

Manage Azure Key Vault

(LAB-204-13-01)

Lab scenario

You have been asked to create a proof-of-concept application that makes use of the Azure SQL Database support for Always Encrypted functionality. All of the secrets and keys used in this scenario should be stored in the key vault. The application should be registered in Azure Active Directory (Azure AD) in order to enhance its security posture. To accomplish these objectives, the proof of concept should include:

- Creating an Azure key vault and storing keys and secrets in the vault.
- Create a SQL Database and encrypting content of columns in database tables by using Always Encrypted.

Lab exercises

- Configure the key vault with a key and a secret.
- Create an application to demonstrate using the key vault for encryption.

Task 1: Provision Resources

Step 1: Review the ARM Template

1. Open the **az-204-13-01_azuredeploy.json** and **az204-13-01_azuredeploy.parameters.json** file in notepad and review its content.

Note: **JSON** template provided with the lab manual.

Step 2: Deploy the Resources

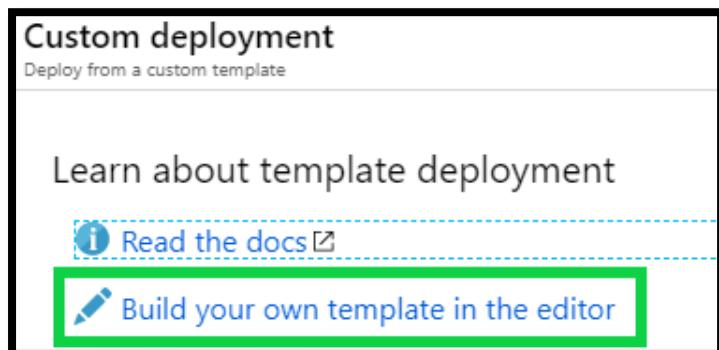
2. From the Azure Portal, go to the left menu, select **Create a resource**
3. Search and Select **Template deployment**



4. Select **Create**



5. Select **Build your own template in the editor**

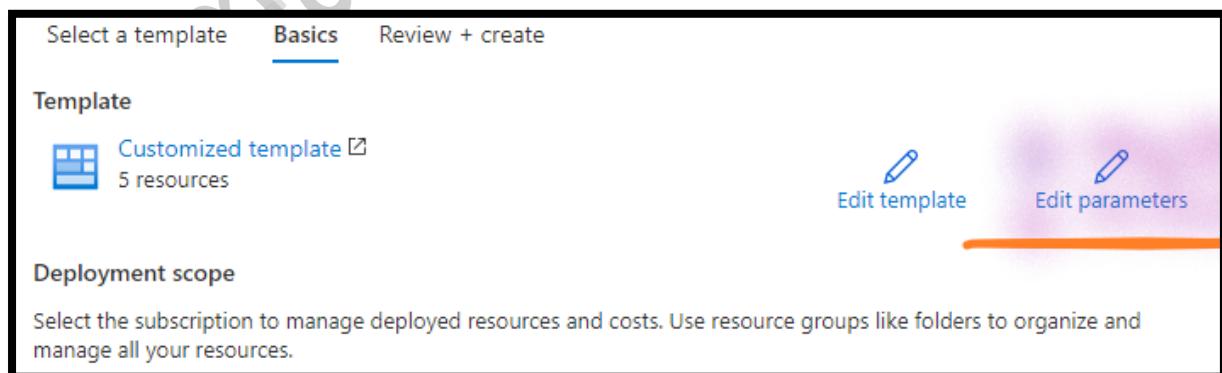


6. On the **Edit template** blade, click **Load file** and **select** and upload the **az-204-13-01_azuredeploy.json** file.

Note: **JSON** template provided with the lab manual.

