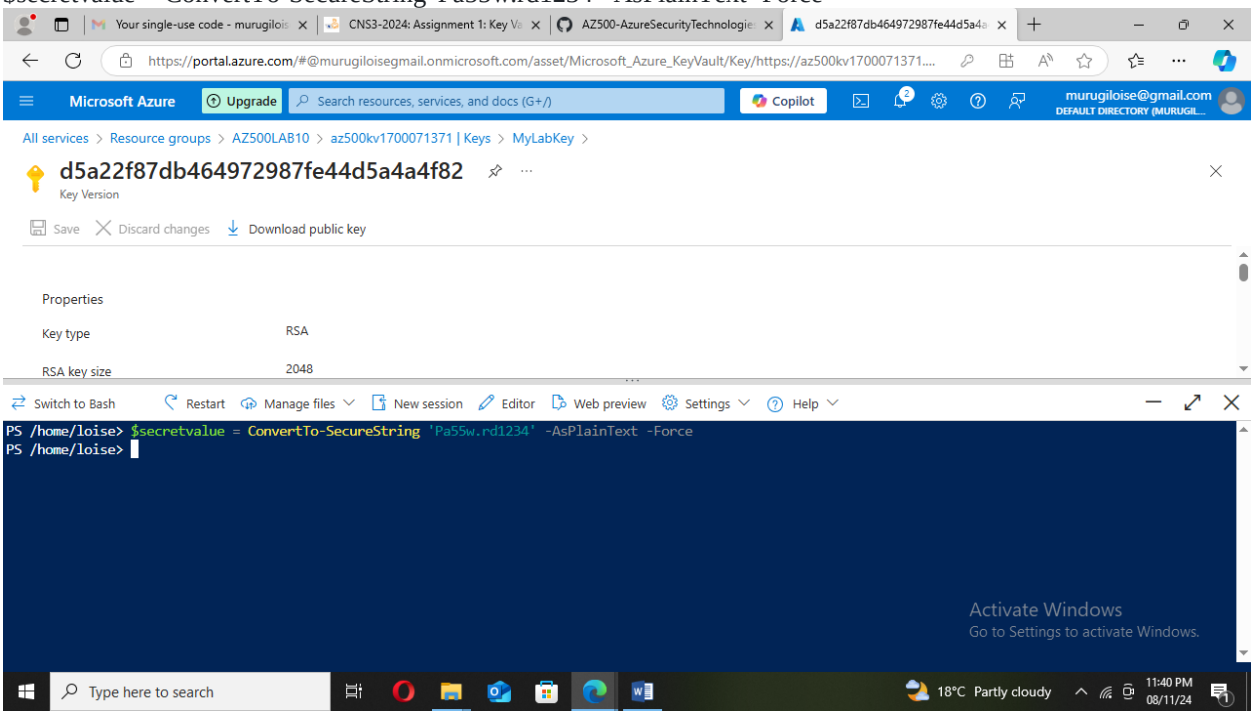
Note: Examine the information about the key you created.



Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)

Task 3: Add a Secret to Key Vault

1. Switch back to the Cloud Shell pane. In the PowerShell session within the Cloud Shell pane, run the following to create a variable with a secure string value:
2. $secretvalue = ConvertTo-SecureString 'Pa55w.rd1234' -AsPlainText -Force



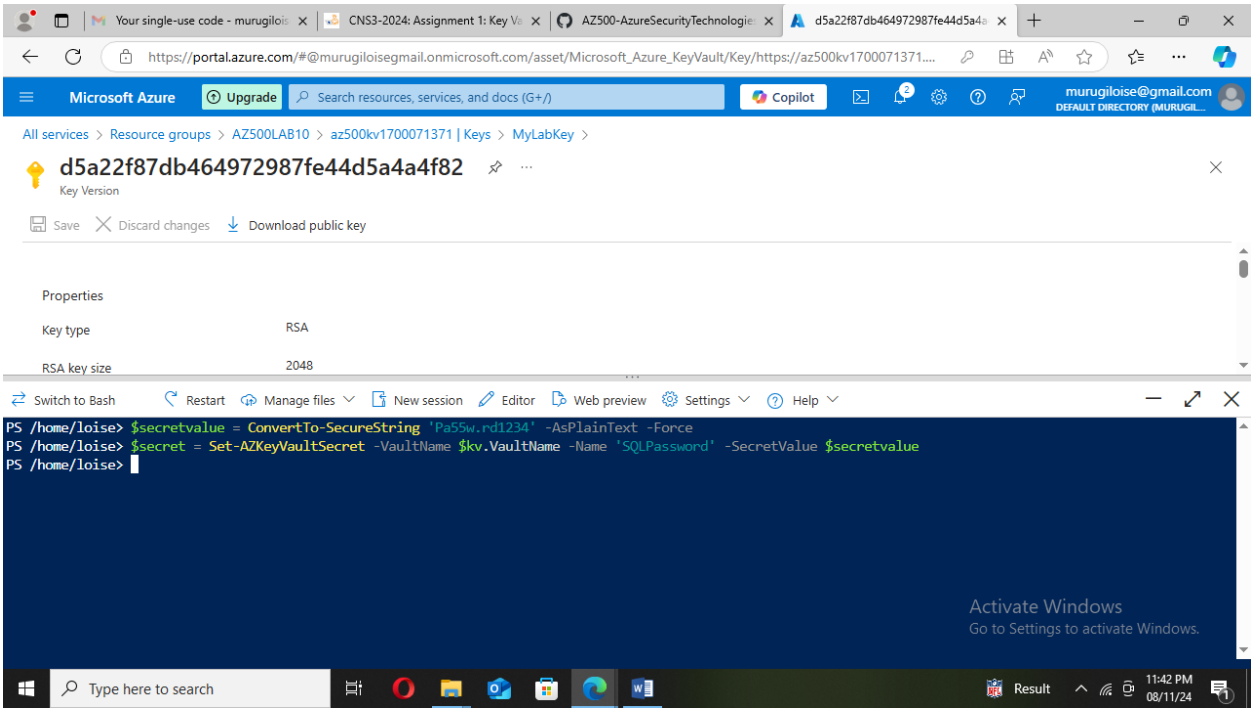
3. In the PowerShell session within the Cloud Shell pane, run the following to add the secret to the vault:

   To generate the value of the Vault name

   $secret = Set-AZKeyVaultSecret -VaultName $kv.VaultName -Name 'SQLPassword' -SecretValue $secretvalue

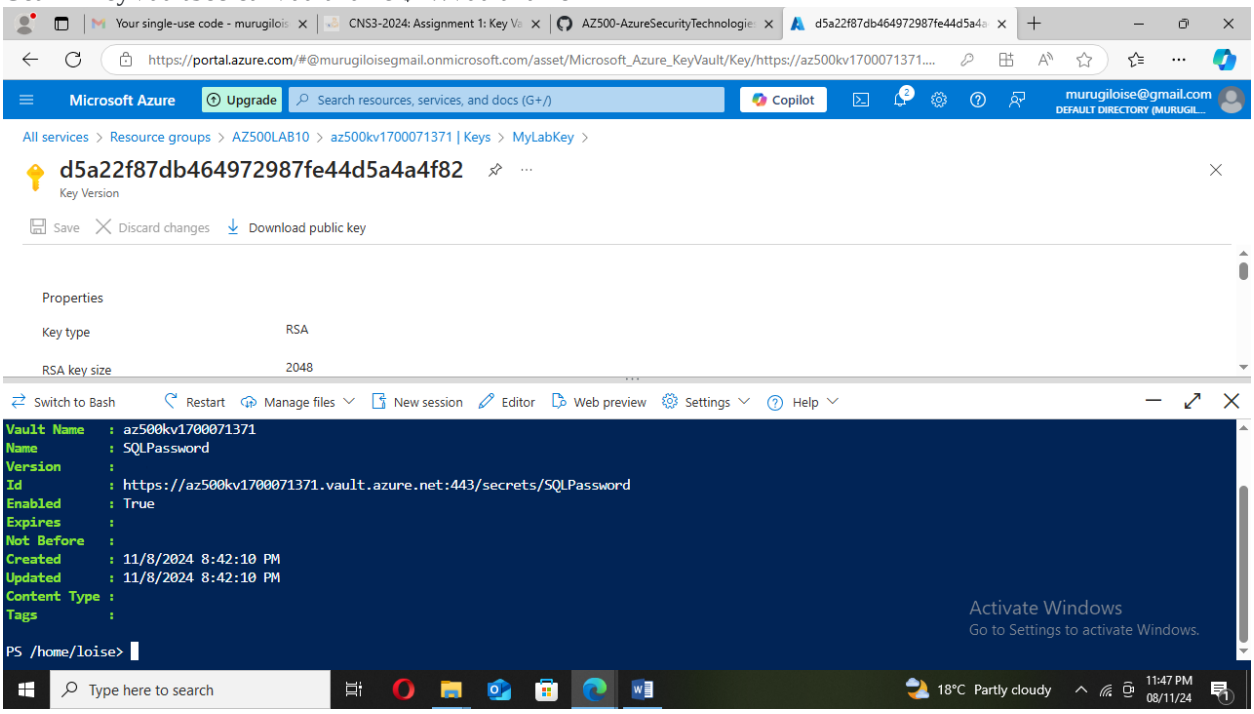
   Note: The name of the secret is SQLPassword.

Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)



4. In the PowerShell session within the Cloud Shell pane, run the following to verify the secret was created.

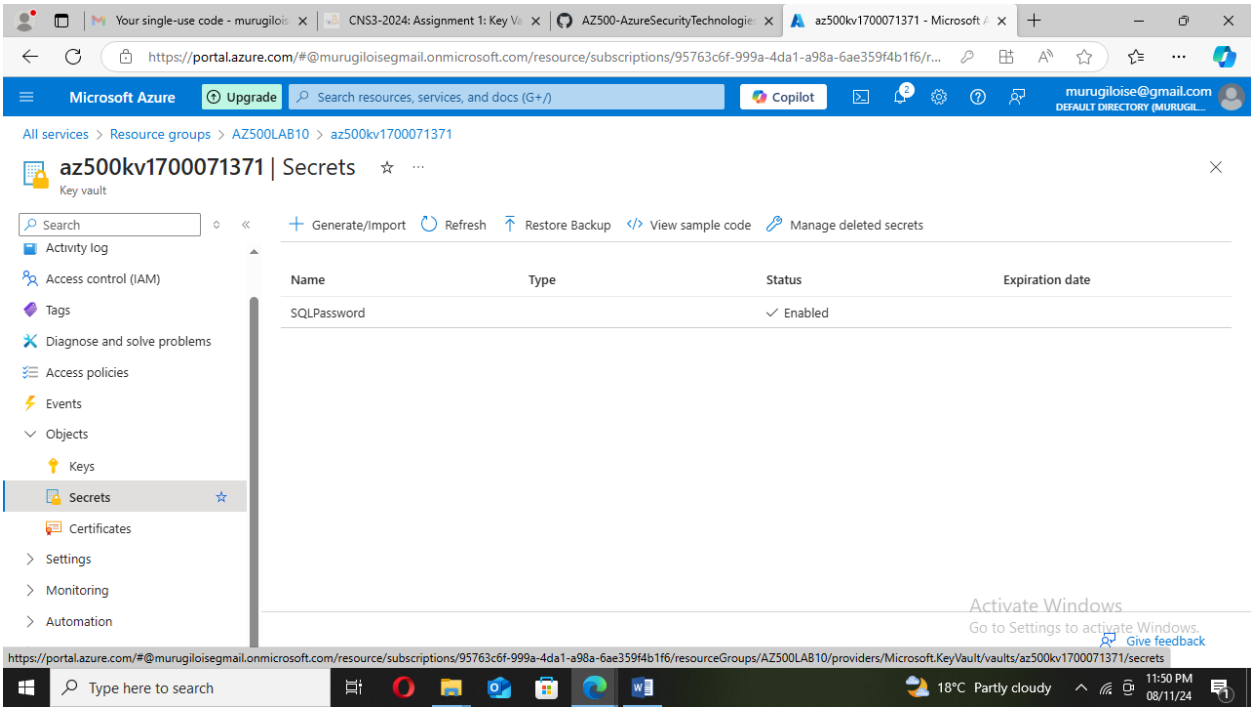
1. Get-AZKeyVaultSecret -VaultName $kv.VaultName



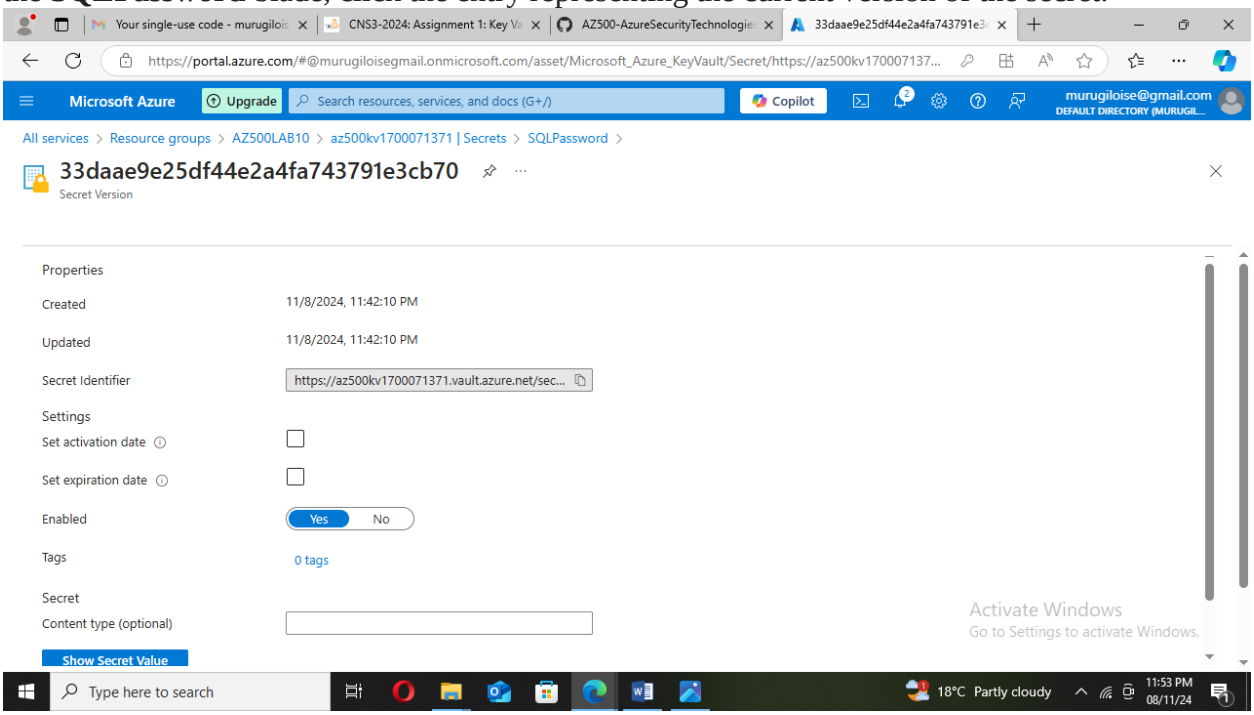
Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)

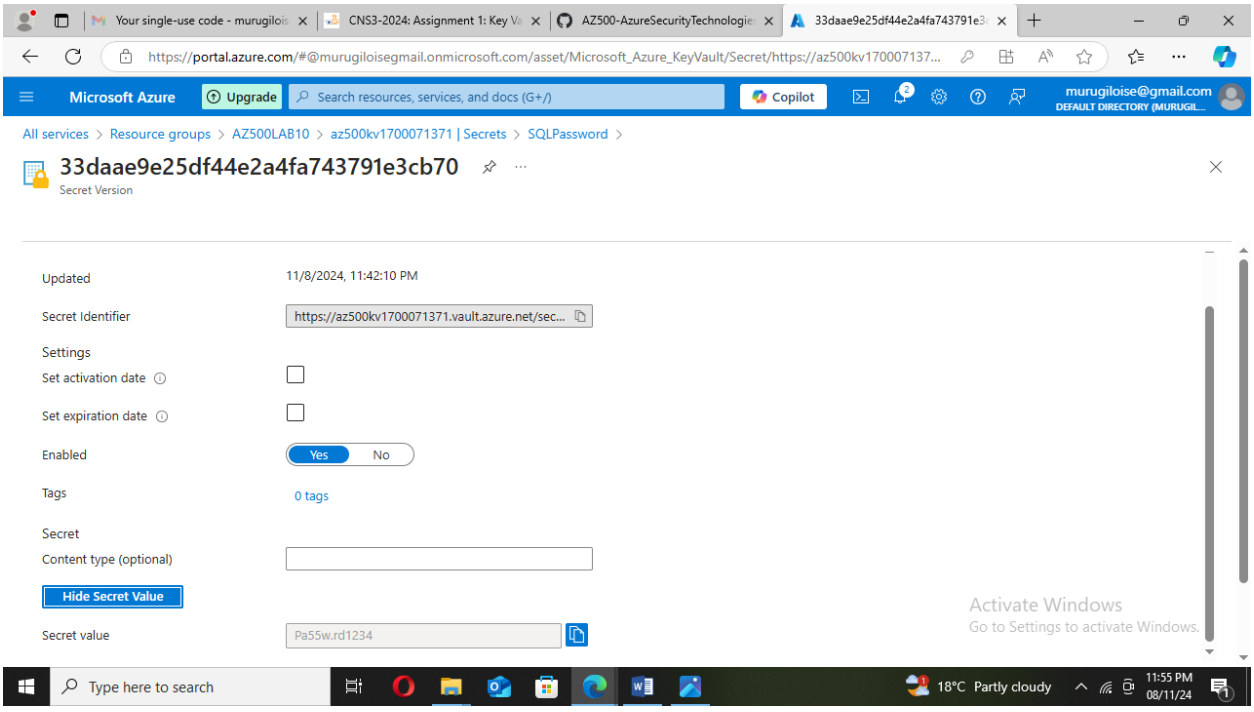
5. Minimize the Cloud Shell pane.
6. In the Azure portal, navigate back to the Key Vault blade, in the **Objects** section, click **Secrets**.

A secret has been created SQLPassword



7. In the list of secrets, click the **SQLPassword** entry and then, on the **SQLPassword** blade, click the entry representing the current version of the secret.





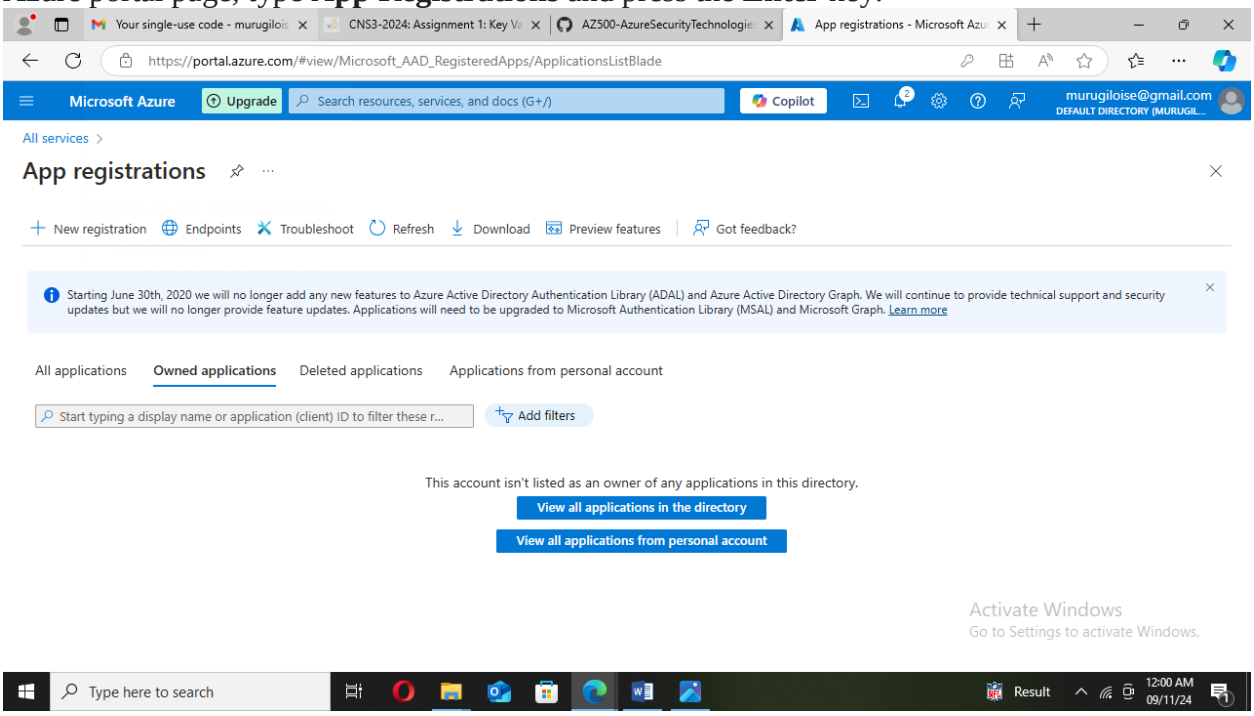
Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)

Exercise 3: Configure an Azure SQL database and a data-driven application

Task 1: Enable a client application to access the Azure SQL Database service.

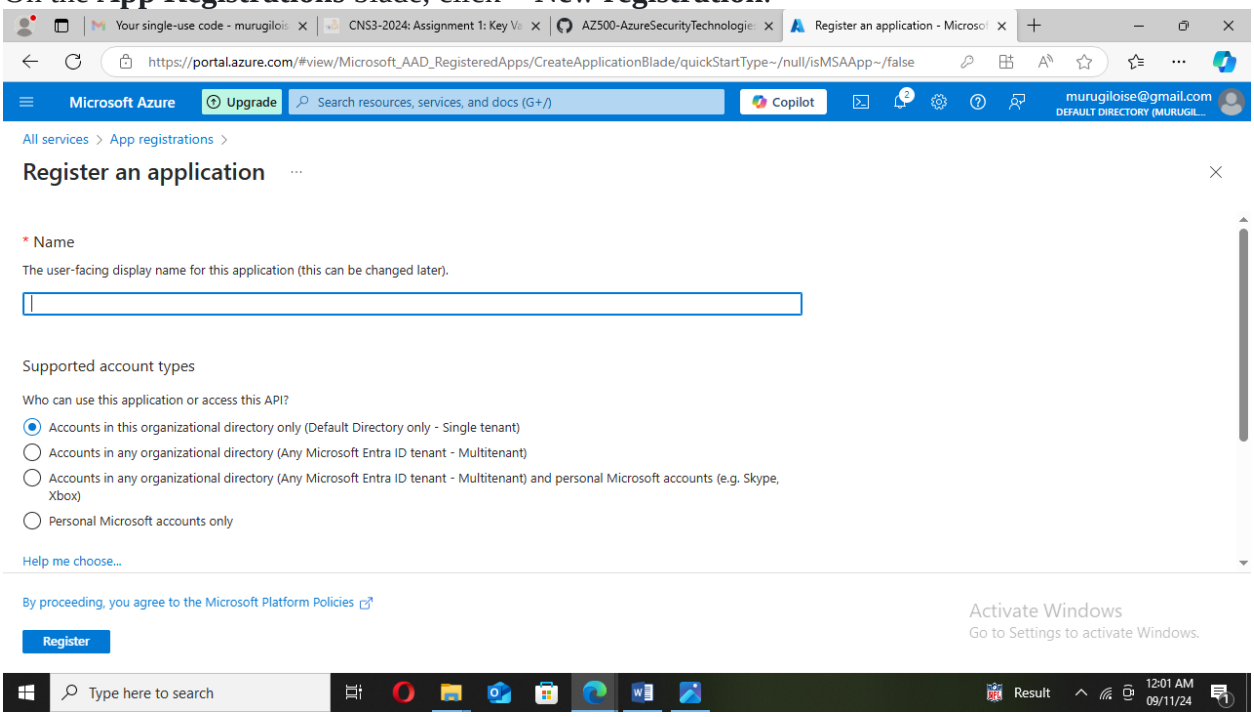
In this task, you will enable a client application to access the Azure SQL Database service. This will be done by setting up the required authentication and acquiring the Application ID and Secret that you will need to authenticate your application.

1. In the Azure portal, in the **Search resources, services, and docs** text box at the top of the Azure portal page, type **App Registrations** and press the **Enter** key.



Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)

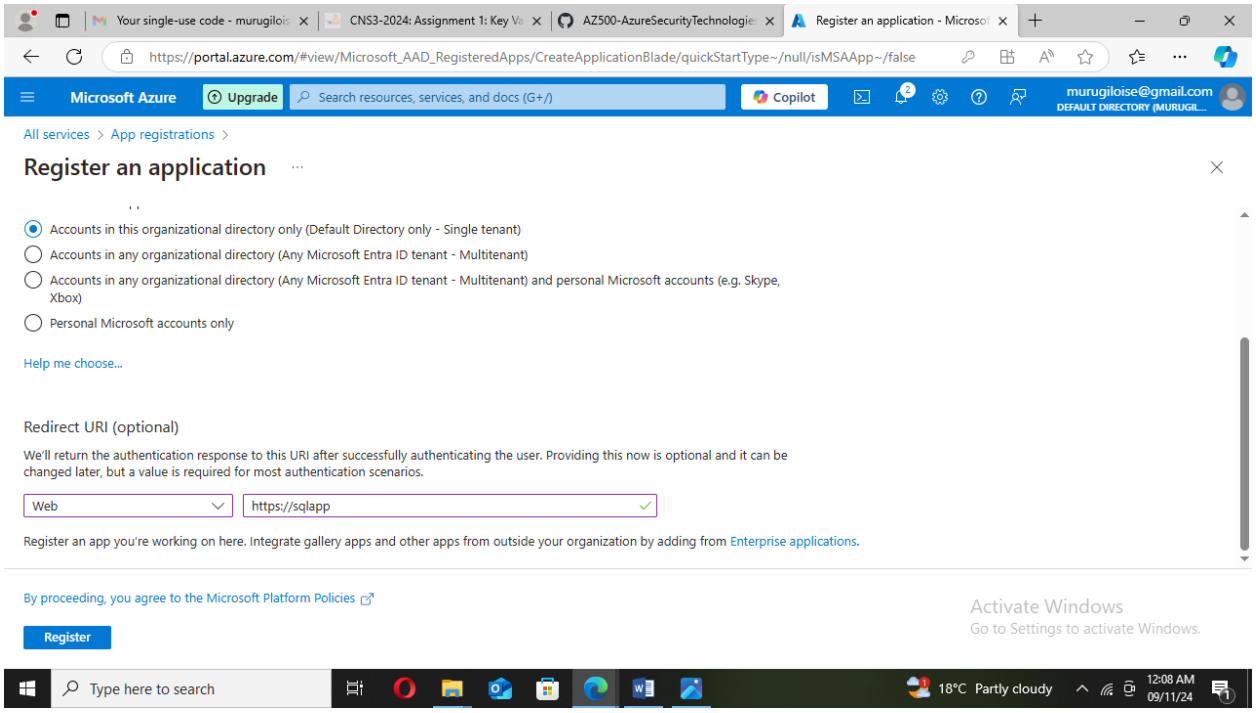
2. On the **App Registrations** blade, click **+ New registration**.



3. On the **Register an application** blade, specify the following settings (leave all others with their default values):

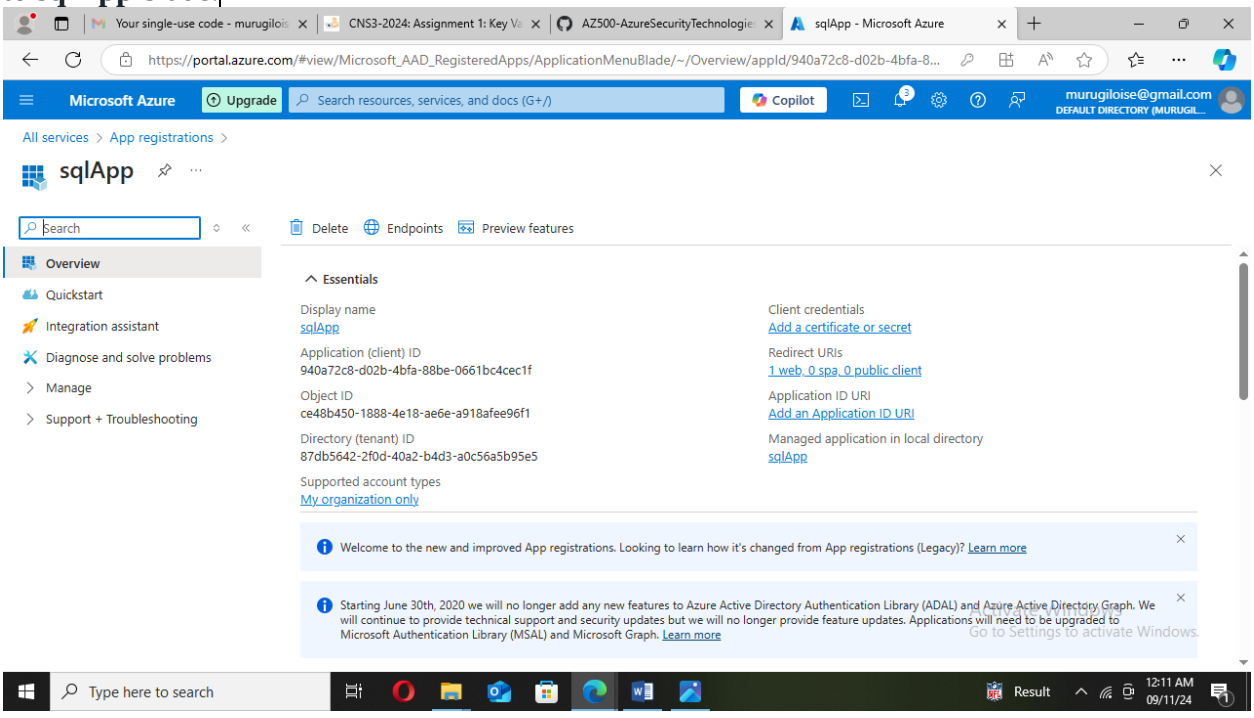
Setting	**Value**
Name	**sqlApp**
Redirect URI (optional)	**Web** and **https://sqlapp**

Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)



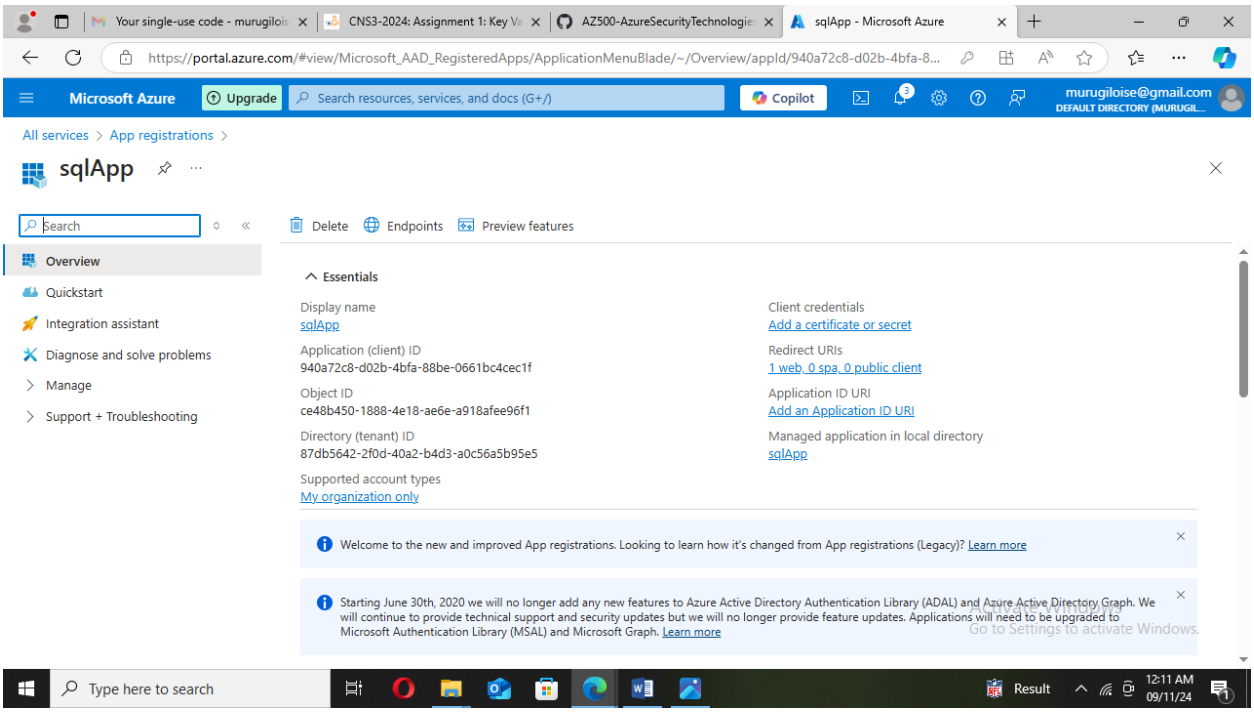
4. On the **Register an application** blade, click **Register**.

   Note: Once the registration is completed, the browser will automatically redirect you to **sqlApp** blade.

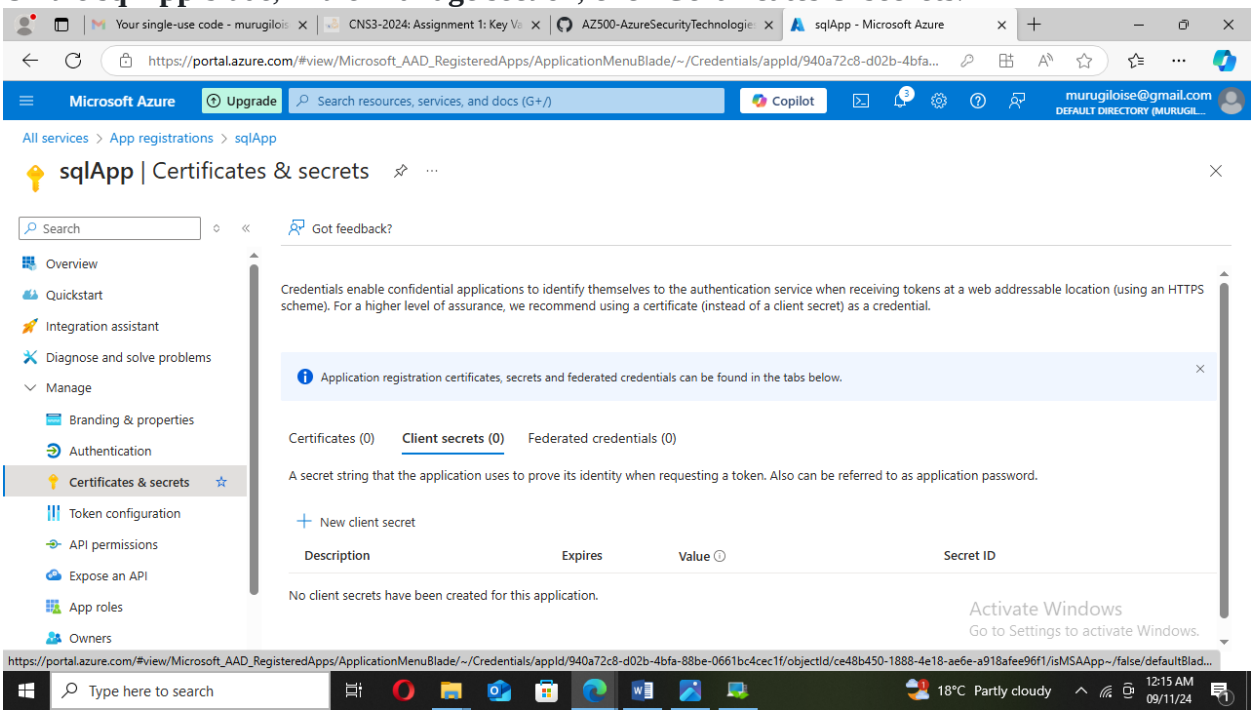


5. On the **sqlApp** blade, identify the value of **Application (client) ID**. The application app ID is 940a72c8-d02b-4bfa-88be-0661bc4cec1f

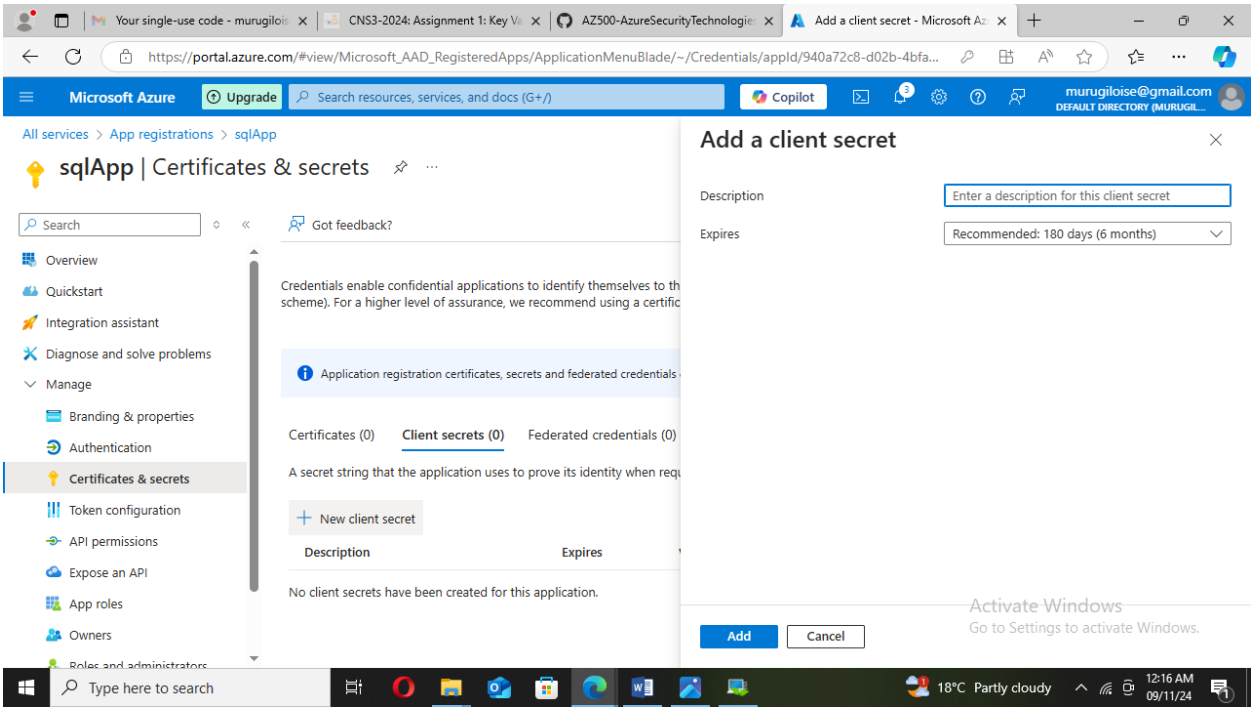
Note: Record this value. You will need it in the next task.



6. On the **sqlApp** blade, in the **Manage** section, click **Certificates & secrets**.



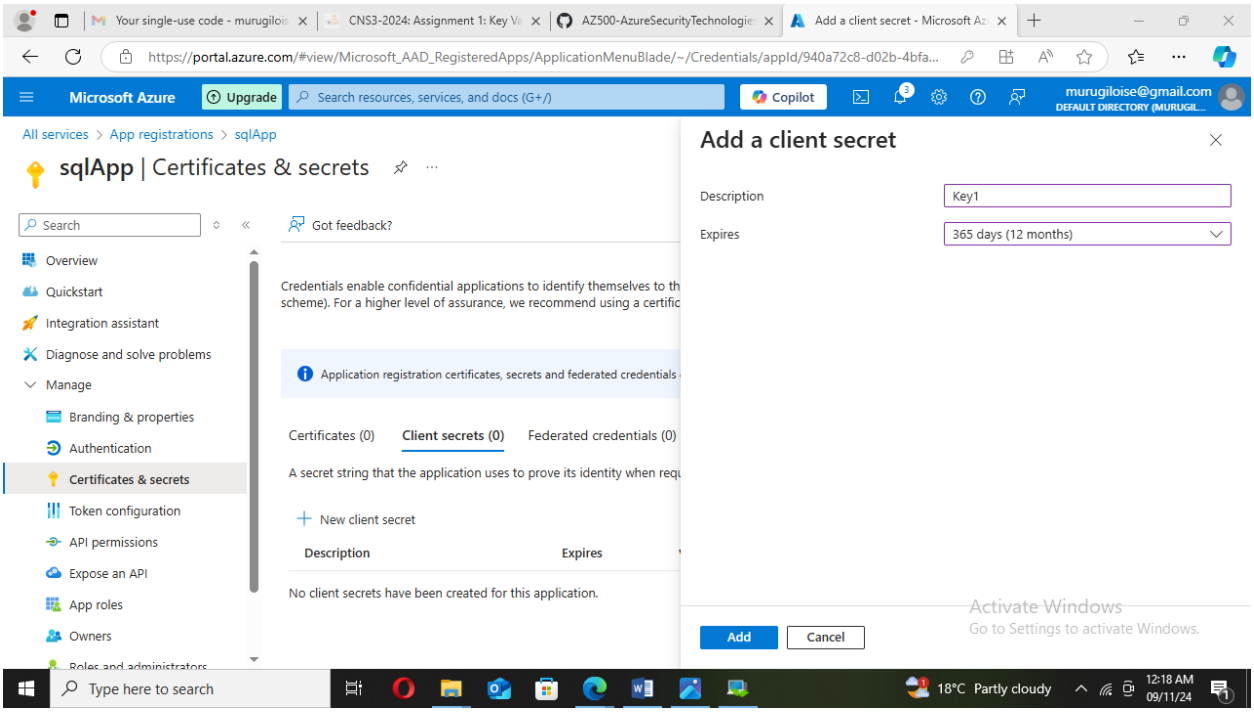
7. On the **sqlApp | Certificates & secrets** blade / **Client Secrets** section, click **+ New client secret**



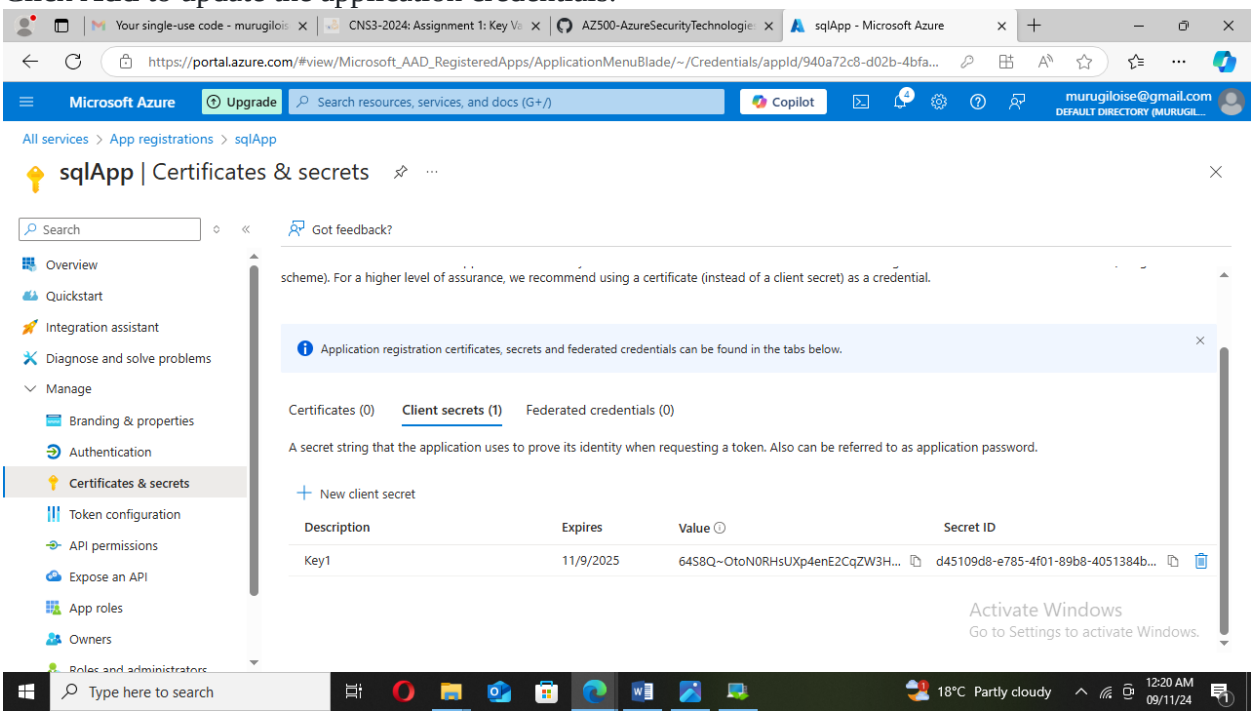
8. In the **Add a client secret** pane, specify the following settings:

Setting	**Value**
Description	**Key1**
Expires	**12months**

Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)



9. Click **Add** to update the application credentials.



10. On the **sqlApp | Certificates & secrets** blade, identify the value of **Key1**.
Value of Key1 is 64S8Q~OtoN0RHsUXp4enE2CqZW3H7dNa0~C5Ab90
Secret ID is d45109d8-e785-4f01-89b8-4051384b9168

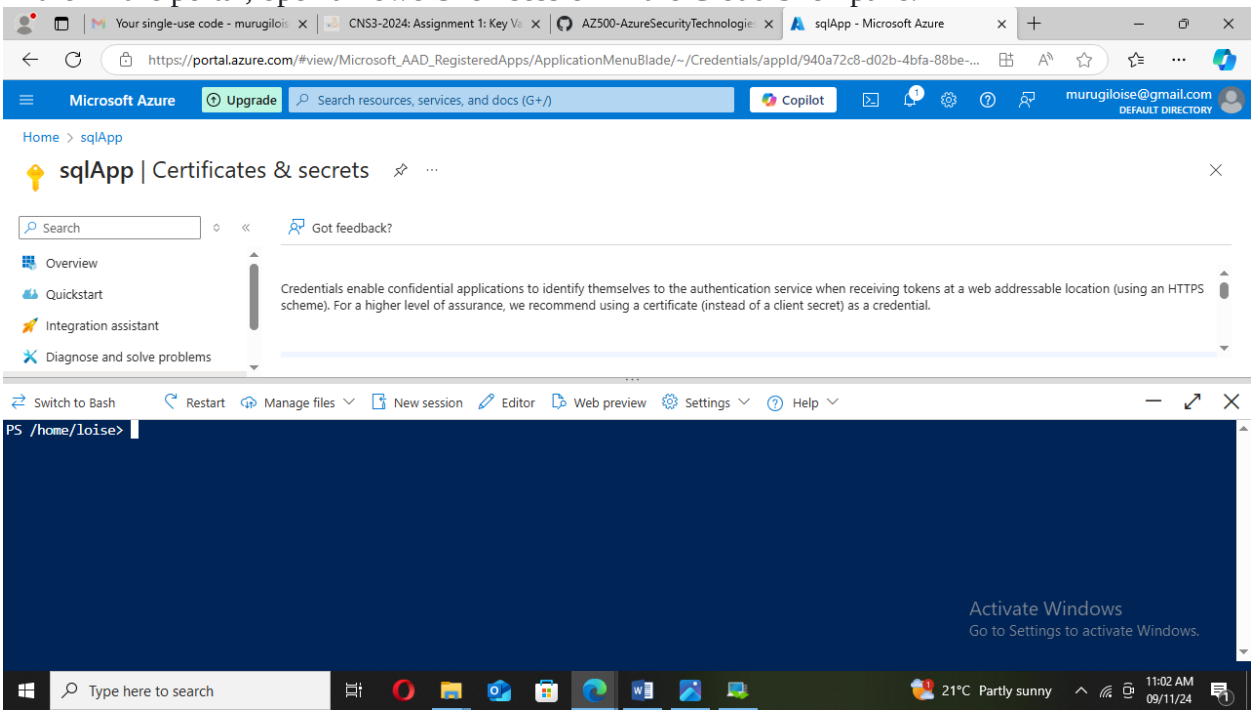
Our Application ID: 940a72c8-d02b-4bfa-88be-0661bc4cec1f
Object ID: ce48b450-1888-4e18-ae6e-a918afee96f1

Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)

Task 2: Create a policy allowing the application access to the Key Vault.

In this task, you will grant the newly registered app permissions to access secrets stored in the Key Vault.