7. Click **Save**

8. It will open the **Custom deployment** blade, click on the **Edit parameters** blade.



9. Click **Load file** and **select** and upload the **az204-13-01_azuredeploy.parameters.json** file.

10. It will open the **Custom deployment** blade. Provide the following **details**:

- i. **Subscription:** Select your **Default subscription**
- ii. **Resource Group:** Select **Create new** and provide resource group name **Az-204-13-01-RG**.
- iii. **Region:** Dropdown Select **East US**

Note: Leave other details as default.

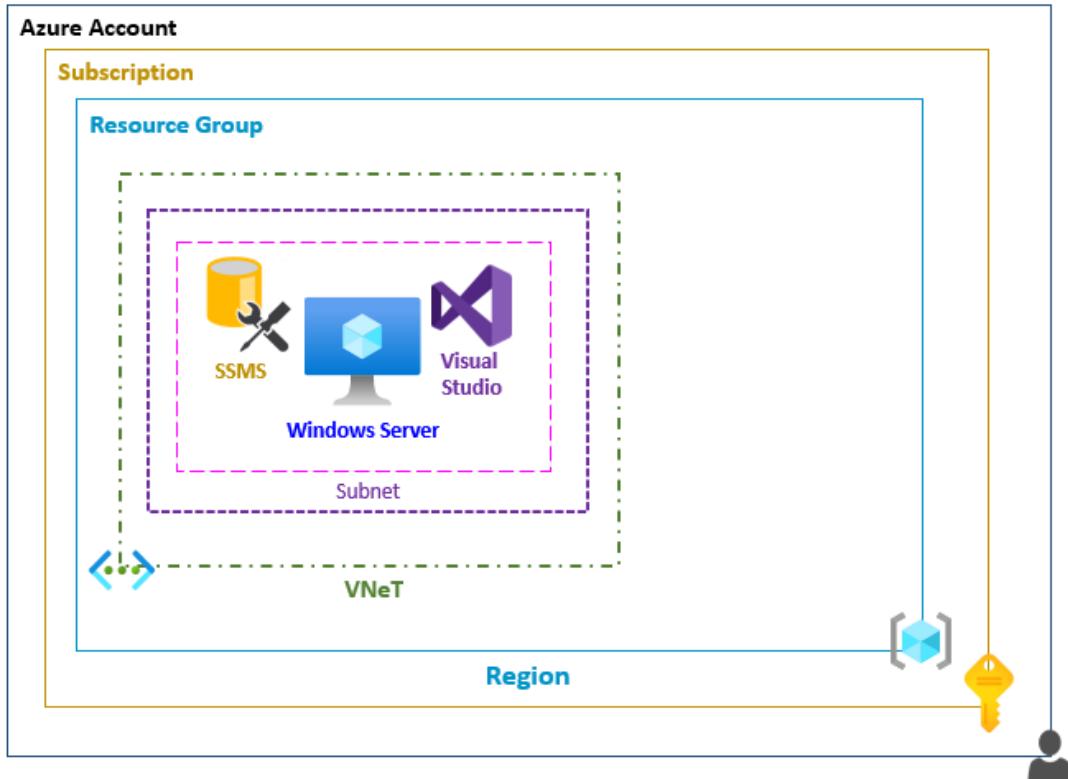
11. Select **Next: review + Create**

12. Select **Create**

Note: Wait for deployment completion. You can verify this from Notification icon.

Task 2: Install Tools

In this task, you will deploy an Azure VM, connect to it, and download and install Visual Studio 2019 and SQL Server Management Studio (SSMS).



Step 1: Connect to Virtual Machine

13. Go to the left side of the menu, select **virtual machines**.
14. Select & open the virtual machine **az204-13-vm** from the list.
15. On the right side of the page copy **Public IP Address**.
16. In the local Desktop/ Laptop (Windows 10), right click on **Start** & **Run**.
17. In the open, write **mstsc**.
18. Enter in the **Public IP Address** of the Azure virtual machine, and then click **Connect**.