1. In the Azure portal, open a PowerShell session in the Cloud Shell pane.

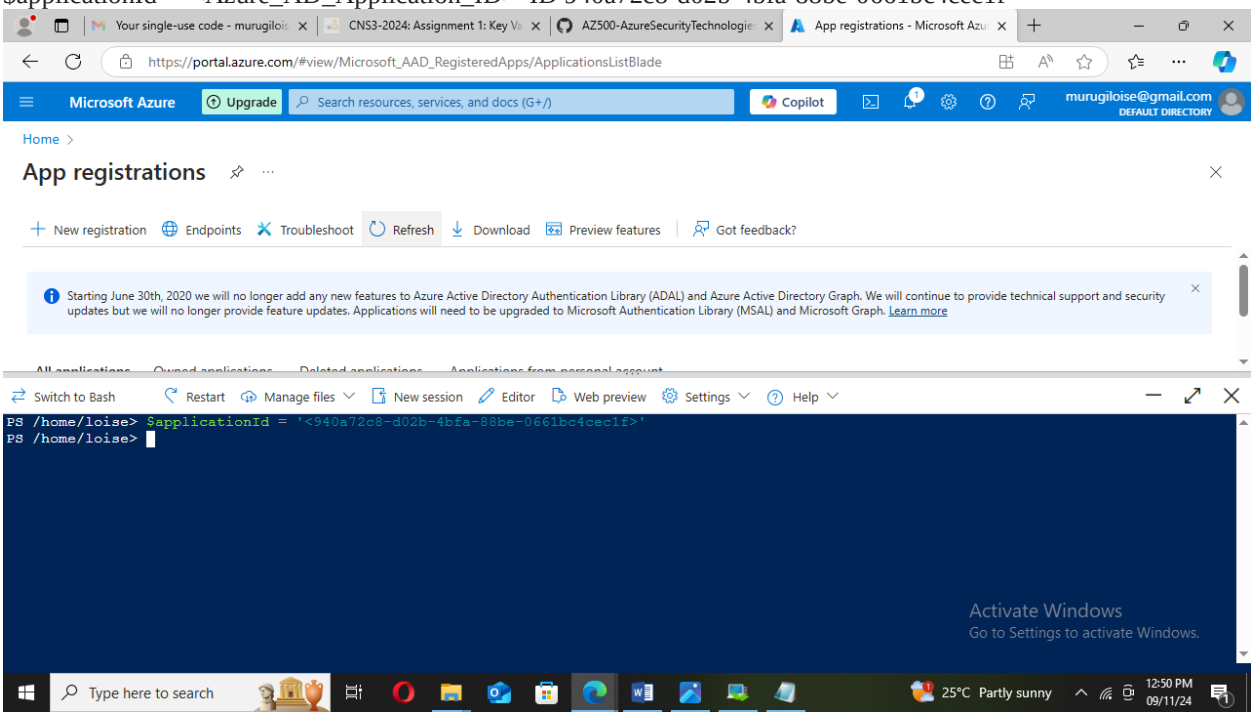


2. Ensure **PowerShell** is selected in the upper-left drop-down menu of the Cloud Shell pane.

3. In the PowerShell session within the Cloud Shell pane, run the following to create a variable storing the **Application (client) ID** you recorded in the previous task (replace the <Azure_AD_Application_ID> placeholder with the value of the **Application (client) ID**):

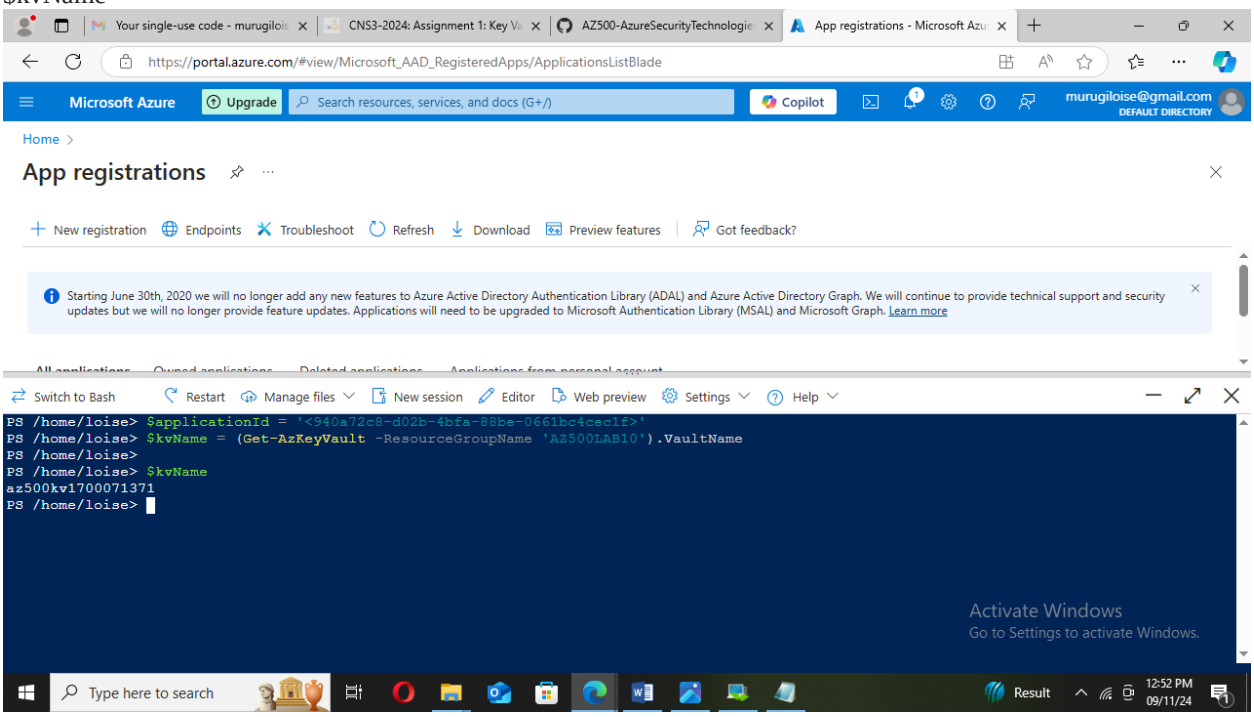
Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)

$applicationId = '<Azure_AD_Application_ID>' ID 940a72c8-d02b-4bfa-88be-0661bc4cec1f

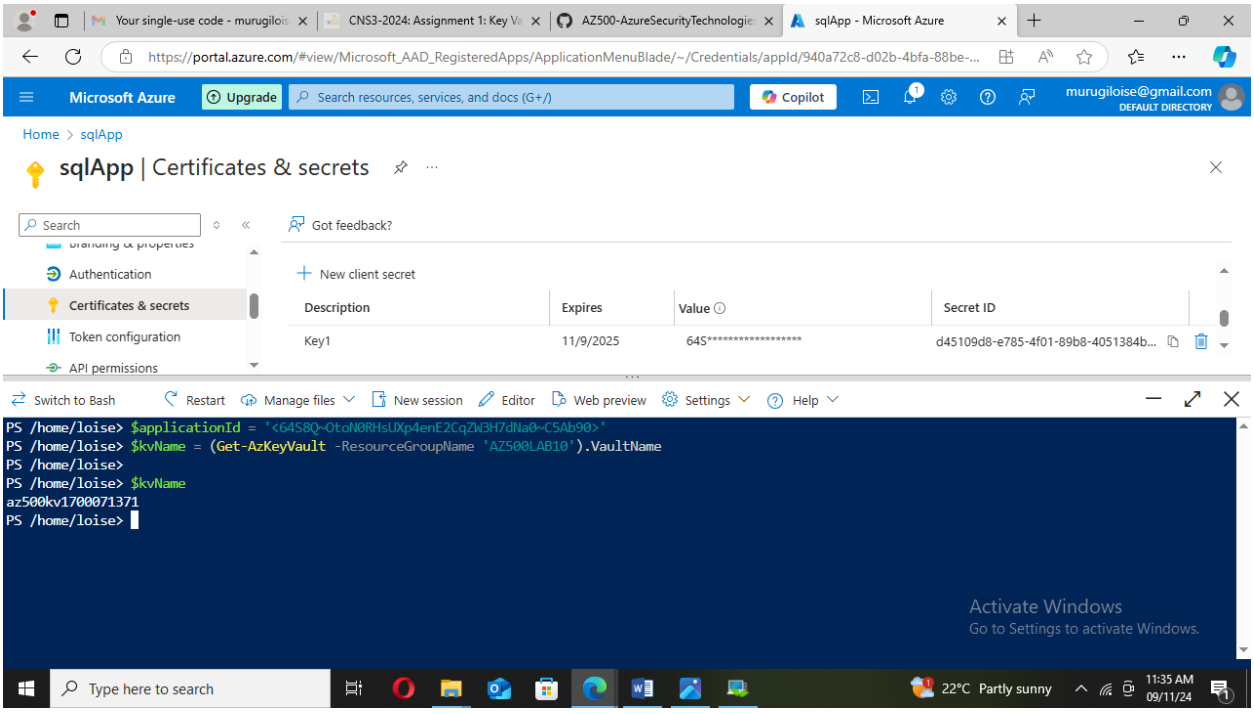


4. In the PowerShell session within the Cloud Shell pane, run the following to create a variable storing the Key Vault name.
$kvName = (Get-AzKeyVault -ResourceGroupName 'AZ500LAB10').VaultName

$kvName

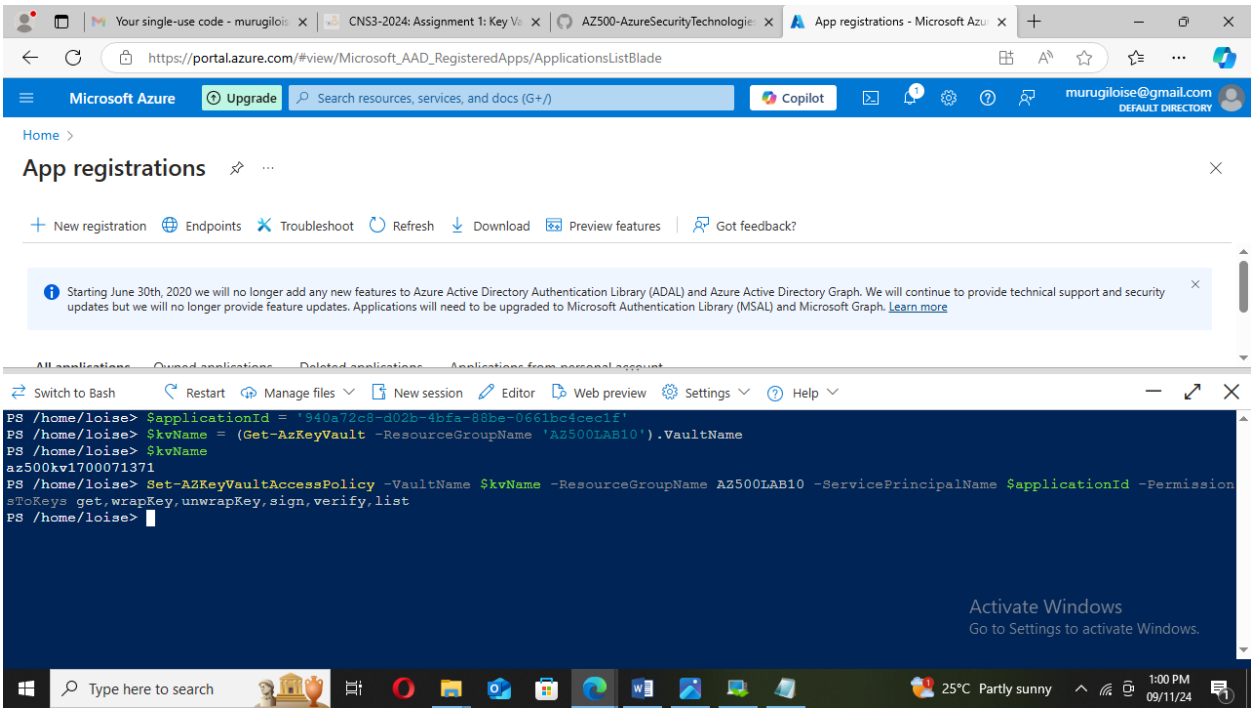


Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)



5. In the PowerShell session within the Cloud Shell pane, run the following to grant permissions on the Key Vault to the application you registered in the previous task:

Set-AZKeyVaultAccessPolicy -VaultName $kvName -ResourceGroupName AZ500LAB10 -ServicePrincipalName $applicationId -PermissionsToKeys get,wrapKey,unwrapKey,sign,verify,list



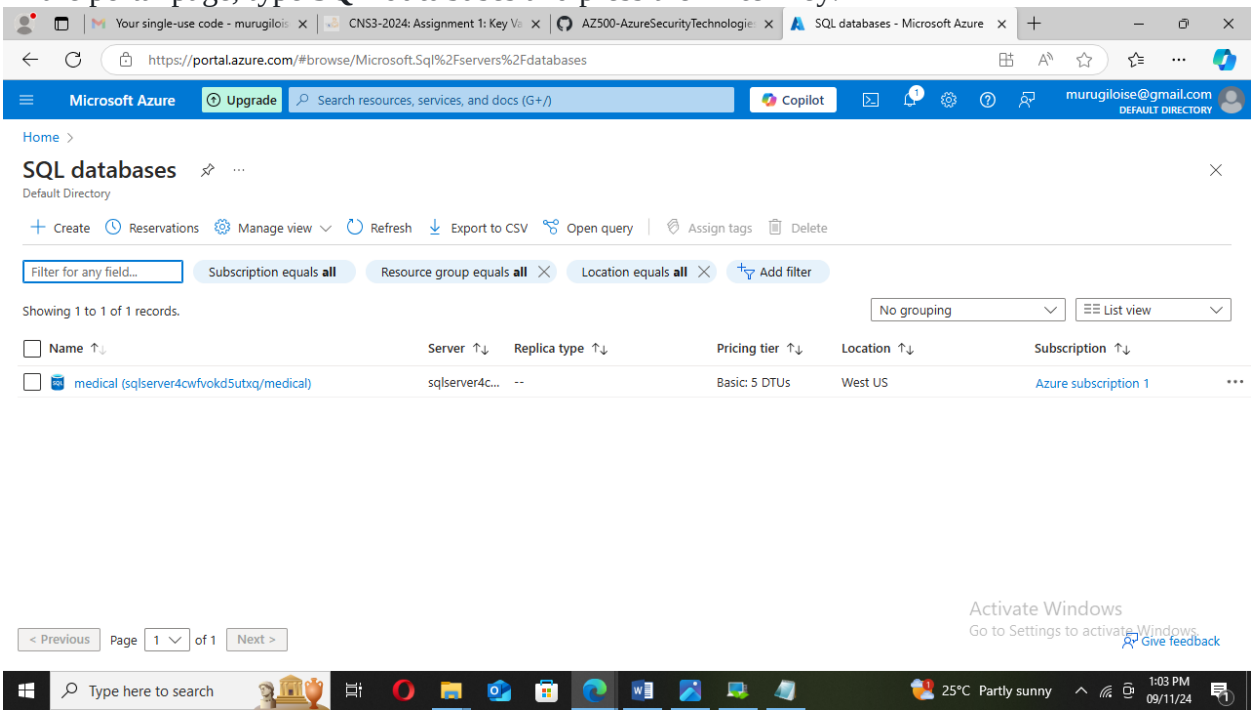
6. Close the Cloud Shell pane.

Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)

Task 3: Retrieve SQL Azure database ADO.NET Connection String

The ARM-template deployment in Exercise 1 provisioned an Azure SQL Server instance and an Azure SQL database named **medical** . You will update the empty database resource with a new table structure and select data columns for encryption

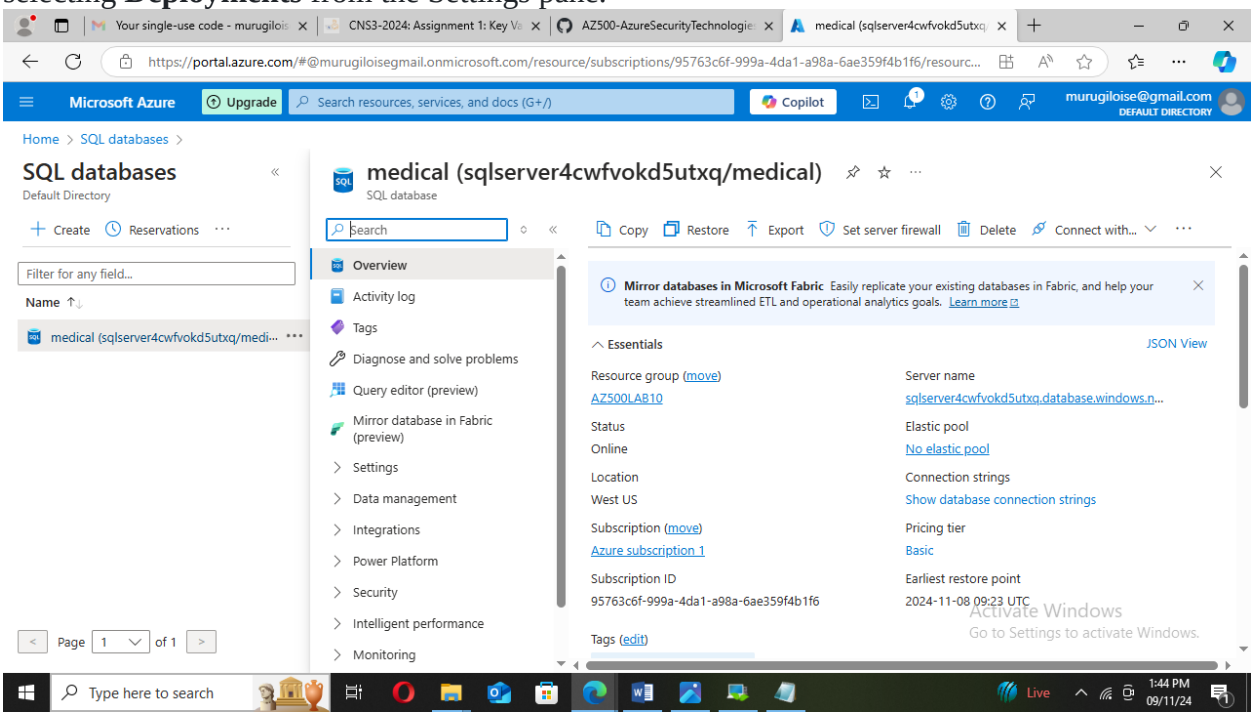
1. In the Azure portal, in the **Search resources, services, and docs** text box at the top of the Azure portal page, type **SQL databases** and press the **Enter** key.



2. In the list of SQL databases, click the **medical()** entry.

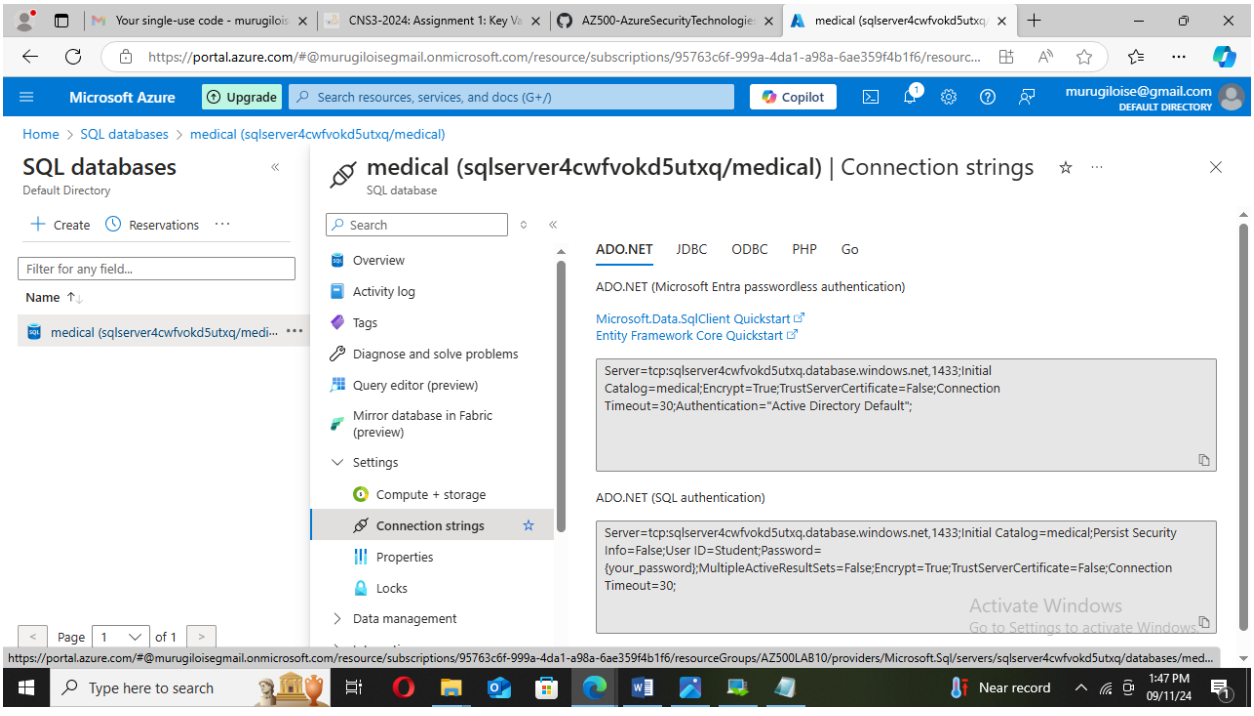
   Note: If the database cannot be found, this likely means the deployment you initiated in Exercise 1 has not completed yet. You can validate this by browsing to the Azure Resource Group "AZ500LAB10" (or the name you chose), and

selecting **Deployments** from the Settings pane.



3. On the SQL database blade, in the **Settings** section, click **Connection strings**.

   Note: The interface includes connection strings for ADO.NET, JDBC, ODBC, PHP, and Go.

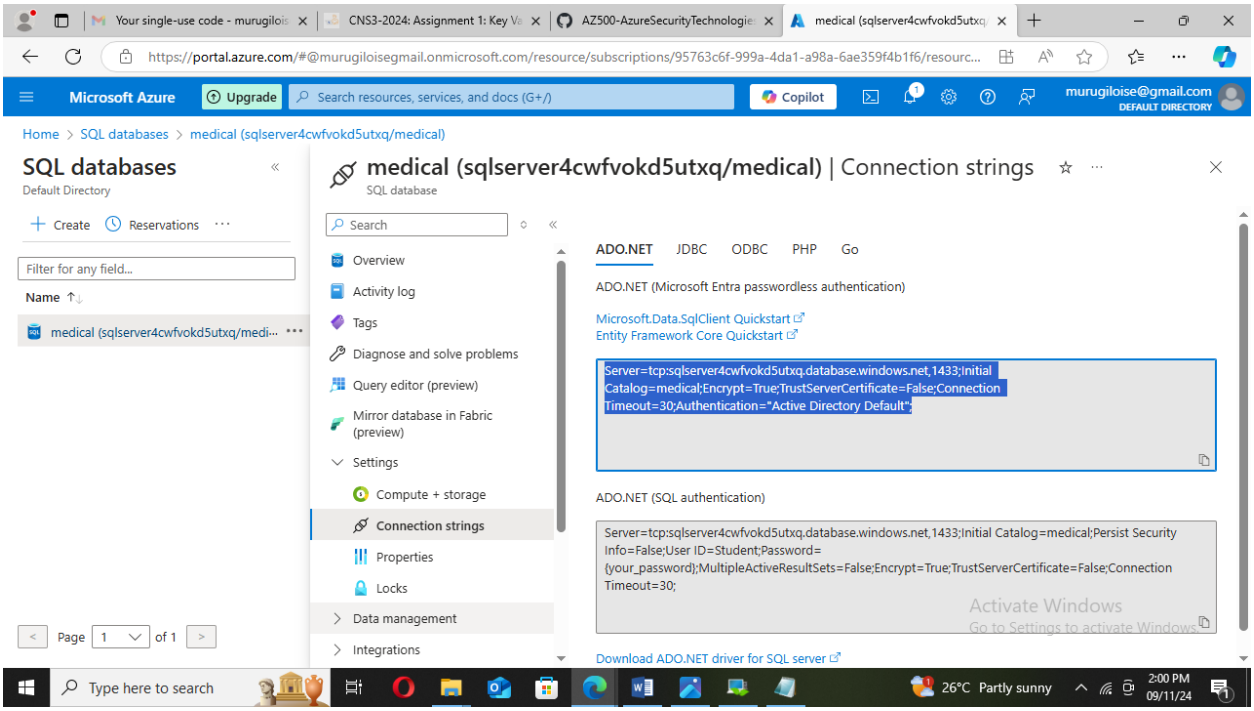


4. Record the **ADO.NET (SQL authentication)** connection string. You will need it later.

Note: When you use the connection string, make sure to replace the {your_password} placeholder with the password that you configured with the deployment in Exercise 1.

ADO.NET (SQL Authenticaytion)

Server=tcp:sqlserver4cwfvokd5utxq.database.windows.net,1433;Initial Catalog=medical;Encrypt=True;TrustServerCertificate=False;Connection Timeout=30;Authentication="Active Directory Default";

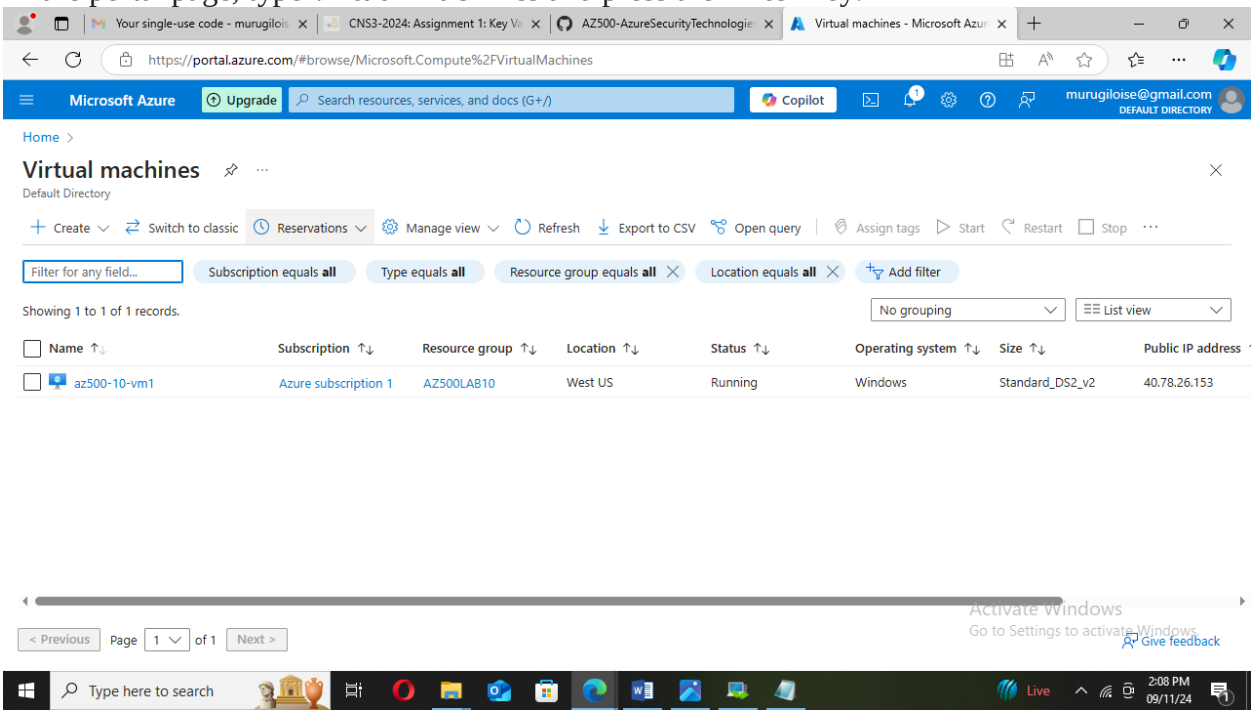


Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)

Task 4: Log on to the Azure VM running Visual Studio 2019 and SQL Management Studio 19

In this task, you log on to the Azure VM, which deployment you initiated in Exercise 1. This Azure VM hosts Visual Studio 2019 and SQL Server Management Studio 19.