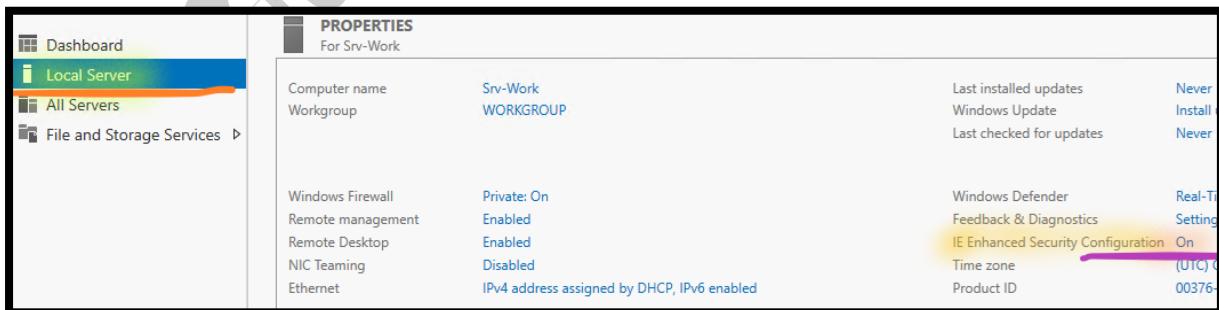
Info: When prompt for username & password, provide below details.

Username: **master**

Password: **Lab@password**

Step 2: Connect to Work Server

19. From the **az204-13-vm** server (Windows 2019), right click on **Start** & **Run**.
20. In the open, write **servermanager.exe**.
 - a. Click **Local Server**.
 - b. Click **IE Enhanced Security Configuration**.



- c. Click **On**, next to **IE Enhanced Security Configuration**.
- d. Set both options to **Off**.



e. Click **Ok**.

Step 3: Install SSMS and Visual Studio

21. Download and Install **SQL Server Management Studio** (SSMS) using the below URL.

<https://docs.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver15>

22. Install **Visual Studio** using the below URL.

<https://visualstudio.microsoft.com/downloads/>

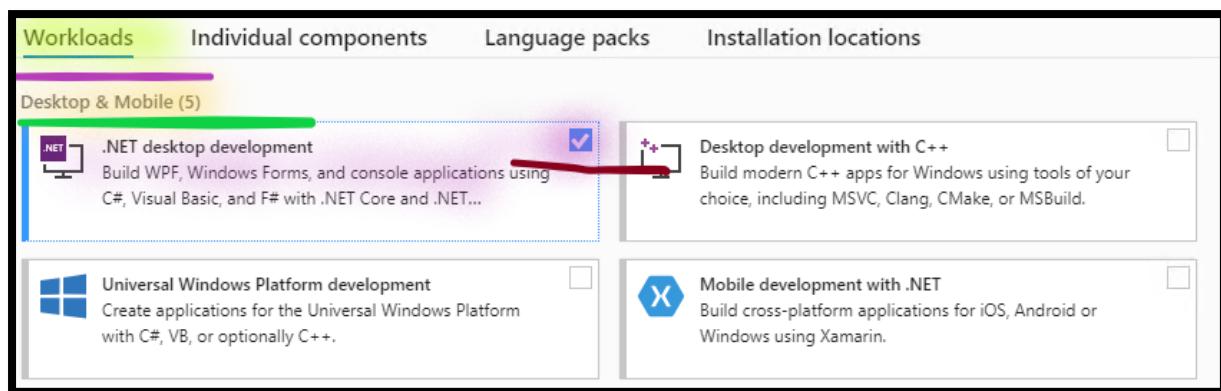
a. Select **Community edition** and **download**

Visual Studio 2019
Version 16.7
[Release notes >](#)
Full-featured integrated development environment (IDE) for Android, iOS, Windows, web, and cloud
[Compare editions >](#)
[How to install offline >](#)