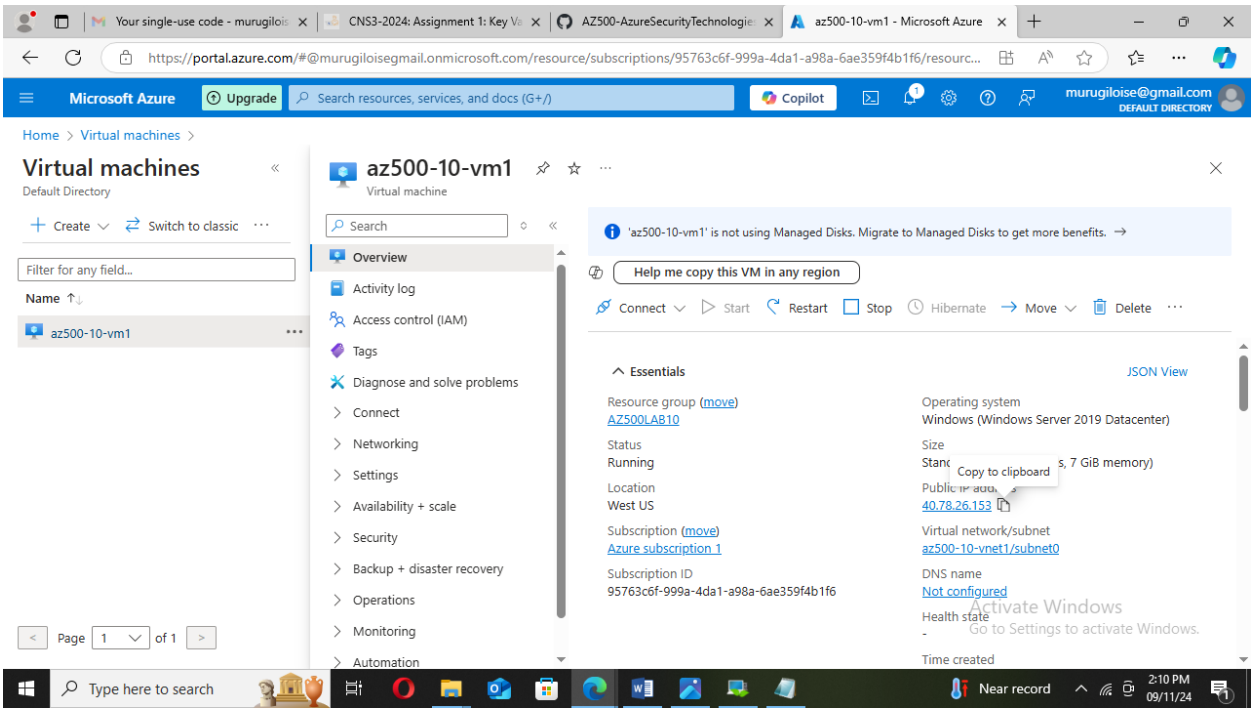
1. In the Azure portal, in the **Search resources, services, and docs** text box at the top of the Azure portal page, type **virtual machines** and press the **Enter** key.



2. In the list of Virtual Machines shown, select the **az500-10-vm1** entry. On the **az500-10-vm1** blade, on the **Essentials** pane, take note of the **Public IP address**. You will use this later.

Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)

Public IP address 40.78.26.153



Task 5: Create a table in the SQL Database and select data columns for encryption

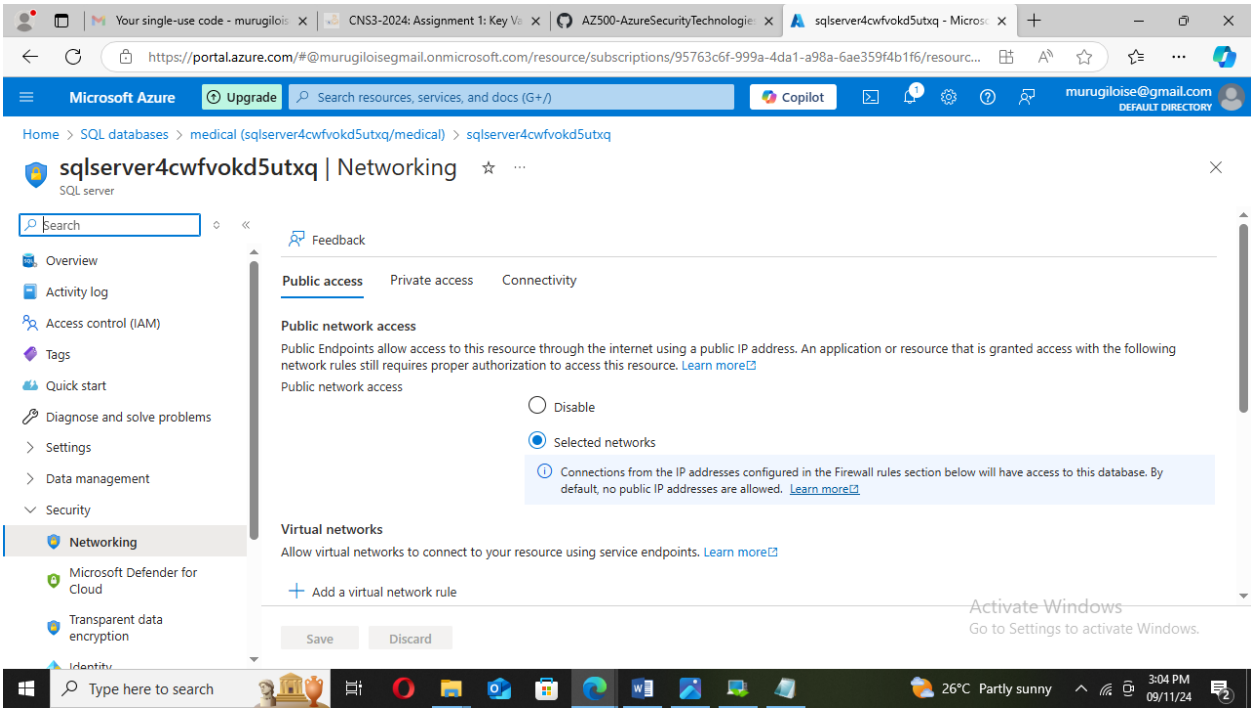
In this task, you will connect to the SQL Database with SQL Server Management Studio and create a table. You will then encrypt two data columns using an autogenerated key from the Azure Key Vault.

1. In the Azure portal, navigate to the blade of the **medical** SQL database, in the **Essentials** section, identify the **Server name** (copy to clipboard), and then, in the toolbar, click **Set server firewall**.

   Note: Record the server name. You will need the server name later in this task.

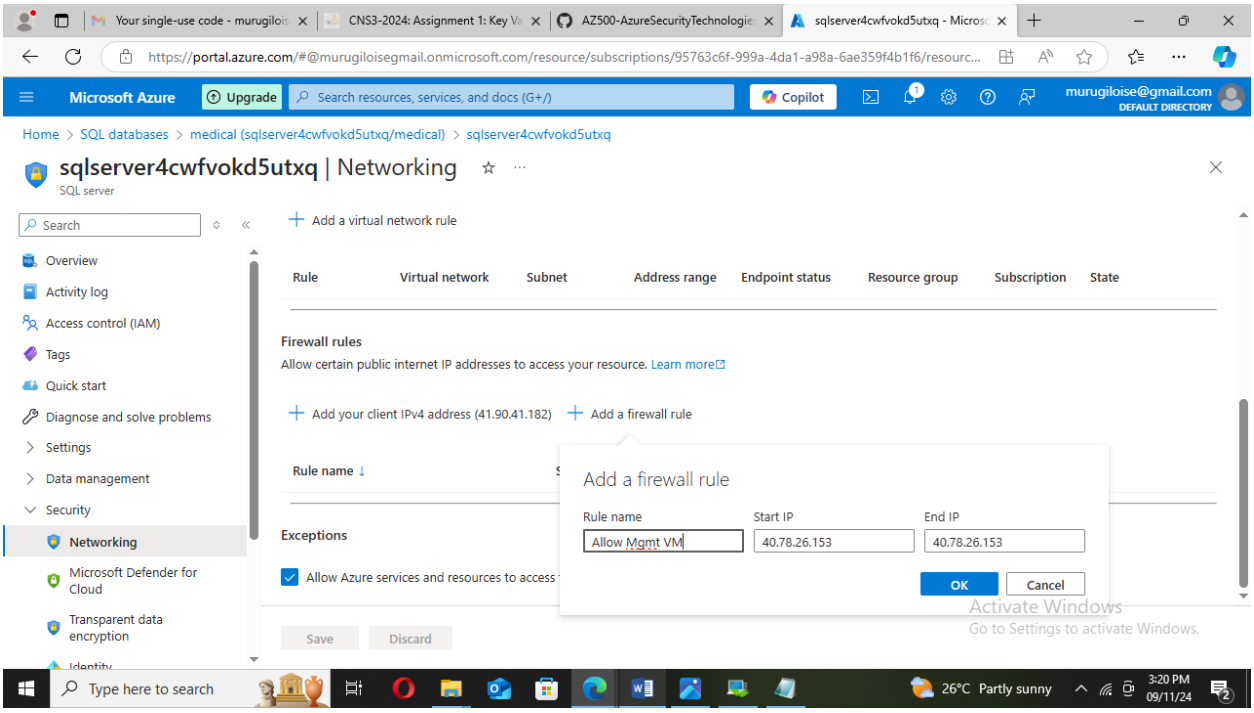
   Server name: sqlserver4cwfvokd5utxq.database.windows.net

Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)



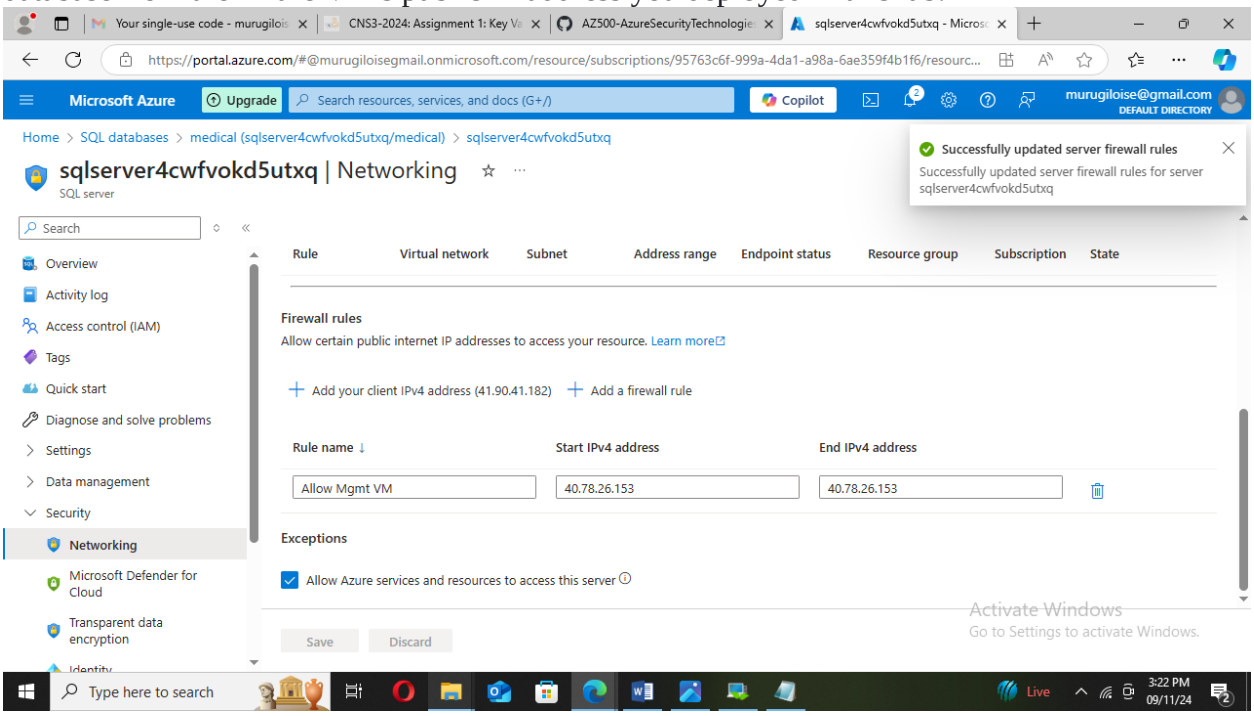
2. On the **Firewall settings** blade, scroll down to Rule Name, click **+ Add a firewall rule**, and specify the following settings:

Setting	Value
Rule Name	**Allow Mgmt VM**
Start IP	the Public IP Address of the az500-10-vm1
End IP	the Public IP Address of the az500-10-vm1

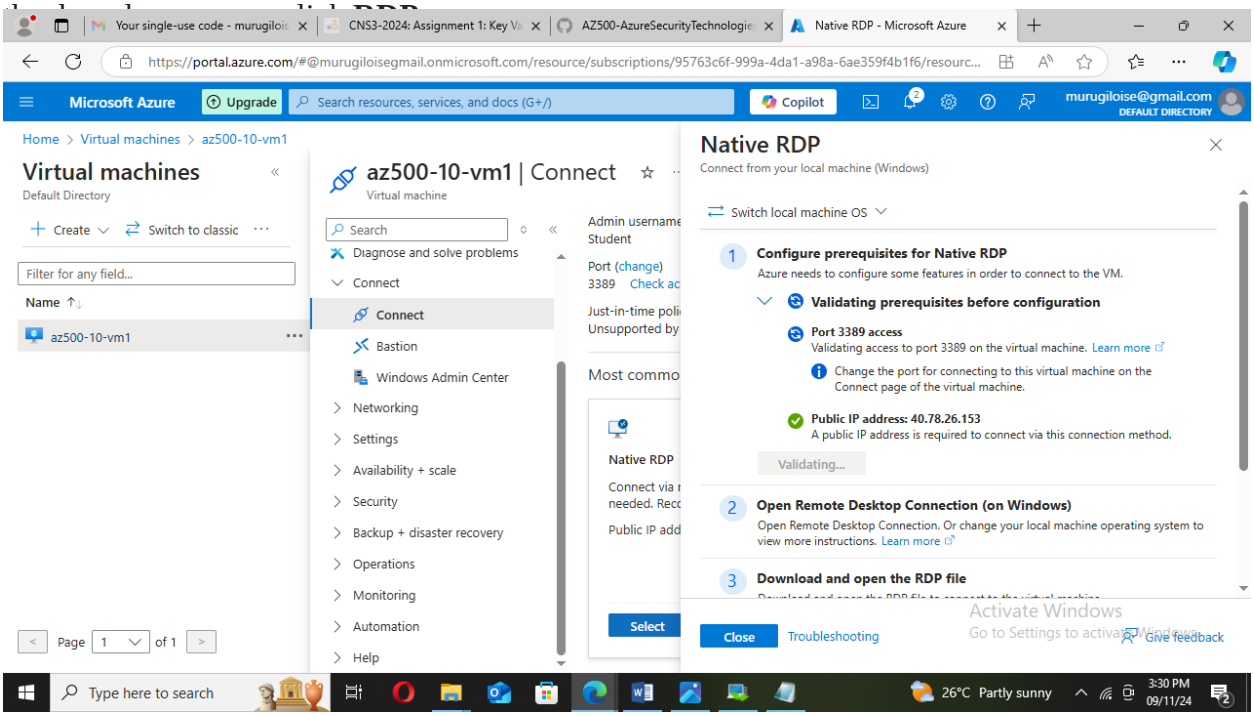


3. Click **Save** to save the change and close the confirmation pane.

Note: This modifies the server firewall settings, allowing connections to the medical database from the Azure VM's public IP address you deployed in this lab.



4. Navigate back to the **az500-10-vm1** blade, click **Overview**, next click **Connect** and, in



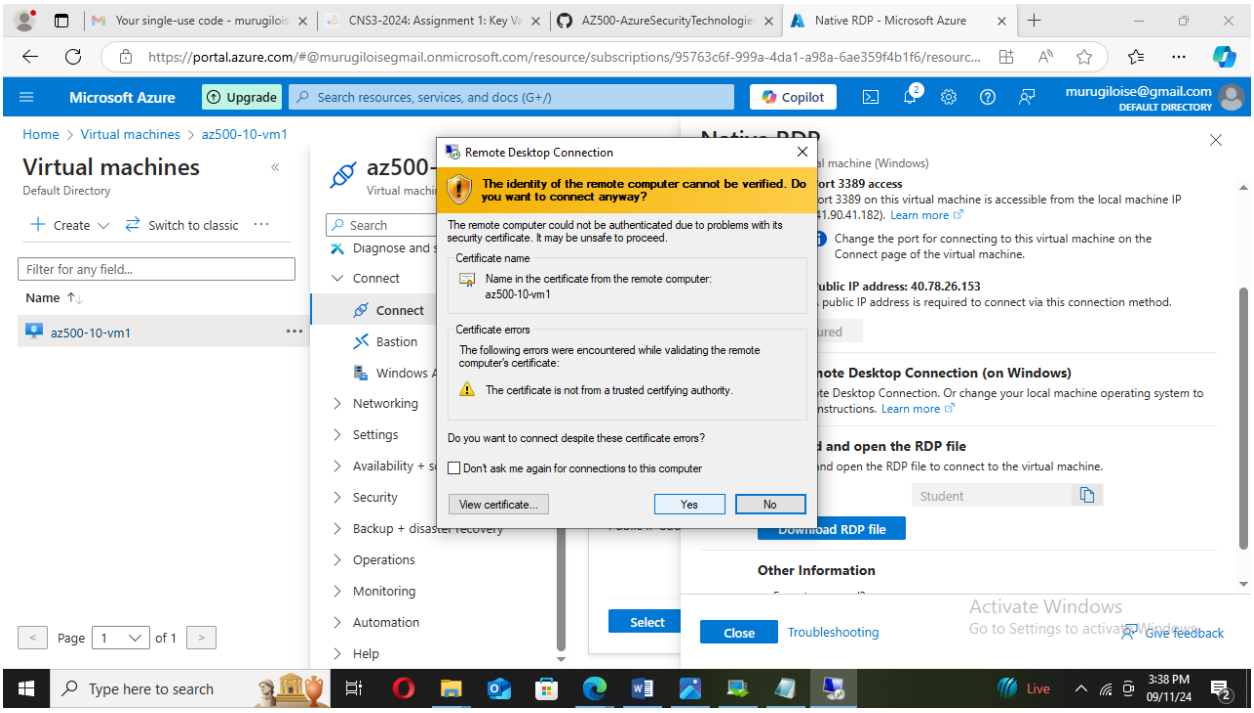
5. Click **Download RDP File** and use it to connect to the **az500-10-vm1** Azure VM via Remote Desktop. When prompted to authenticate, provide the following credntials:

Setting	Value
Username	**Student**
Password	**Please use your personal password created in Lab 02 > Exercise 1 > Task 1 > Step 9.**

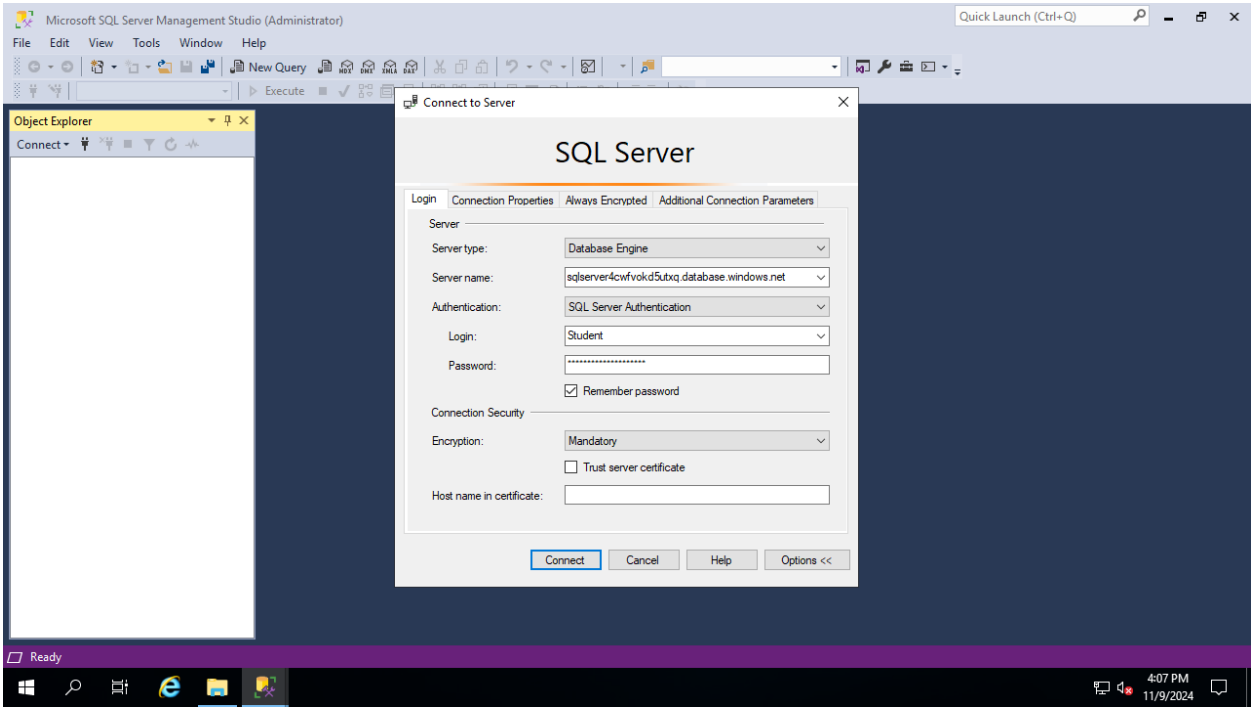
Note: Wait for the Remote Desktop session and **Server Manager** to load. Close Server Manager.

Note: The remaining steps in this lab are performed within the Remote Desktop session to the **az500-10-vm1** Azure VM.

Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)



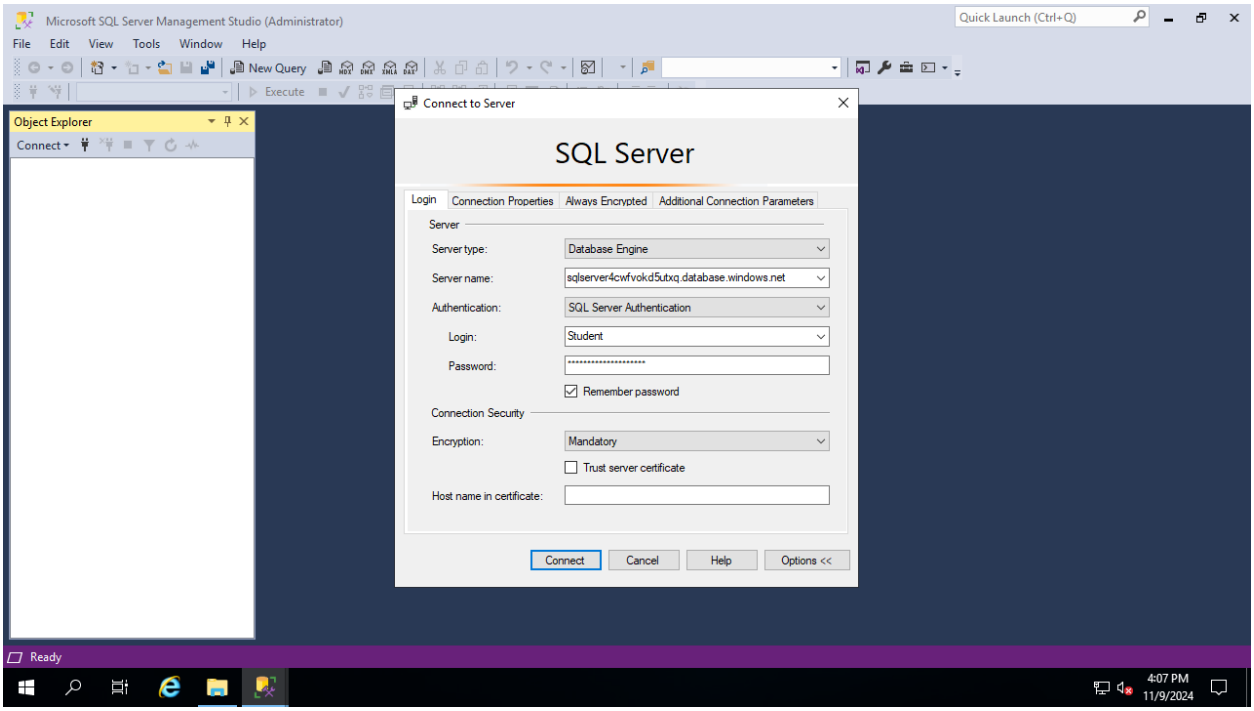
6. Install SQL Server Management Studio on **az500-10-vm1.** Azure VM.
7. Open **SQL Server Management Studio.** In this part we've opened the sql server management studio



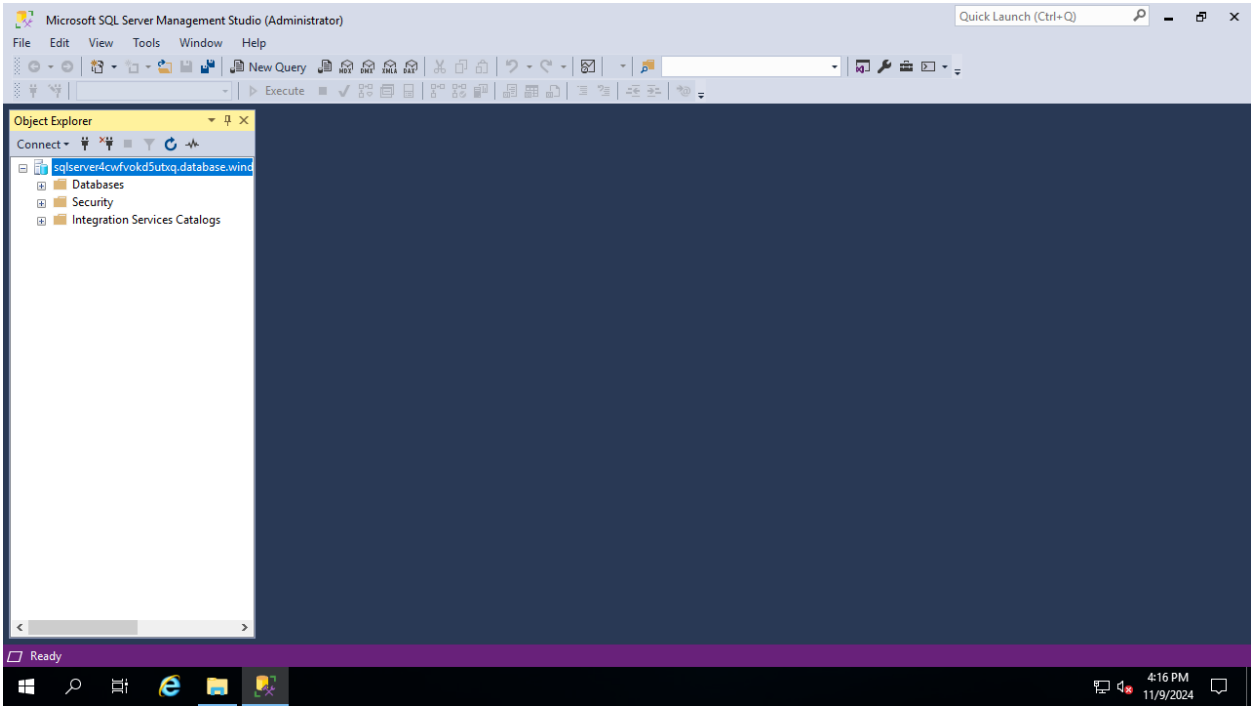
8. In the **Connect to Server** dialog box, specify the following settings:

Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)

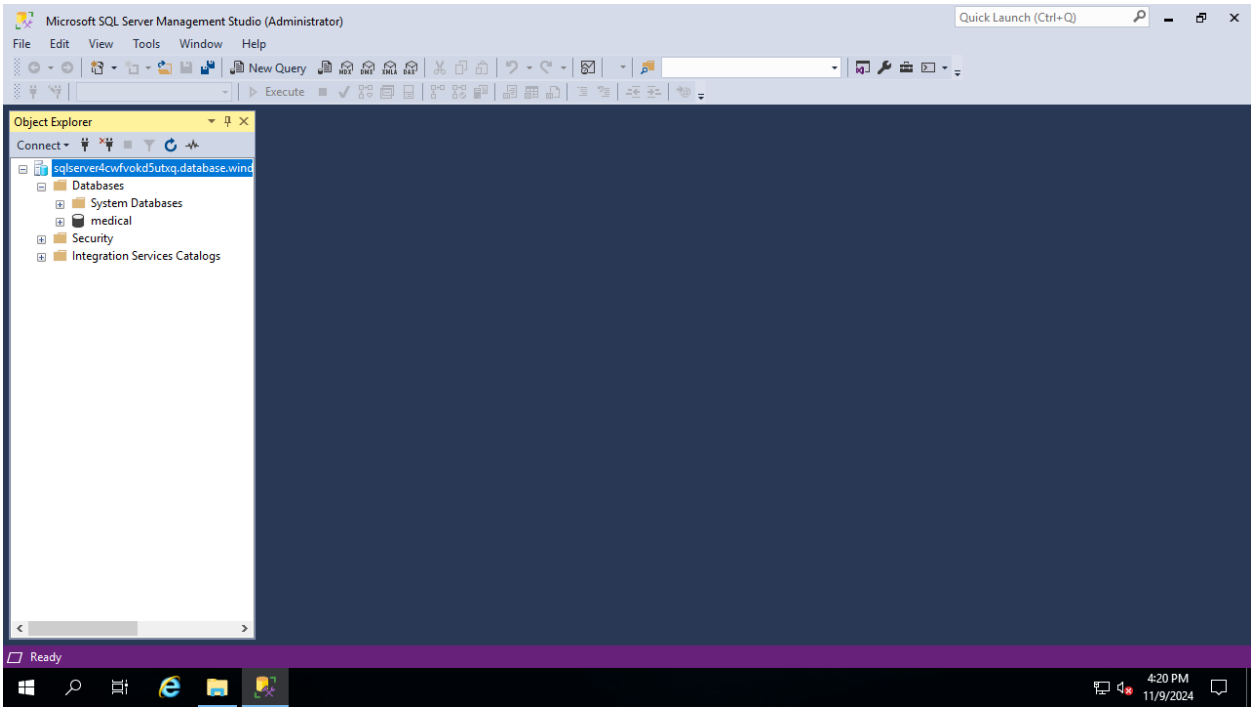
Setting	Value
Server Type	**Database Engine**
Server Name	the server name you identified earlier in this task
Authentication	**SQL Server Authentication**
Username	**Student**
Password	**Please use your personal password created in Lab 02 > Exercise 2 > Task 1 > Step 3.**



9. In the **Connect to Server** dialog box, click **Connect**. We have successfully connected to the server.

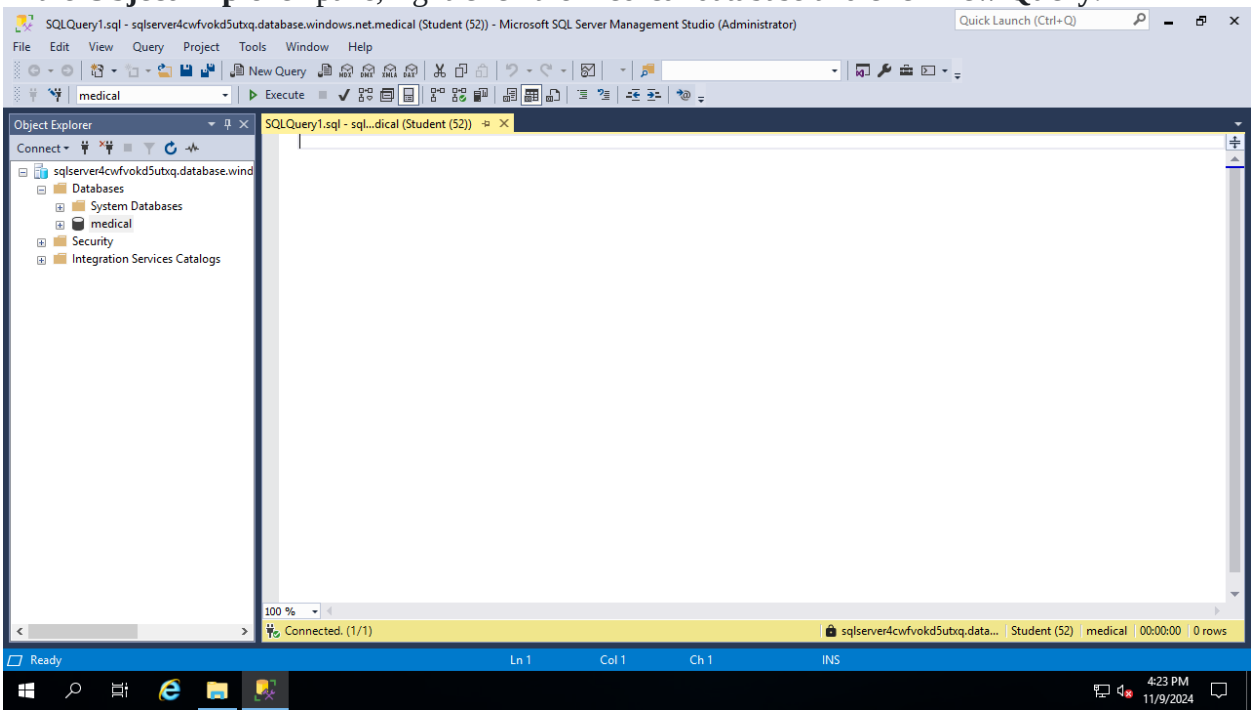


10. Within the **SQL Server Management Studio** console, in the **Object Explorer** pane, expand the **Databases** folder. From the screenshot below we can see our database Medical



Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)

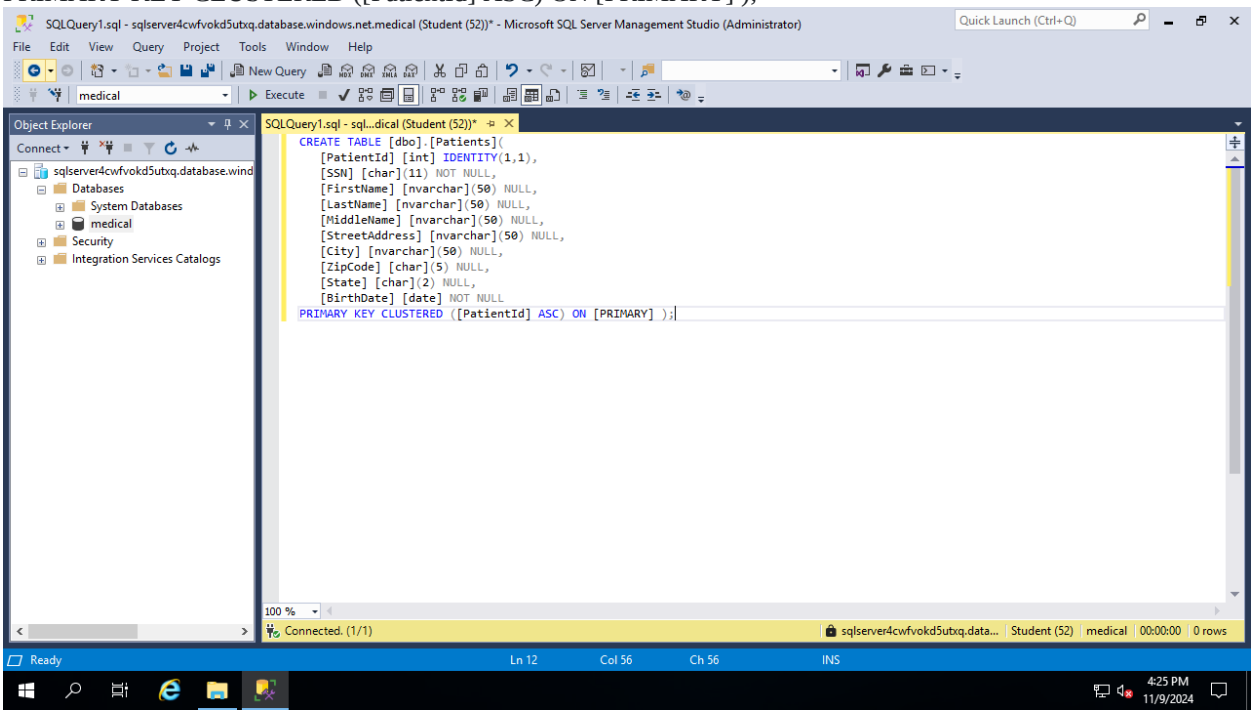
11. In the **Object Explorer** pane, right-click the **medical** database and click **New Query**.



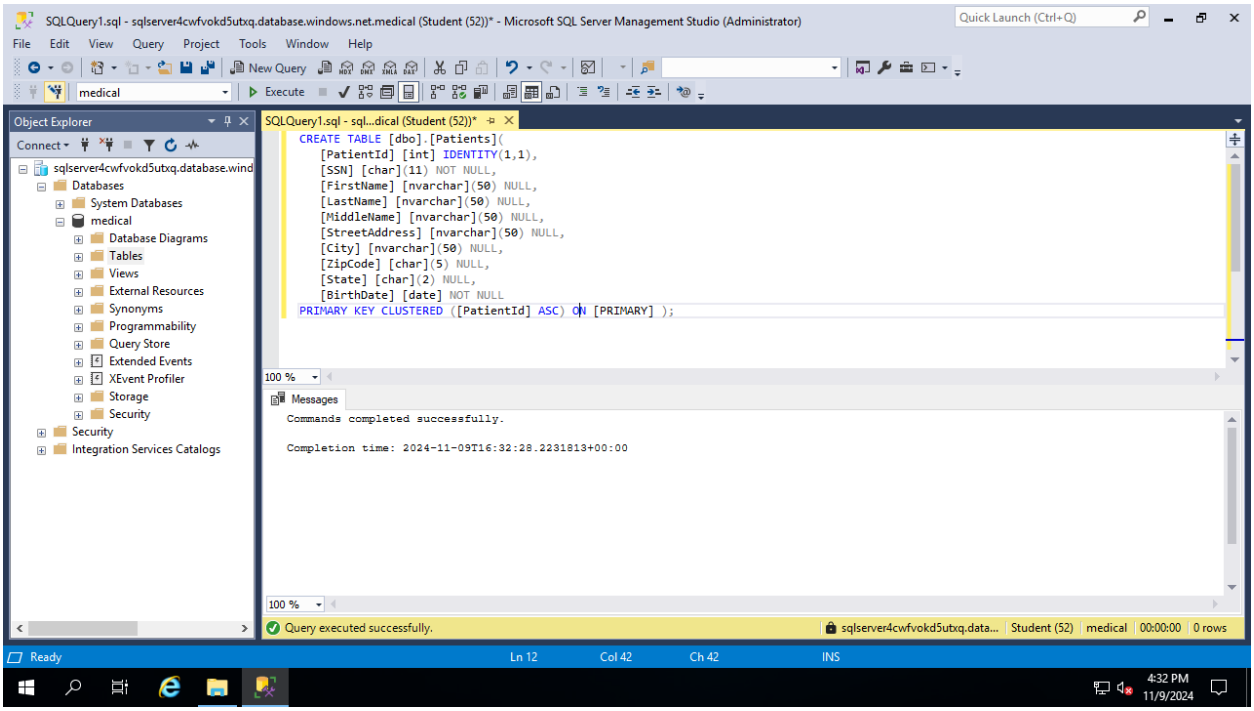
12. Paste the following code into the query window and click **Execute**. This will create a **Patients** table.

```
CREATE TABLE [dbo].[Patients](
 [PatientId] [int] IDENTITY(1,1),
[SSN] [char](11) NOT NULL,
[FirstName] [nvarchar](50) NULL,
[LastName] [nvarchar](50) NULL,
[MiddleName] [nvarchar](50) NULL,
[StreetAddress] [nvarchar](50) NULL,
[City] [nvarchar](50) NULL,
[ZipCode] [char](5) NULL,
[State] [char](2) NULL,
[BirthDate] [date] NOT NULL
```

```
PRIMARY KEY CLUSTERED ([PatientId] ASC) ON [PRIMARY] );
```

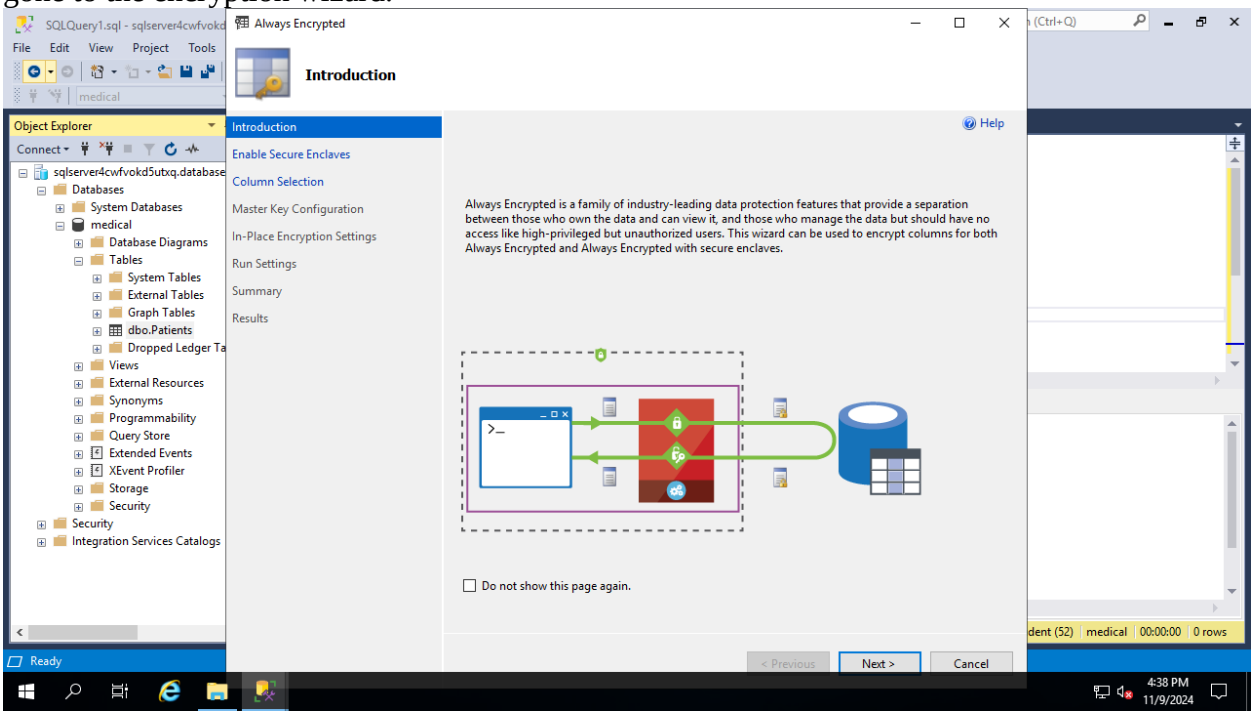


The command has been executed successfully.

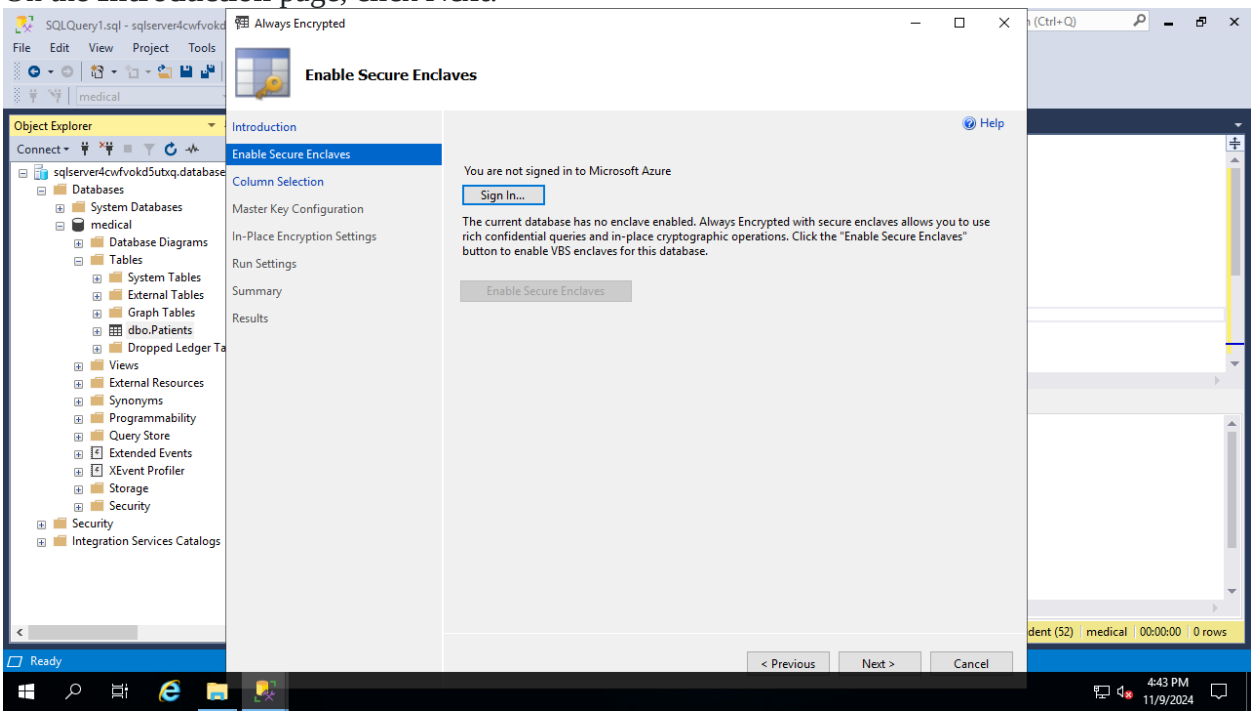


13. After the table is created successfully, in the **Object Explorer** pane, expand the **medical** database node, the **tables** node, right-click the **dbo.Patients** node, and click **Encrypt Columns**.

Note: This will initiate the **Always Encrypted** wizard. From the screenshot below we've gone to the encryption wizard.

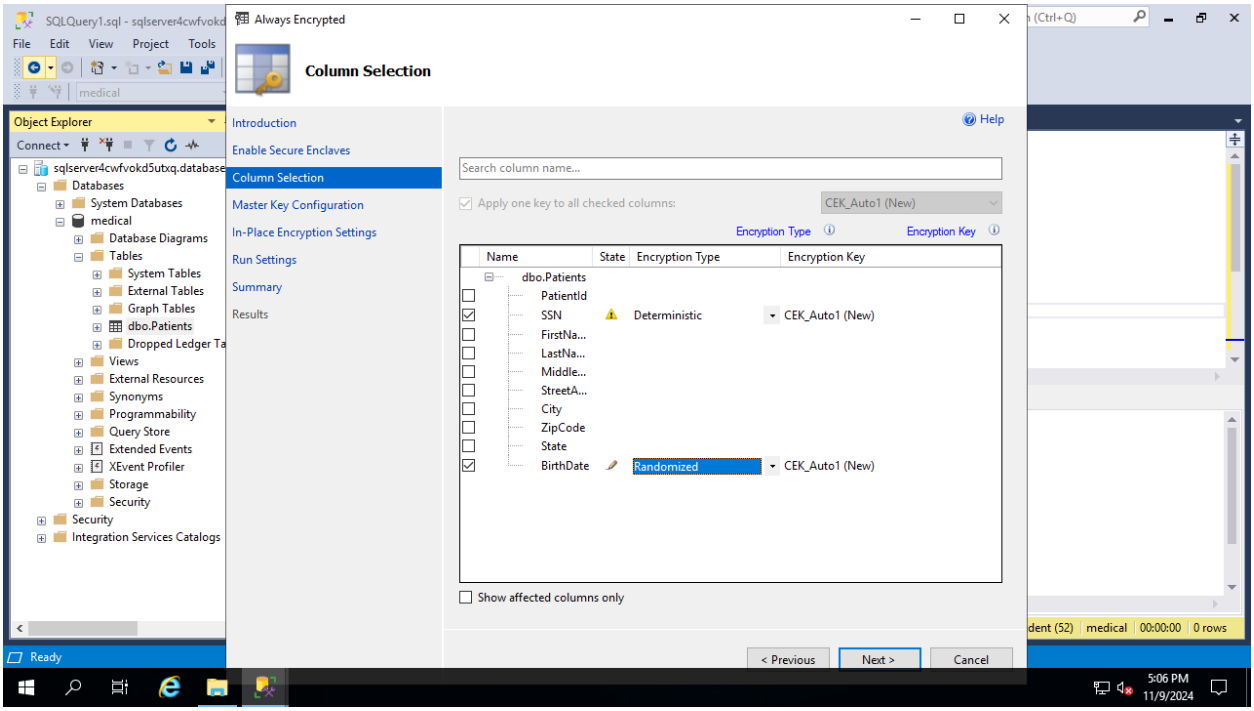


14. On the **Introduction** page, click **Next**.



15. On the **Column Selection** page, select the **SSN** and **Birthdate** columns, set the **Encryption Type** of the **SSN** column to **Deterministic** and of the **Birthdate** column to **Randomized**, and click **Next**.