Community
Powerful IDE, free for students, open-source contributors, and individuals

Free download ↓

b. In the **Workloads** window, in the **Desktop & Mobile** section, **select** the **.NET desktop development** checkbox

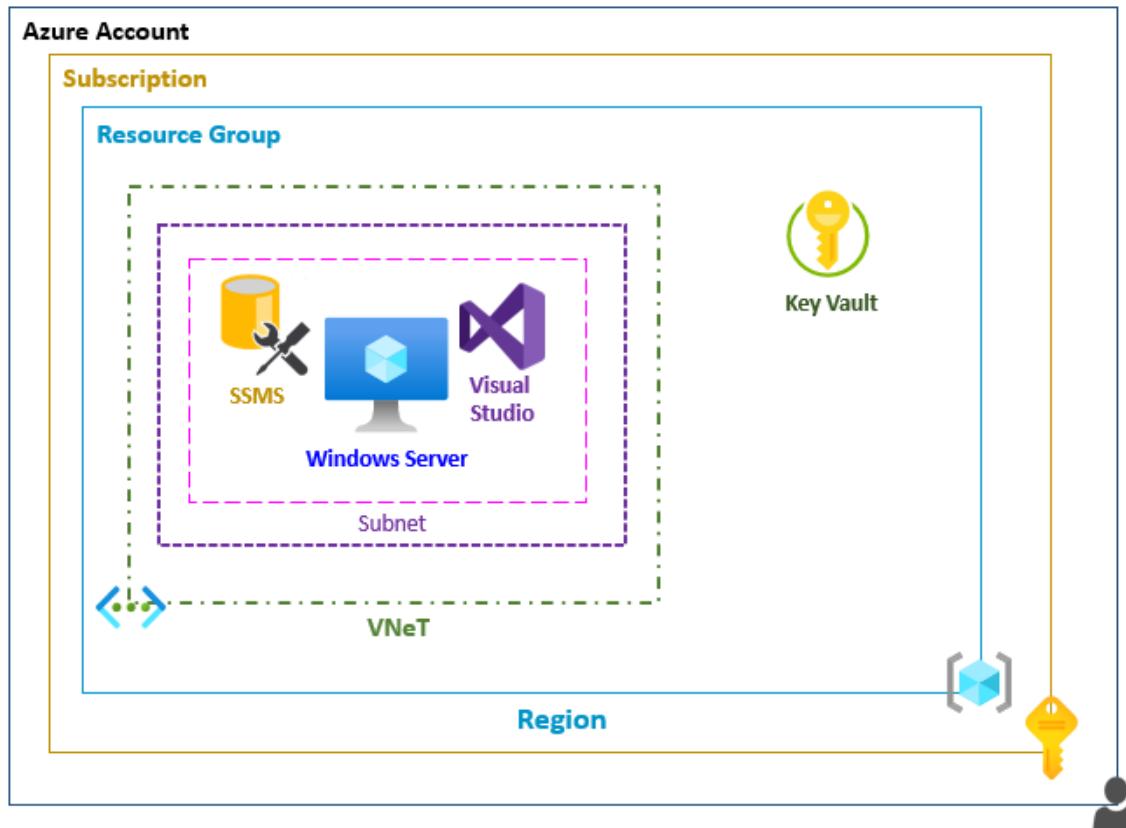


c. Click **Install**.

Note: Visual studio installation takes ~ 15 mnts. **Don't wait**, goto the next step.

Task 3: Create Azure Key Vault

In this task, you will create a lab resource group and a key vault. You will also configure the key vault permissions.



Step 1: Create Azure Key Vault

23. From the Azure Portal, **go to the left menu**, select **Create a resource**.

24. Search and Select **Key Vault** from the list

25. Select **Create** and **configure**:

a. **Subscription:** Select **default subscription**

b. **Resource group:** Select **existing** resource group **Az-204-13-01-RG**

c. **Key Vault name:** Provide key vault name **lab204keyvault123**

Note: Replace 123 to make unique name.

d. **Region:** Select region **East US**

e. **Pricing tier:** Select **Standard**

Note: Leave other options as default.