From the screenshot below we have set **SSN** column to **Deterministic** and of the **Birthdate** column to **Randomized**

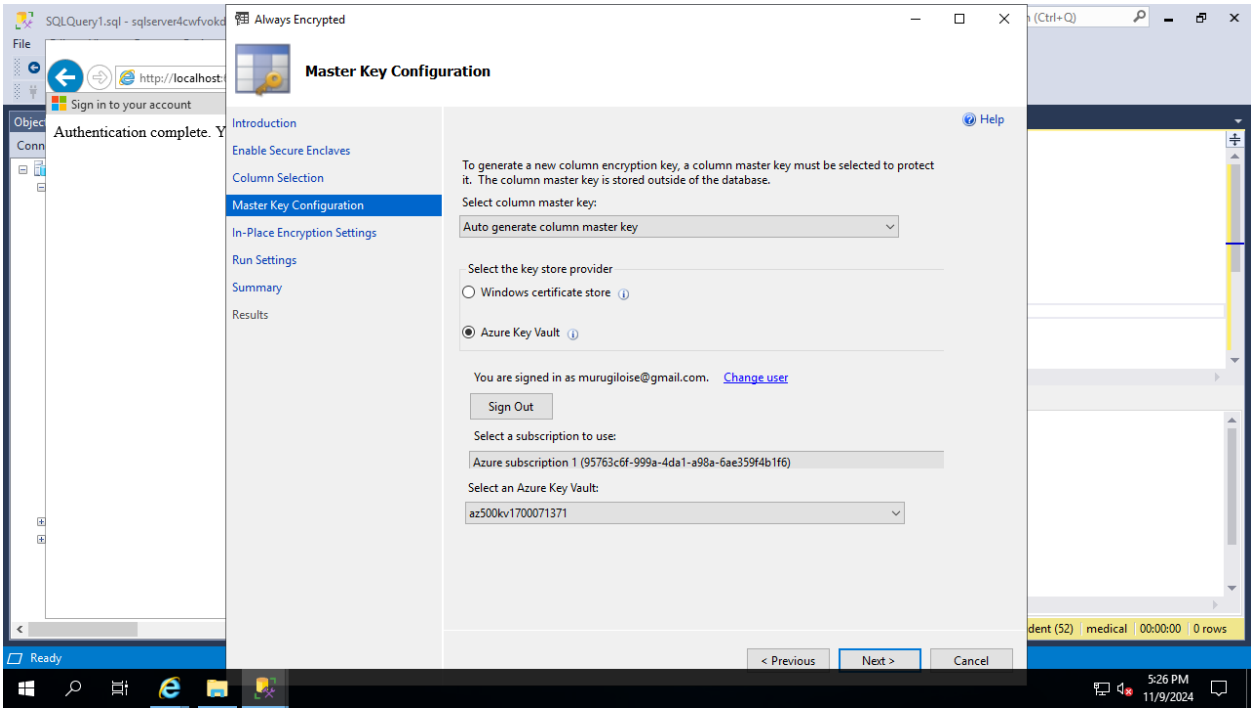


Note: While performing the encryption if any error thrown like **Exception has been thrown by the target of an innvocation** related to **Rotary(Microsoft.SQLServer.Management.ServiceManagement)** then make sure the **Key Permission's** values of **Rotation Policy Operations** are **unchecked**, if not in the Azure portal navigate to the **Key Vault** >> **Access Policies** >> **Key Permissions** >> Uncheck all the values under the **Rotation Policy Operations** >> Under **Privileged Key Operations** >> Uncheck **Release**.

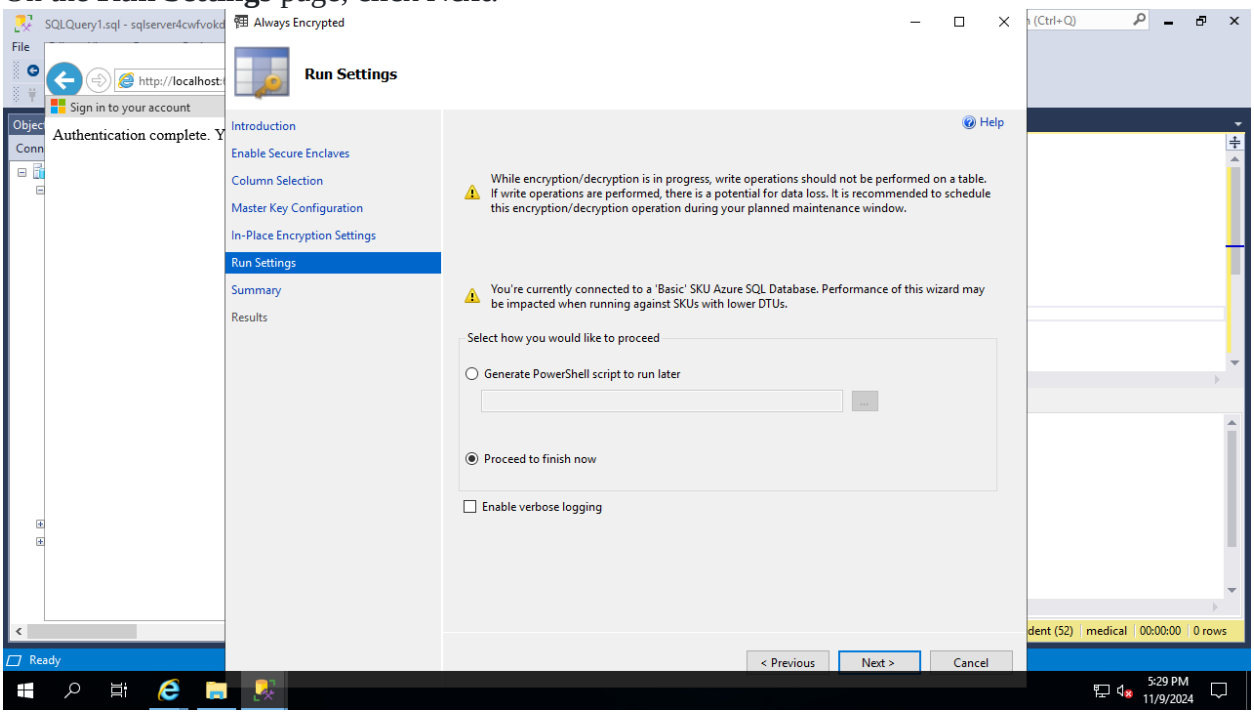
16. On the **Master Key Configuration** page, select **Azure Key Vault**, click **Sign in**, when prompted, authenticate by using the same user account you used to provision the Azure Key Vault instance earlier in this lab, ensure that that Key Vault appears in the **Select an Azure Key Vault** drop down list, and click **Next**.

    After logging in to Azure we wre able to pull the Azure Key Vault and the subscription as seen below.

Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)



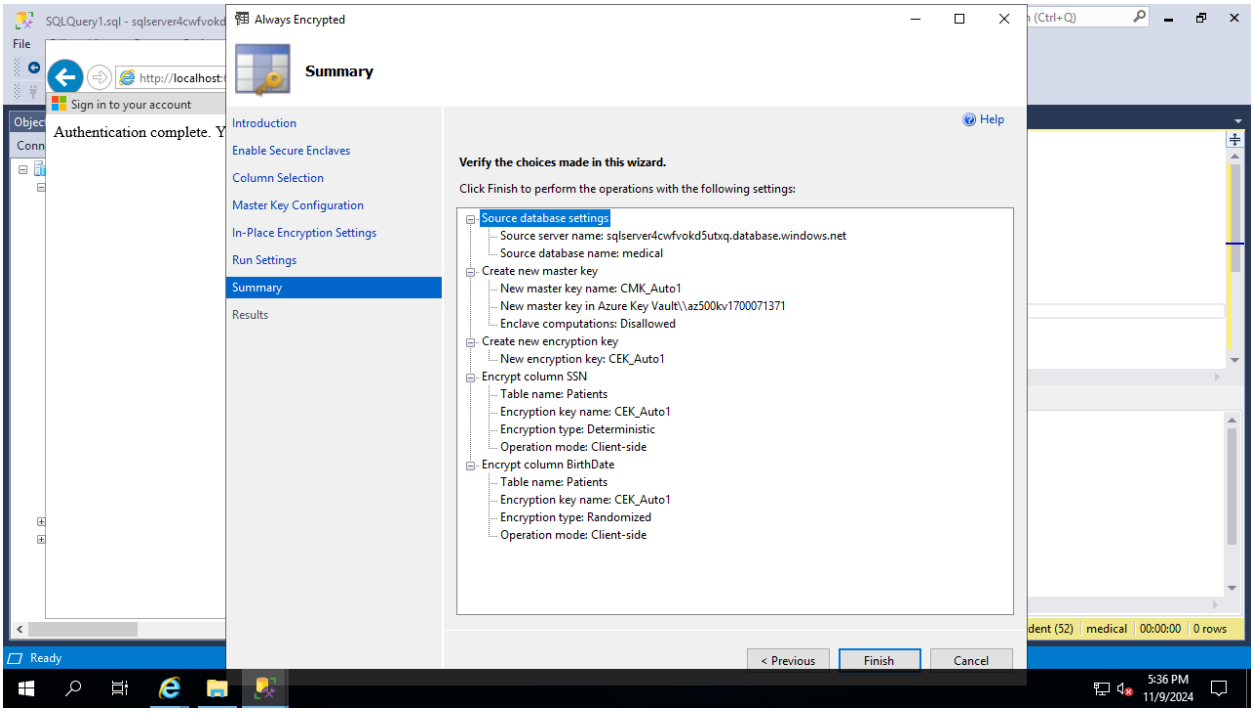
17. On the **Run Settings** page, click **Next**.



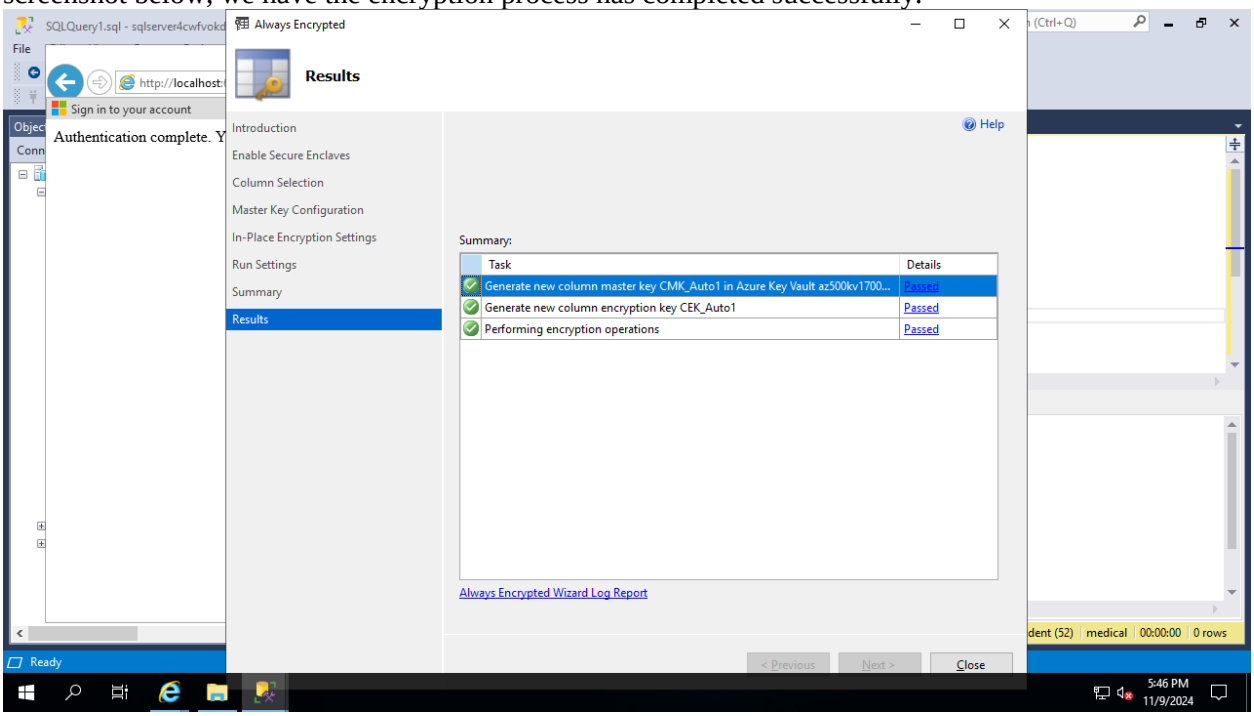
18. On the **Summary** page, click **Finish** to proceed with the encryption. When prompted, sign in again by using the same user account you used to provision the Azure Key Vault instance earlier in this lab.

    At this point we click on finish.

Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)



19. Once the encryption process is complete, on the **Results** page, click **Close**. From The screenshot below, we have the encryption process has completed successfully.

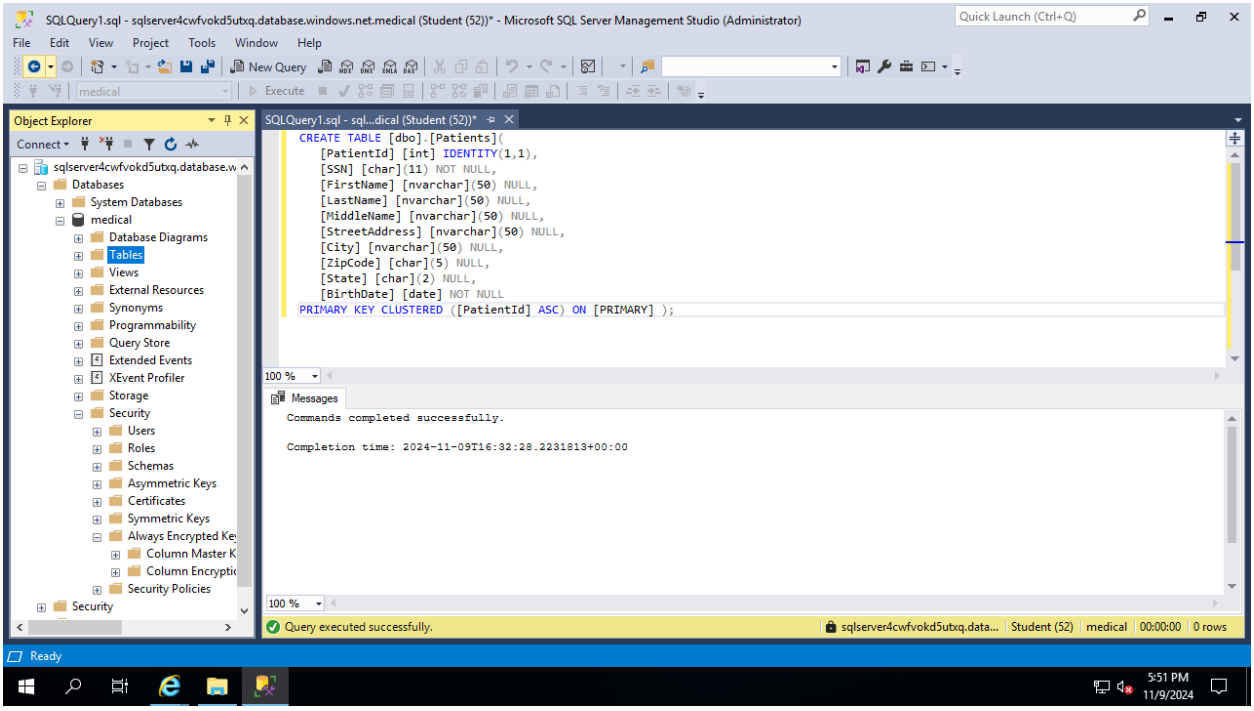


20. In the **SQL Server Management Studio** console, in the **Object Explorer** pane, under the **medical** node, expand the **Security** and **Always Encrypted Keys** subnodes.

    Note: The **Always Encrypted Keys** subnode contains the **Column Master Keys** and **Column Encryption Keys** subfolders.

Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)

From the screenshot below, under Security and Always Encrypted Keys subnodes. We can see the Column Master Keys and Column Encryption Keys subfolders.

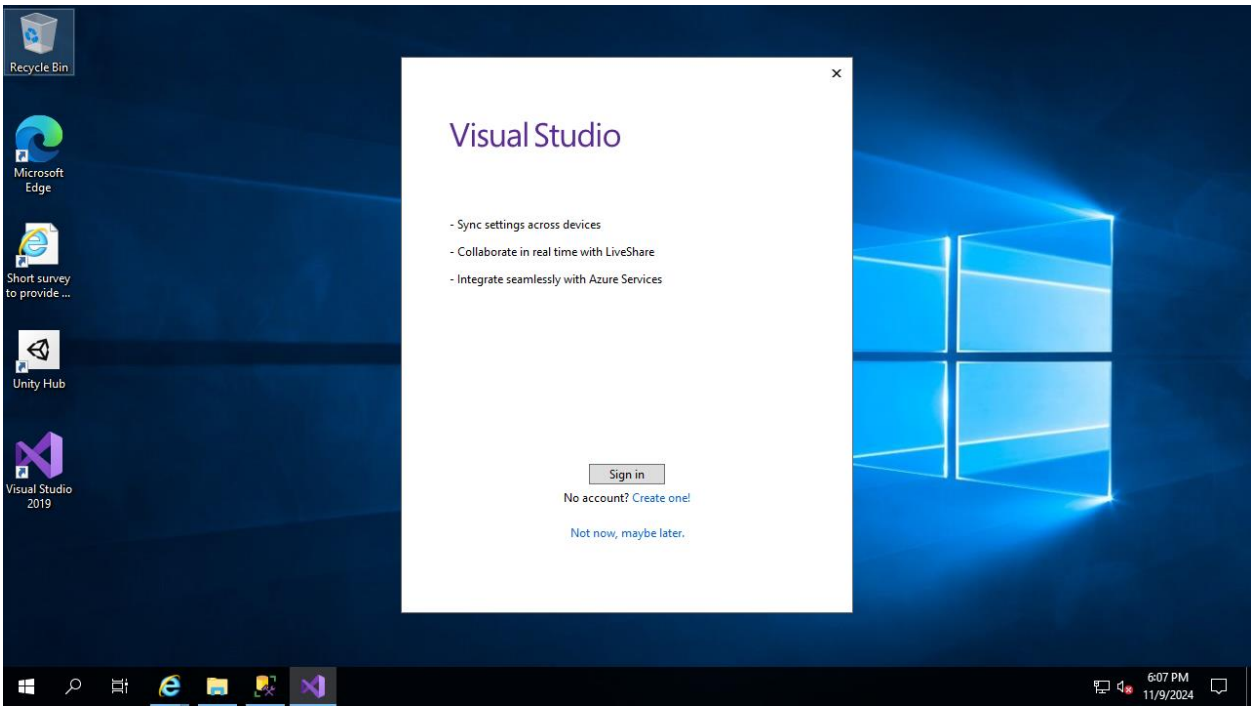


Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)

Exercise 4: Demonstrate the use of Azure Key Vault in encrypting the Azure SQL database

Task 1: Run a data-driven application to demonstrate the use of Azure Key Vault in encrypting the Azure SQL database

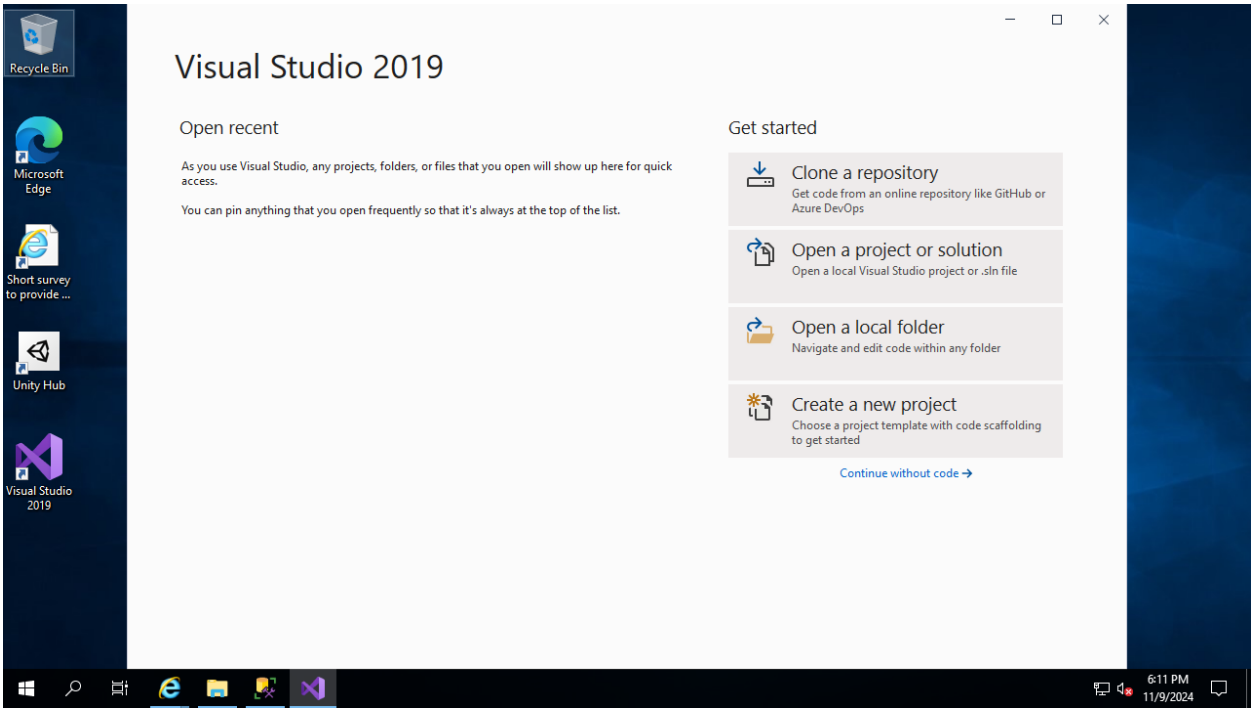
1. From the RDP session to the **az500-10-vm1**, launch **Visual Studio 2019** from the **Start menu**.



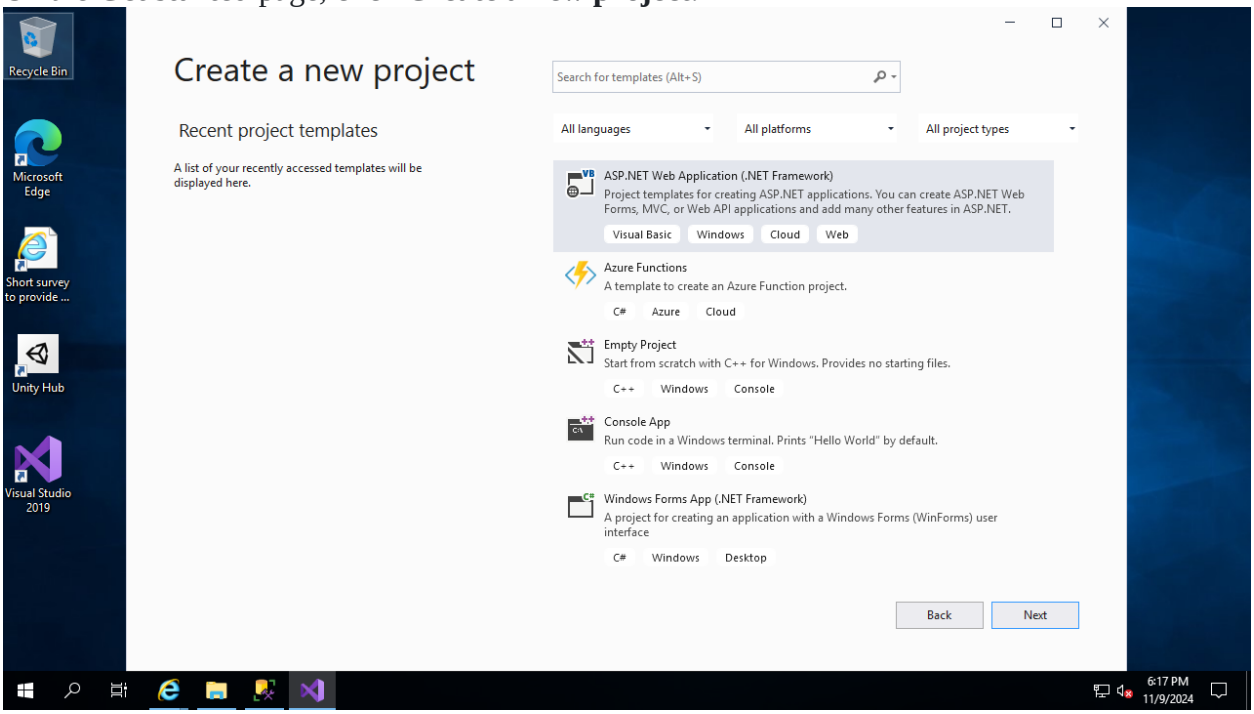
2. Switch to the window displaying Visual Studio 2019 welcome message, click the **Sign in** button and, when prompted, provide the credentials you used to authenticate to the Azure subscription you are using in this lab.

   We have been able to login successfully to the visual studio as seen below.

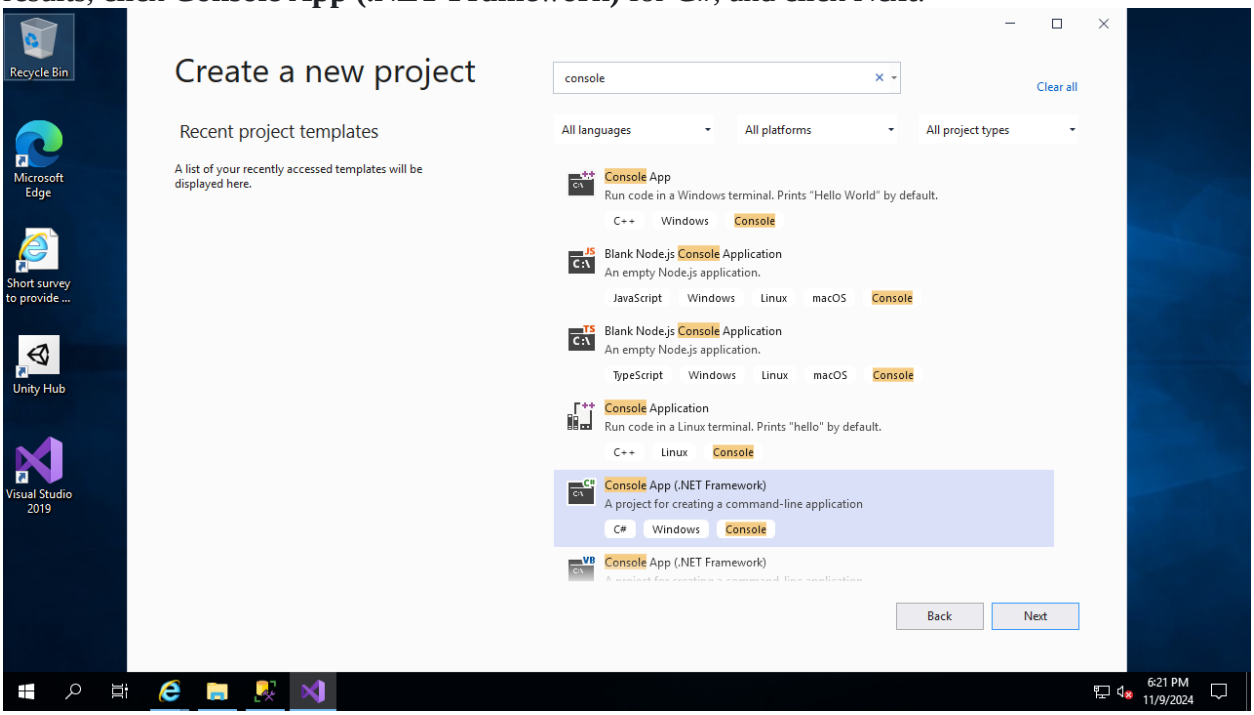
Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)



3. On the **Get started** page, click **Create a new project**.



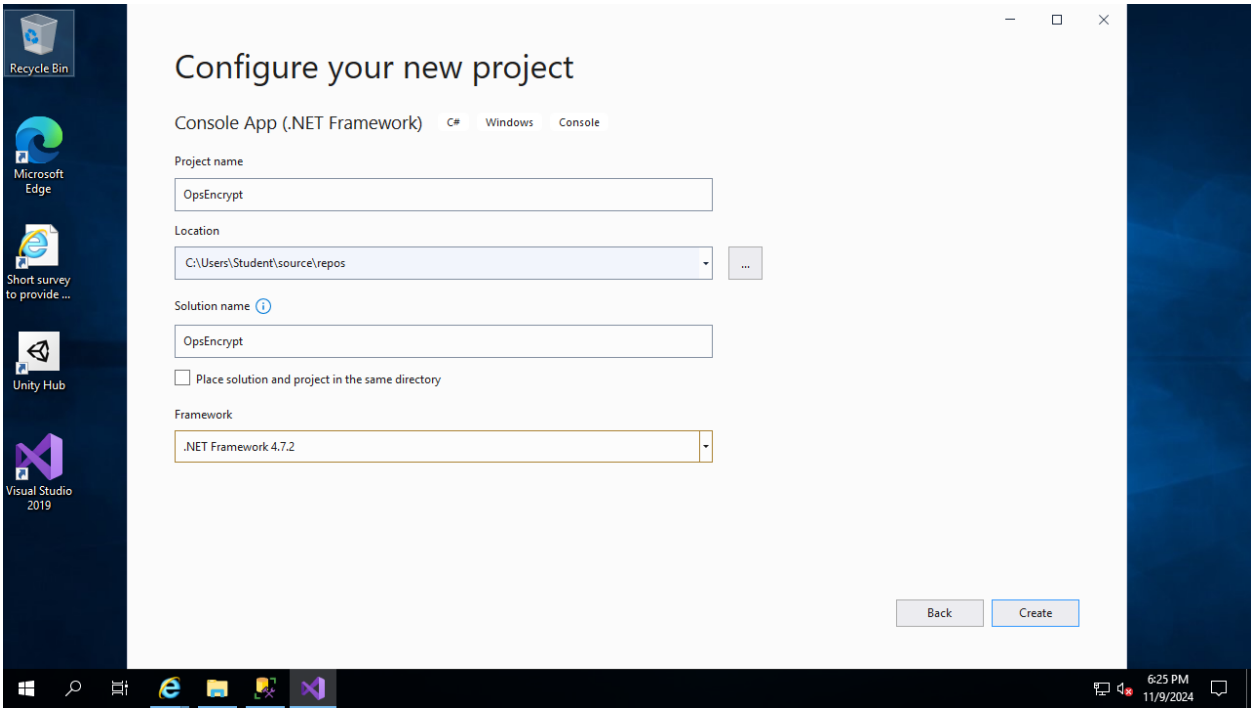
4. In the list of project templates, search for **Console App (.NET Framework)**, in the list of results, click **Console App (.NET Framework)** for **C#**, and click **Next**.



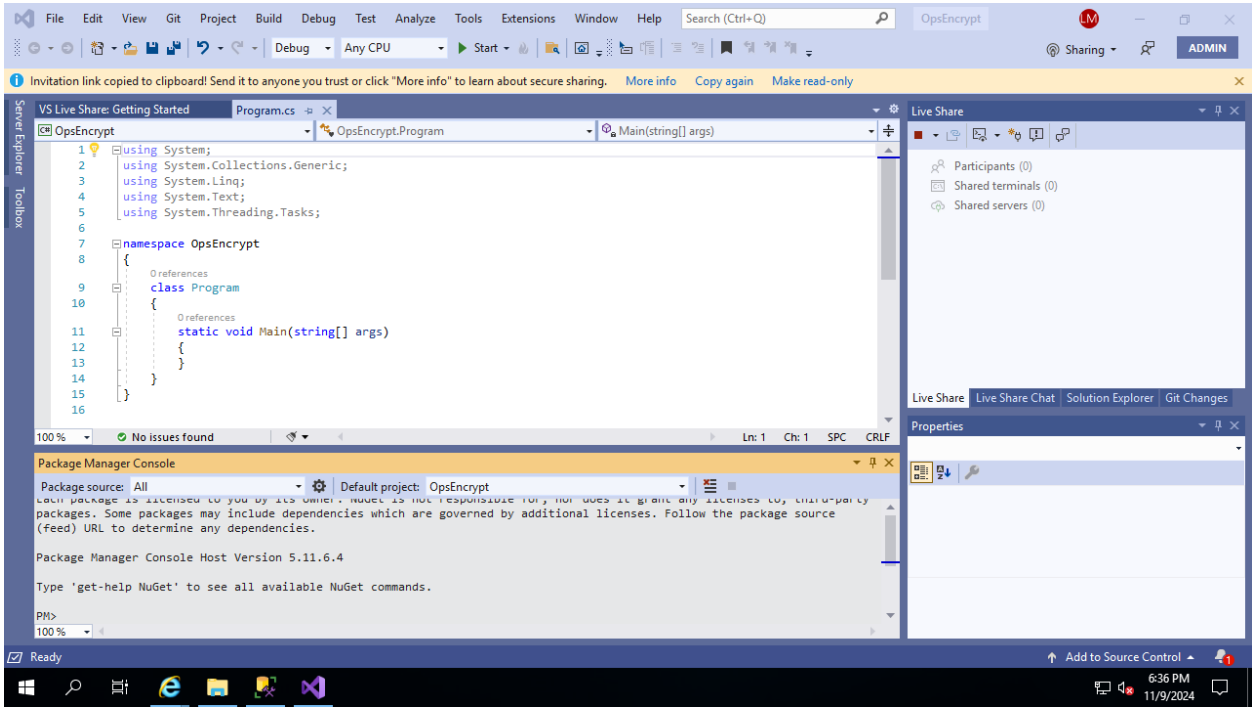
5. On the **Configure your new project** page, specify the following settings (leave other settings with their default values), then click **Create**:

Setting	**Value**
Project name	**OpsEncrypt**
Solution name	**OpsEncrypt**
Framework	**.NET Framework 4.7.2**

FDrom the screenshot below we have configured our new project **OpsEncrypt**



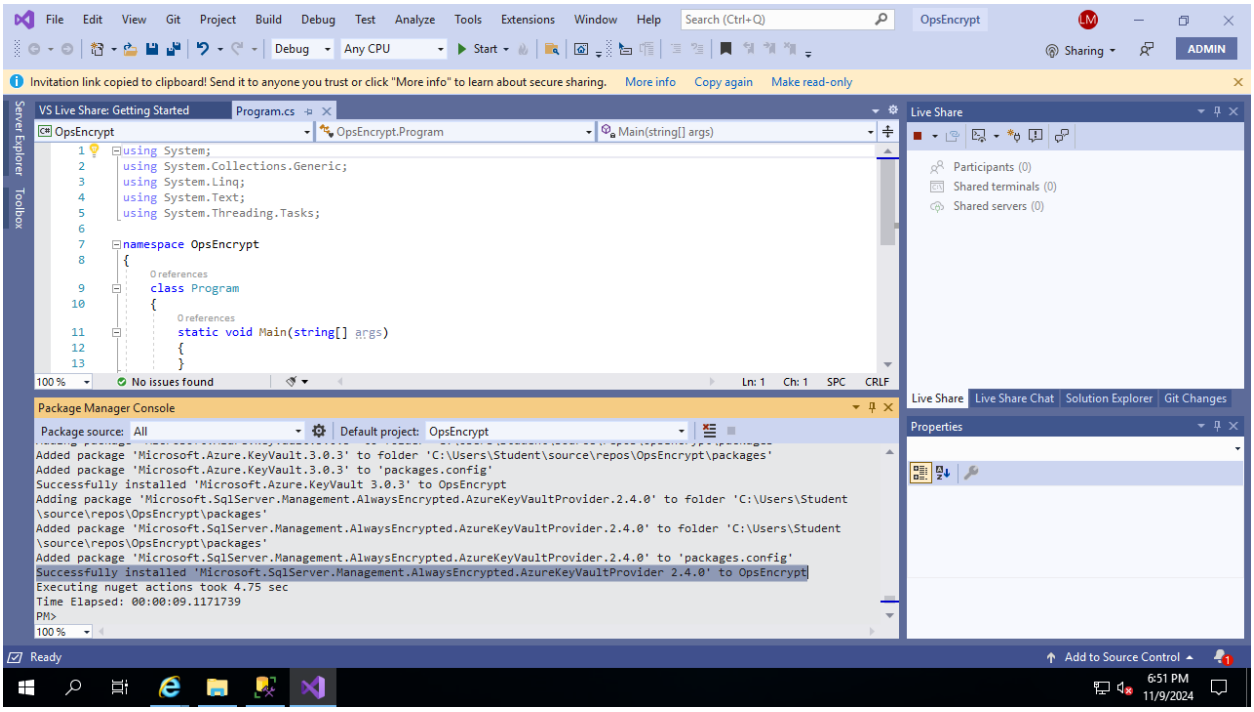
6. In the Visual Studio console, click the **Tools** menu, in the drop down menu, click **NuGet Package Manager**, and, in the cascading menu, click **Package Manager Console**. From the screenshot below, we have successfully navigated to the Package Manager Console



7. In the **Package Manager Console** pane, run the following to install the first required **NuGet** package:

Install-Package Microsoft.SqlServer.Management.AlwaysEncrypted.AzureKeyVaultProvider

We have successfully installed the package Microsoft.SqlServer.Management.AlwaysEncrypted.AzureKeyVaultProvider

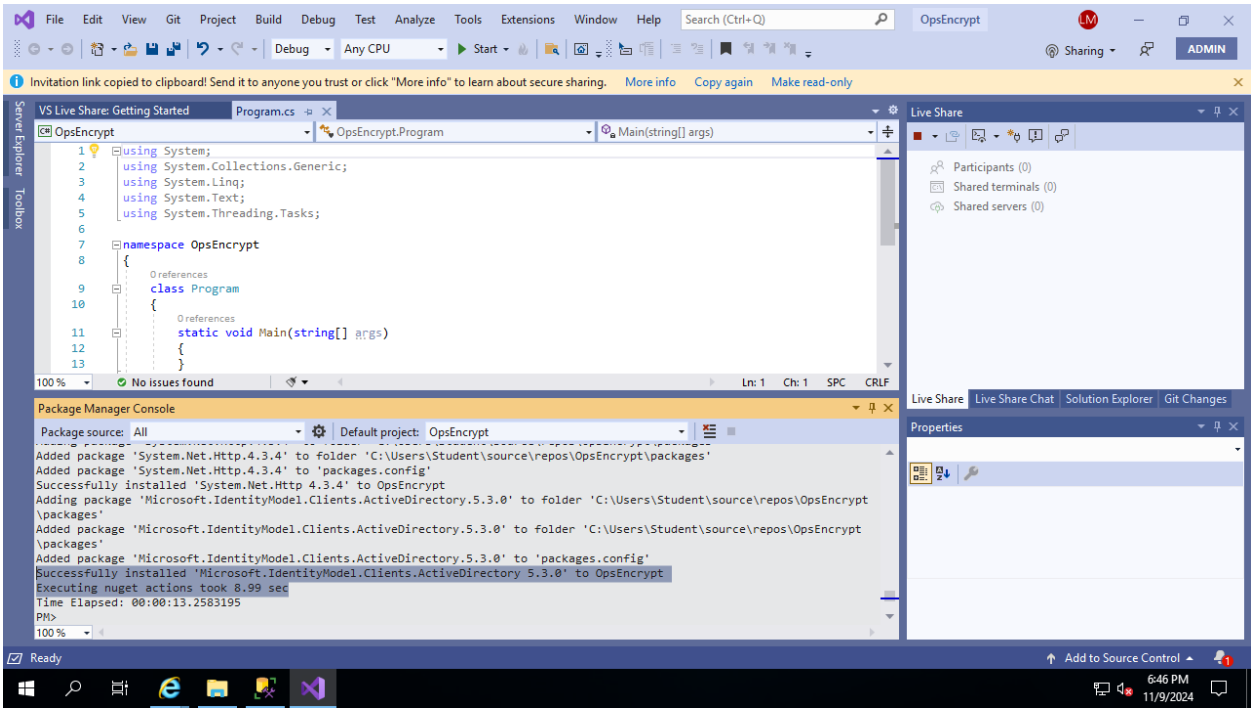


8. In the **Package Manager Console** pane, run the following to install the second required **NuGet** package:

Install-Package Microsoft.IdentityModel.Clients.ActiveDirectory

We have successfully installed the package Microsoft.IdentityModel.Clients.ActiveDirectory

Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)

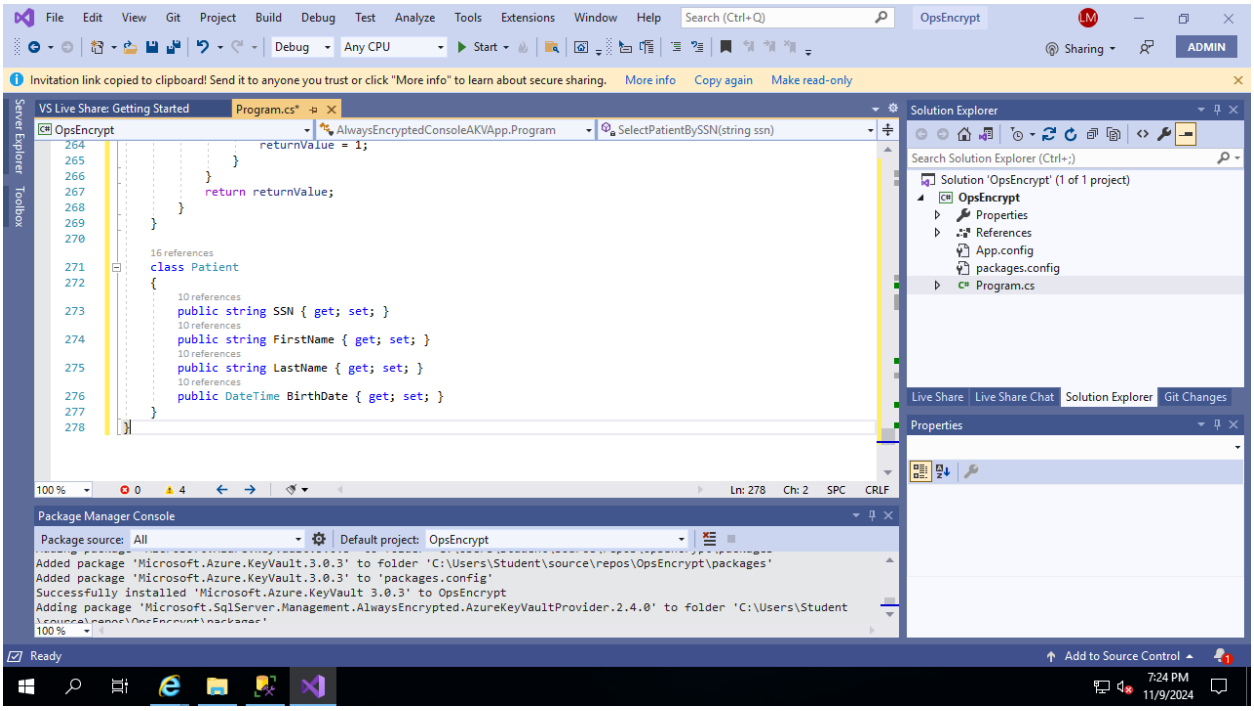


9. Minimize the RDP session to your Azure virtual machine, then navigate to **\Allfiles\Labs\10\program.cs**, open it in Notepad, and copy its content into Clipboard.

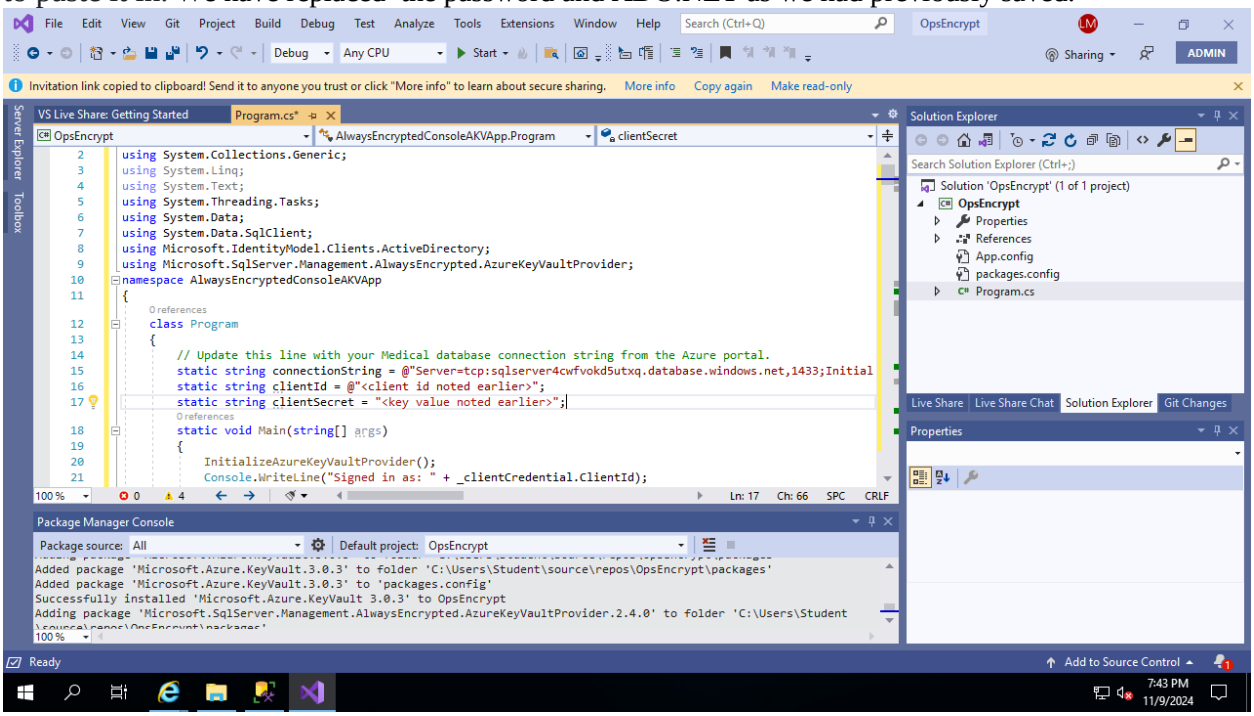
   We have copied the content program.cs on a clipboard

10. Return to the RDP session, and in the Visual Studio console, in the **Solution Explorer** window, click **Program.cs** and replace its content with the code you copied into Clipboard.

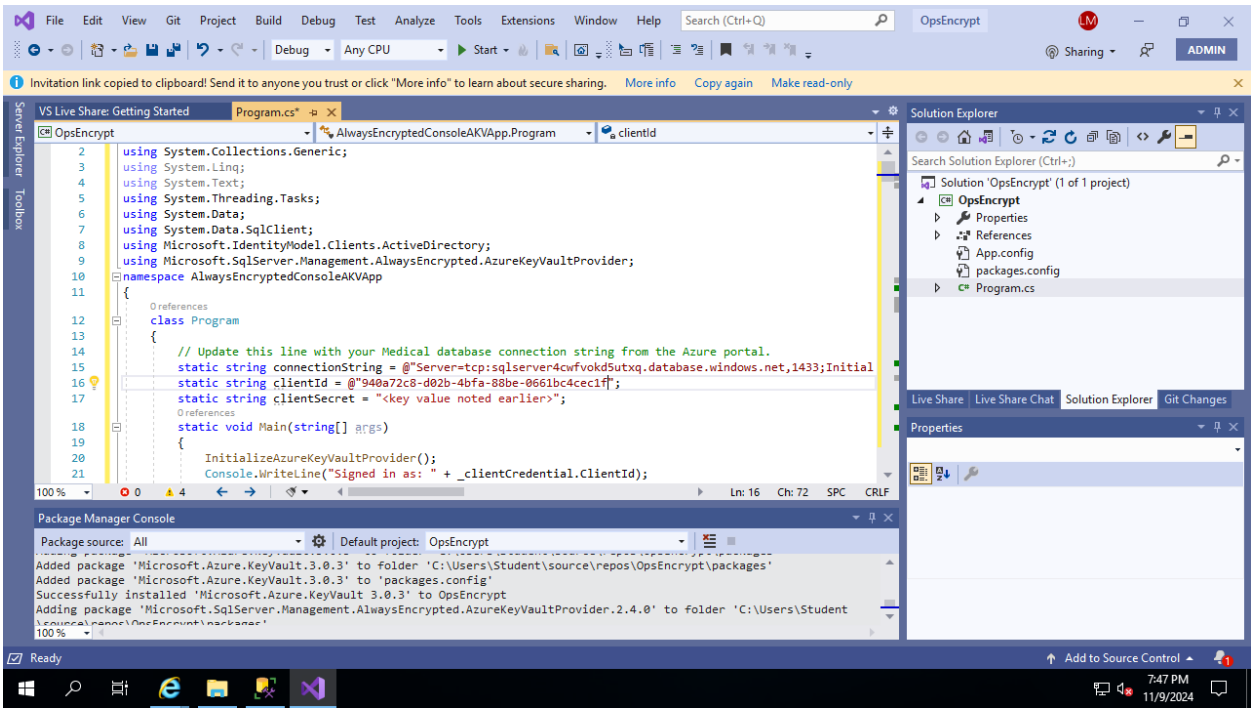
    We have replaced the code we had copied from program.cs lab 10 in the solution explorer on the visual studio as seen below.



11. In the Visual Studio window, in the **Program.cs** pane, in line 15, replace the <connection string noted earlier> placeholder with the Azure SQL database **ADO.NET** connection string you recorded earlier in the lab. In the connection string, replace the {your_password} placehodler, with the password that you specified in the deployment in Exercise 1. If you saved the string on the lab computer, you may need to leave the RDP session to copy the ADO string, then return to the Azure virtual machine to paste it in. We have replaced the password and ADO.NET as we had previously saved.