Subscription *	Visual Studio Enterprise
Resource group *	RG-304-MySQL-R1
Instance details	
Key vault name *	lab304keyvault123
Region *	(US) East US

f. Select **Next: Access policy**

Note: Leave other options as default.

g. Select **Next: Networking**

Note: Leave other options as default.

h. Select **Next: Tags**

i. Select **Next: Review + Create**

j. Select **Create**

Note: **Wait** (~5 mnts.) till deployment gets completed.

The screenshot shows a deployment summary. At the top, a green checkmark indicates "Your deployment is complete". Below it, deployment details are listed: name (lab304keyvault1234), subscription (Visual Studio Enterprise), and resource group (RG-304-MySQL-R12). The start time is 4/15/2020, 10:39:22 PM, and the correlation ID is a889e0ff-58be-4bbe-bbc6-a84f15975195. A section titled "Deployment details" includes a download link. A table below shows the deployment status: one item, "lab304keyvault1234", is listed under Resource, Type (Microsoft.KeyVault/vaults), Status (OK), and Operation details (link).

Resource	Type	Status	Operation details
lab304keyvault1234	Microsoft.KeyVault/vaults	OK	Operation details

Step 2: Configure Access Policy

26. From the **Azure Portal**, go to the left menu, select **resource group**

27. Open resource group **Az-204-13-01-RG**

28. Open Key Vault **lab204keyvault123**

29. Select **Access policies** under **settings**

30. Select **Add access policy** and **Configure**:

The screenshot shows the "Access policies" section of the Key Vault settings. On the left, a sidebar lists "Tags", "Diagnose and solve problems", "Settings" (with "Keys", "Secrets", "Certificates", and "Access policies" highlighted in blue), and "Networking". The main area shows "Enable Access to:" with three checkboxes for "Azure Virtual Machines for deployment", "Azure Resource Manager for template deployment", and "Azure Disk Encryption for volume encryption". Under "Permission model", there are two radio buttons: "Vault access policy" (selected) and "Azure role-based access control (preview)". A green button "+ Add Access Policy" is at the bottom, and a link "Current Access Policies" is at the very bottom.

a. **Configure from template:** Dropdown and select **Key, Secret, & Certificate Management**

b. **Key permissions:** Dropdown and select **Select All**

Note: You can see the **16 permissions** selected.

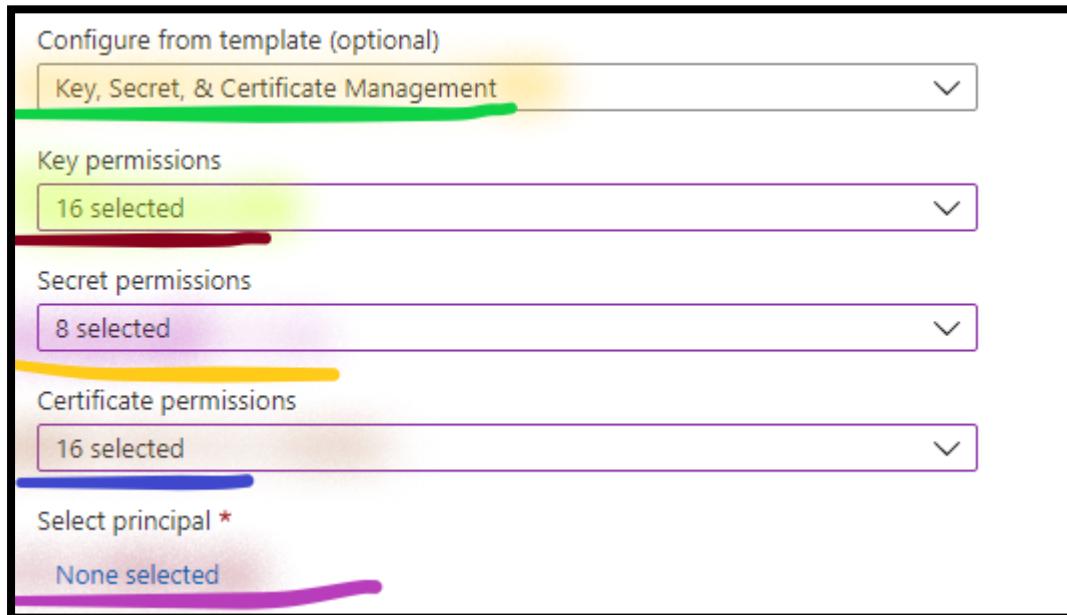
c. **Secret permissions:** Dropdown and select **Select All**