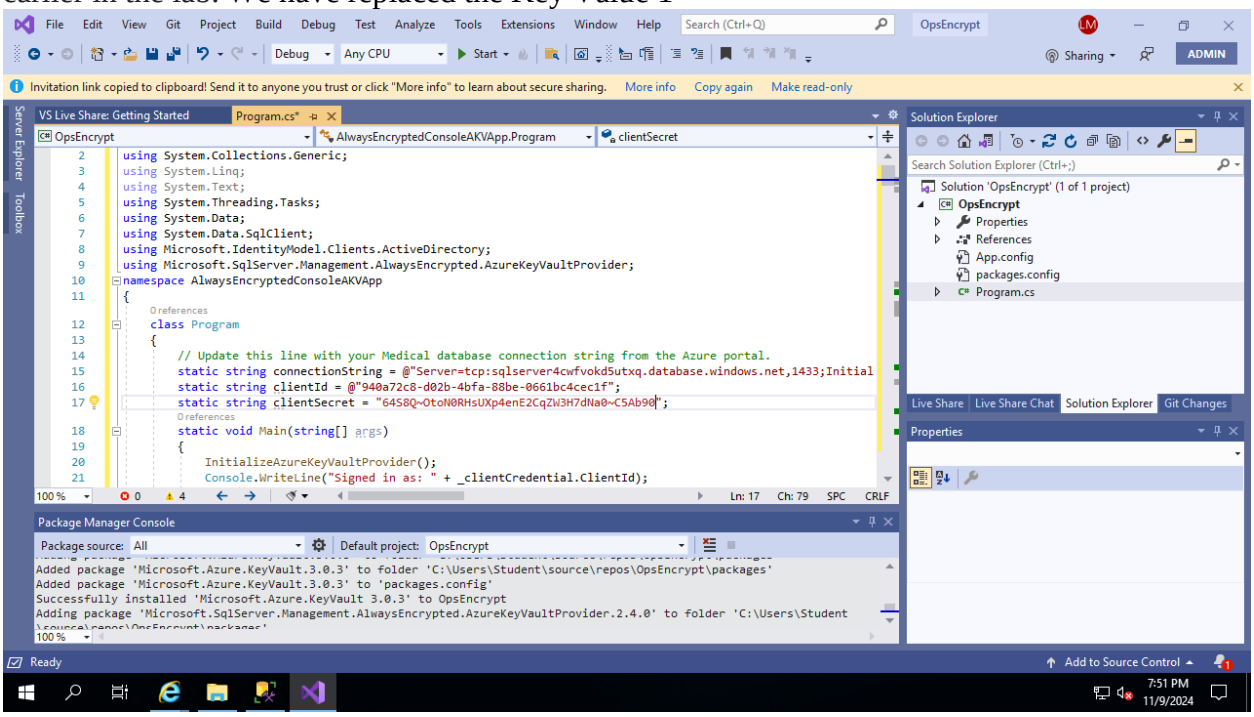


12. In the Visual Studio window, in the **Program.cs** pane, in line 16, replace the <client id noted earlier> placeholder with the value of **Application (client) ID** of the registered app you recorded earlier in the lab. We have replaced the Apploication ID as previously registered.

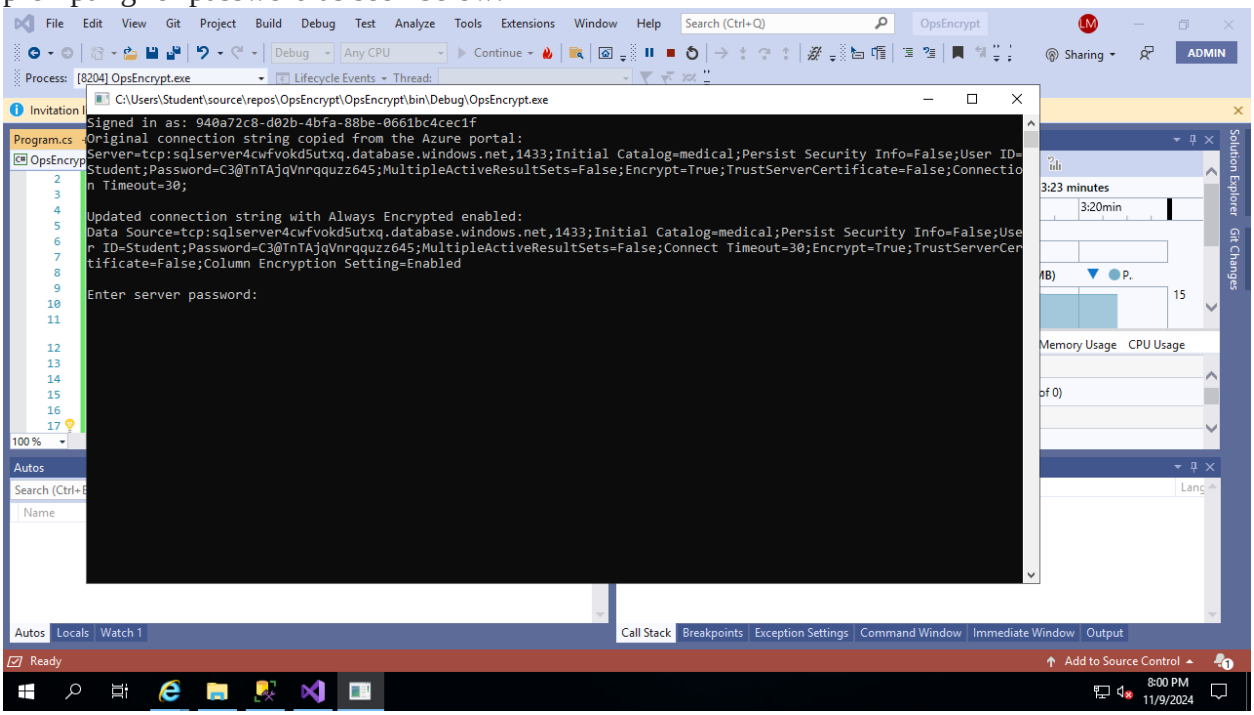


13. In the Visual Studio window, in the **Program.cs** pane, in line 17, replace the <key value noted earlier> placeholder with the the value of **Key1** of the registered app you recorded

earlier in the lab. We have replaced the Key Value 1

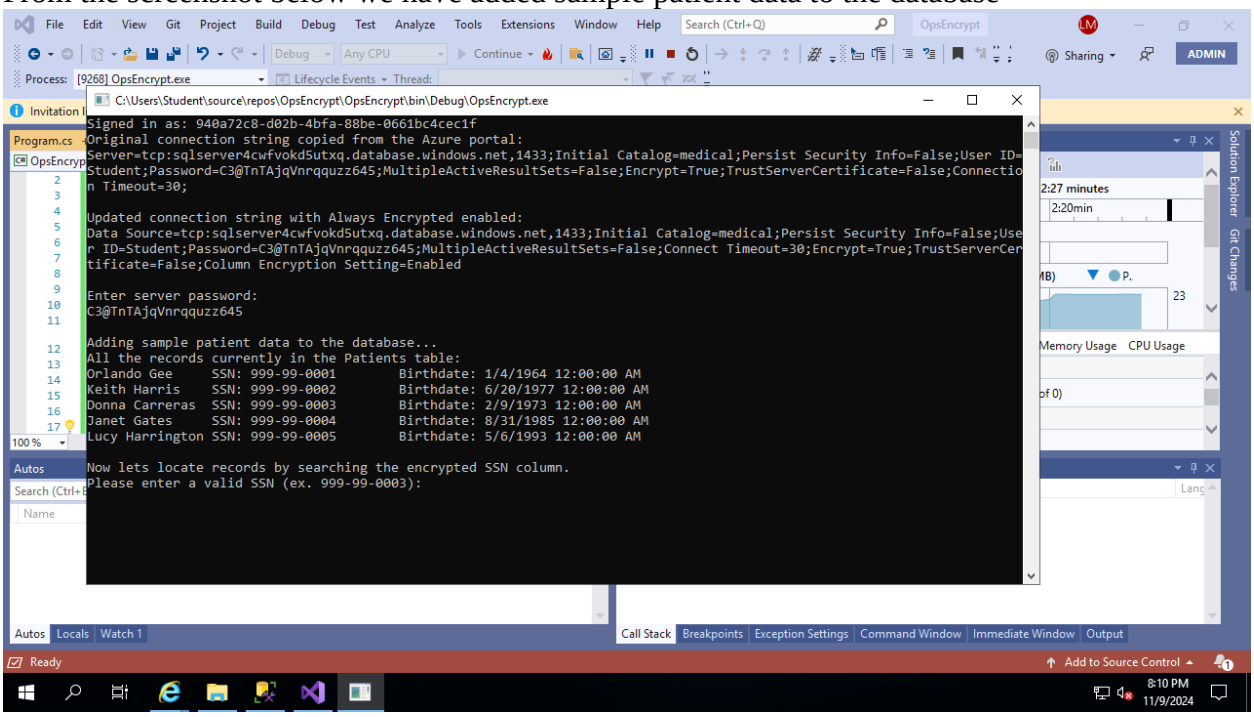


14. In the Visual Studio console, click the **Start** button to initiate the build of the console application and start it. After pressing the start button we get a command prompt window prompting for password as seen below.

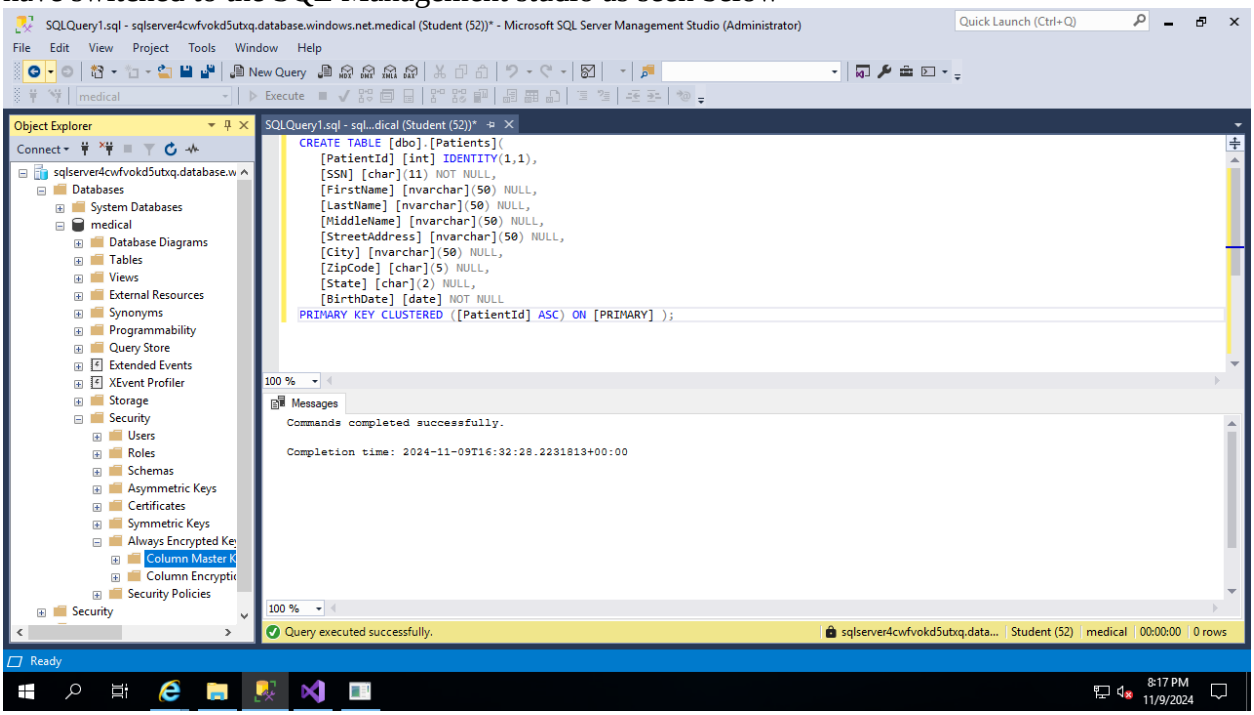


15. The application will start a Command Prompt window. When prompted for password, type the password that you specified in the deployment in Exercise 1 to connect to Azure SQL Database.

From the screenshot below we have added sample patient data to the database

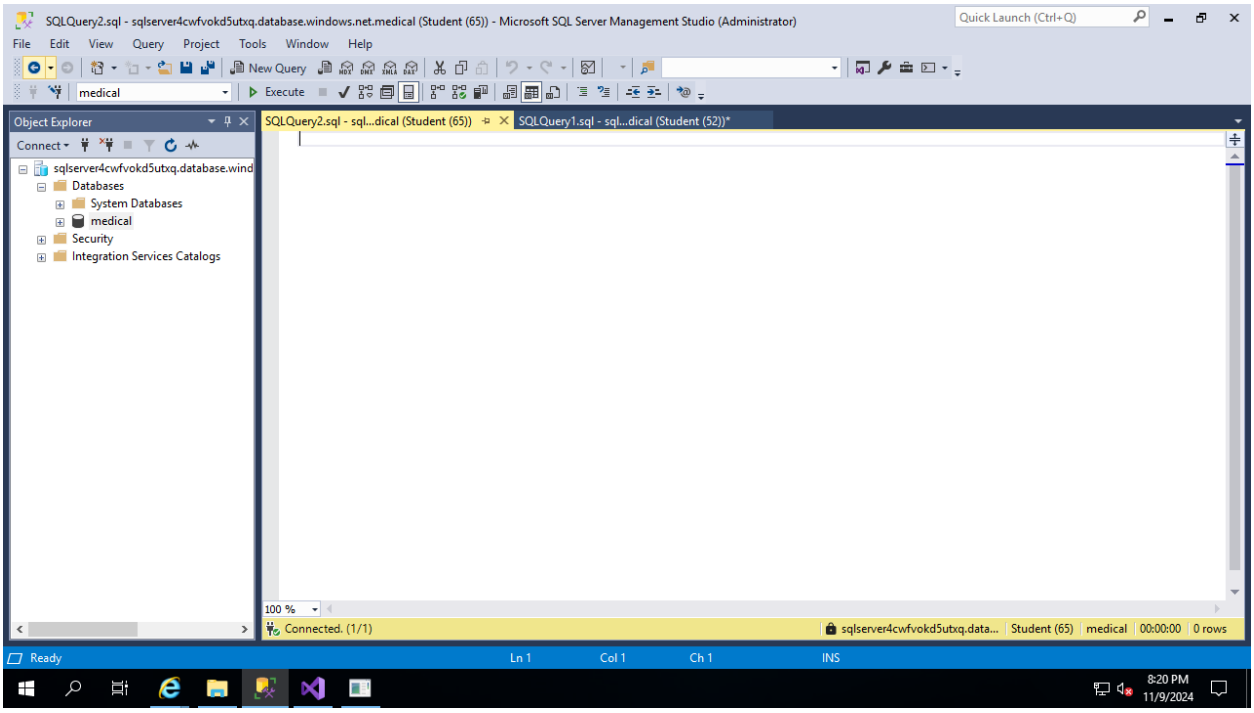


16. Leave the console app running and switch to the **SQL Management Studio** console. We have switched to the SQL Management studio as seen below



17. In the **Object Explorer** pane, right-click the **medical database** and, in the right-click menu, click **New Query**

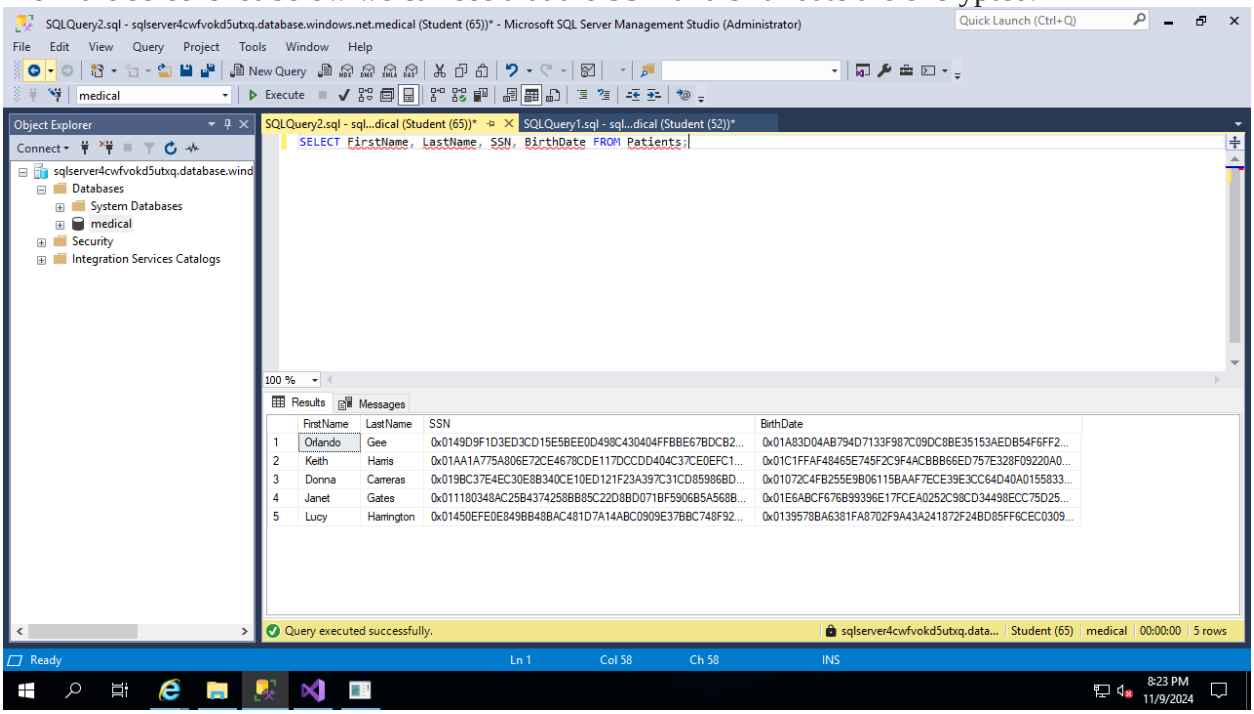
Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)



18. From the query window, run the following query to verify that the data that loaded into the database from the console app is encrypted.

SELECT FirstName, LastName, SSN, BirthDate FROM Patients;

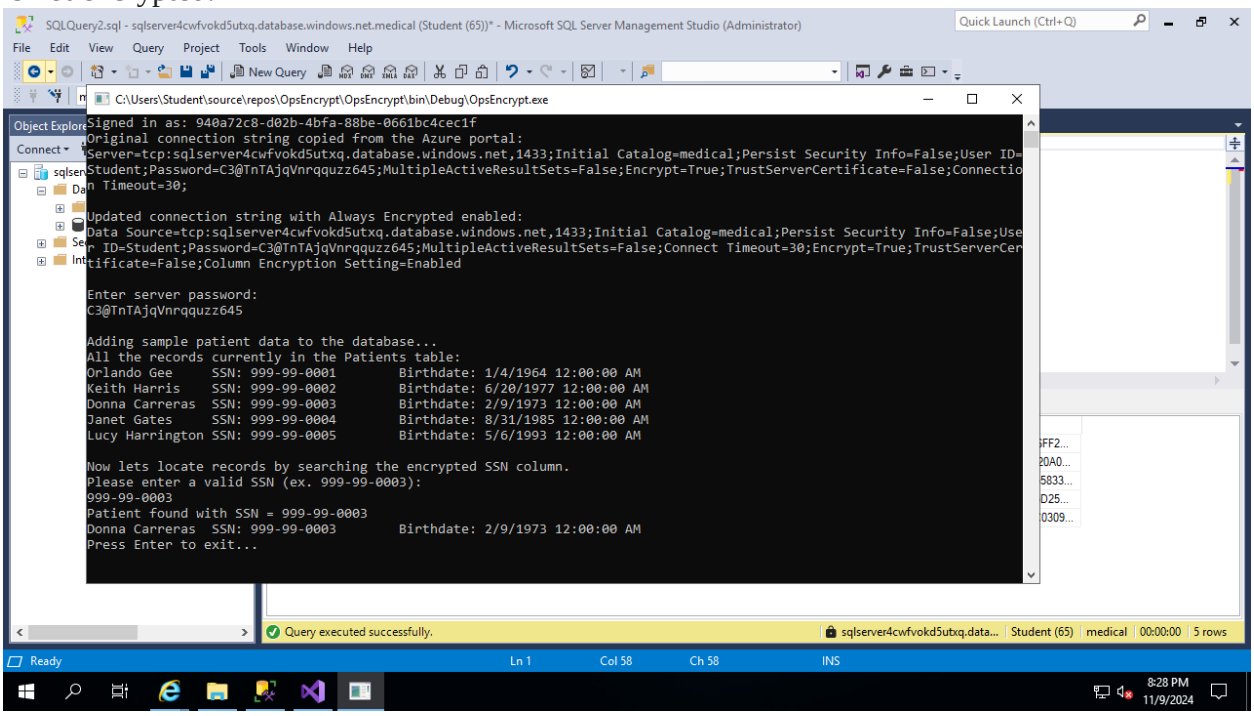
From the screenshot below we can see that the SSN and birth date are encrypted.



19. Switch back to the console application where you are prompted to enter a valid SSN. This will query the encrypted column for the data. At the Command Prompt, type the following and press the Enter key:

    999-99-0003
    Note: Verify that the data returned by the query is not encrypted.

    From the screenshot below we can see the data pulled for patient with SSN 999-99-0003 is not encrypted.



20. To terminate the console app, press the Enter key

    Press enter to exit

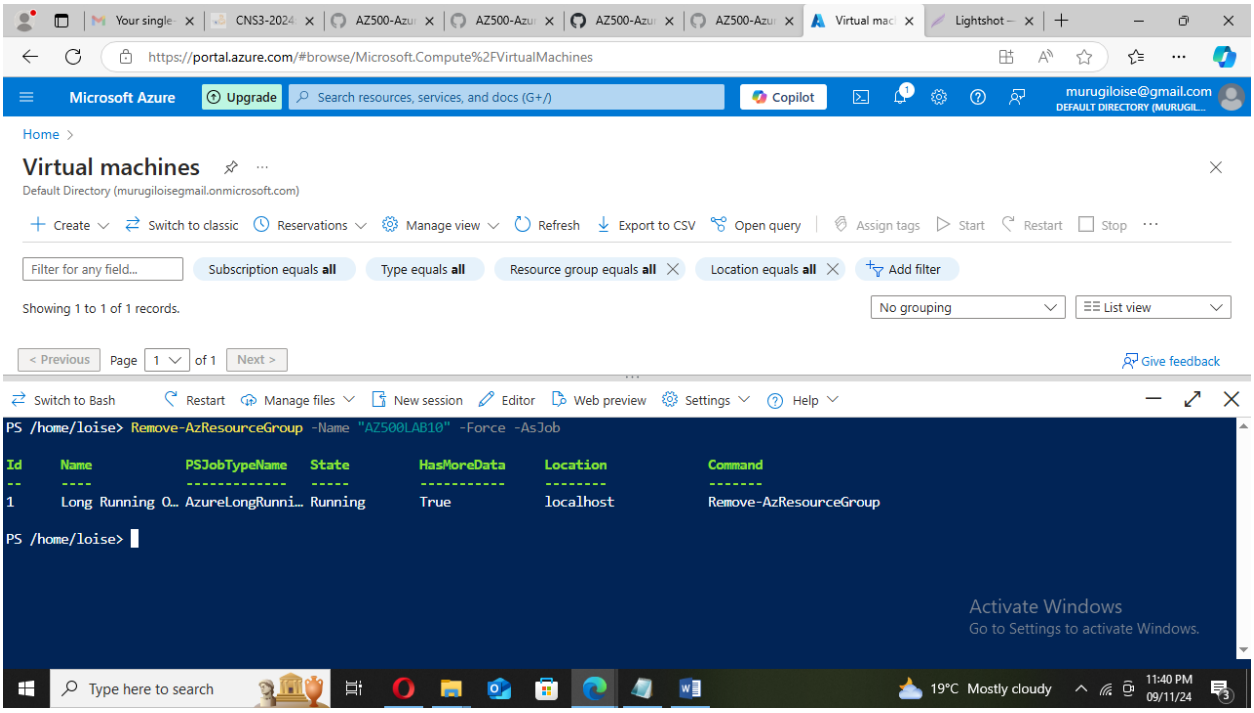
Clean up resources

1. In the Azure portal, open the Cloud Shell by clicking the first icon in the top right of the Azure portal.
2. In the upper-left drop-down menu of the Cloud Shell pane, if needed select **PowerShell** and, when prompted, click **Confirm**.
3. In the PowerShell session within the Cloud Shell pane, run the following to remove the resource groups you created in this lab:

    Remove-AzResourceGroup -Name "AZ500LAB10" -Force –AsJob

    We have successfully removed the AZ500LAB10 resources as seen below.

Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)



Conclusion

In this exercise, we have deployed an Azure VM and an Azure SQL database setting up Azure subscription 1, with a resource group AZ500LAB10 with West US as the region. We have then created a Key Vault resource in PowerShell then set access policies, then set the key as MyLabKey and add a secret to the Key Vault. We have registered an application sqlApp with an Application ID and secret.

We then then logged in to the Azure VM, the deployment we initiated in earlier in the activity. The Azure VM hosts Visual Studio 2019 and SQL Server Management Studio 19 and updated Azure SQL database named medical with a new table structure and selected data columns for encryption. Then set the server firewall with a rule name Allow Mgmt VM with the Public IP Address of the az500-10-vm1. Then Download RDP File and use it to connect to the az500-10-vm1 Azure VM via Remote Desktop.

We then installed Install SQL Server Management Studio on az500-10-vm1. Azure VM where we created Patients table and encrypted columns for dbo.Patients. We then created a Console application using Visual Studio to load data into the encrypted columns and then access that data securely using a connection string that accesses the key in the Key Vault where the SSN and Birtdate were encrypted and from the console when we enter the SSN to query the encrypted column for the data.

Week 9 Assignment 1: Key Vault (Implementing Secure Data by setting up Always Encrypted)