Note: You can see the **8 permissions** selected.

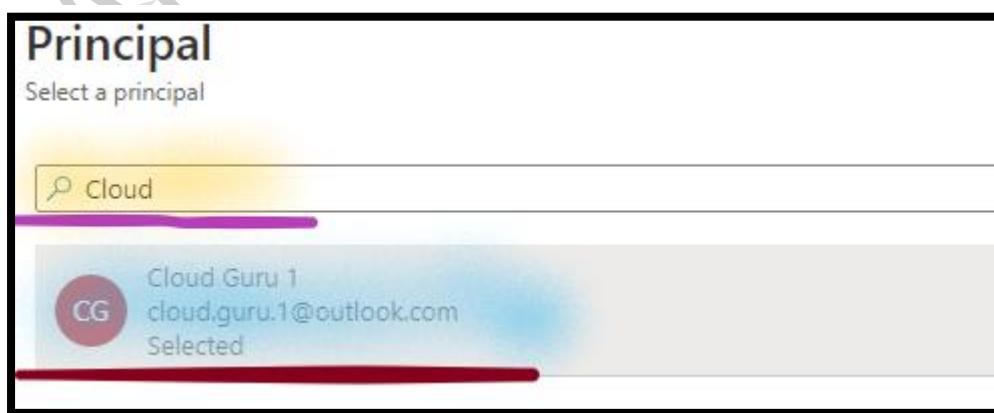
d. **Certificate permissions:** Dropdown and select **Select All**

Note: You can see the **16 permissions** selected.

e. **Select principal:** Click **None selected**



i. On the **Principal** blade, search and select **root user account**



ii. Click **Select**

f. Select **Add**

31. Select **Save**

Step 3: Create Keys

In this task, you will add a key to the key vault to know how to add the key in Key vault. This is not going to use in C# code.

32. Select **Keys** under **settings**

33. Select **Generate/ Import** and **Configure:**

The screenshot shows the 'lab500keyvault123 | Keys' page in the Azure portal. The top navigation bar includes a search bar, 'Generate/Import' (which is highlighted in blue), 'Refresh', and 'Restore Backup'. On the left, there's a sidebar with 'Settings' (selected) and three tabs: 'Keys' (highlighted in green), 'Secrets', and 'Certificates'. The main content area has a table with columns 'Name' and 'Status'. A message at the top of the table says 'There are no keys available.'

a. **Options:** Dropdown & Select **Generate**

b. **Name:** Write **Az204LabKey**

Note: Leave other options as default.

The screenshot shows the 'Options' section of a Key Vault configuration page. It includes fields for 'Name' (az500labkey), 'Key Type' (RSA selected), 'RSA Key Size' (2048 selected), and activation/expiration date checkboxes. An 'Enabled?' switch is set to 'Yes'.

Options	Generate
Name *	az500labkey
Key Type	RSA EC
RSA Key Size	2048 3072 4096
Set activation date?	<input type="checkbox"/>
Set expiration date?	<input type="checkbox"/>
Enabled?	Yes No

c. Select **Create**

Step 4: Create Secrets

In this task, you will add a secret to the key vault to know how to add the Secret in Key vault. This is not going to use C# code.

34. Select **Secrets** under **settings**

35. Select **Generate/ Import** and **Configure**

a. **Options:** Dropdown & Select **Manual**

b. **Name:** Write **SQLPassword**

c. **Value:** Write **Pa55w.rd1234**

Note: Leave other options as default.

Upload options

Manual

Name * ⓘ SQLPassword

Value * ⓘ
Content type (optional)

Set activation date? ⓘ

Set expiration date? ⓘ

Enabled? Yes No

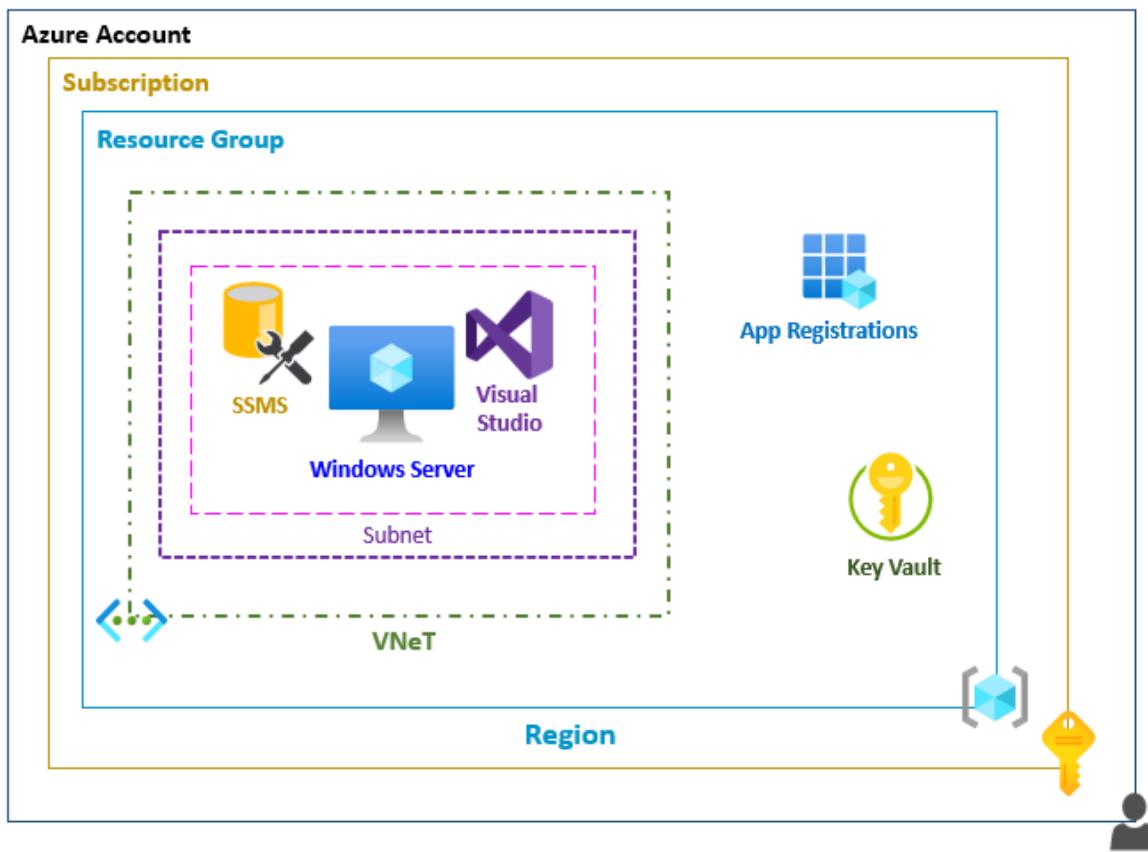


d. Select **Create**

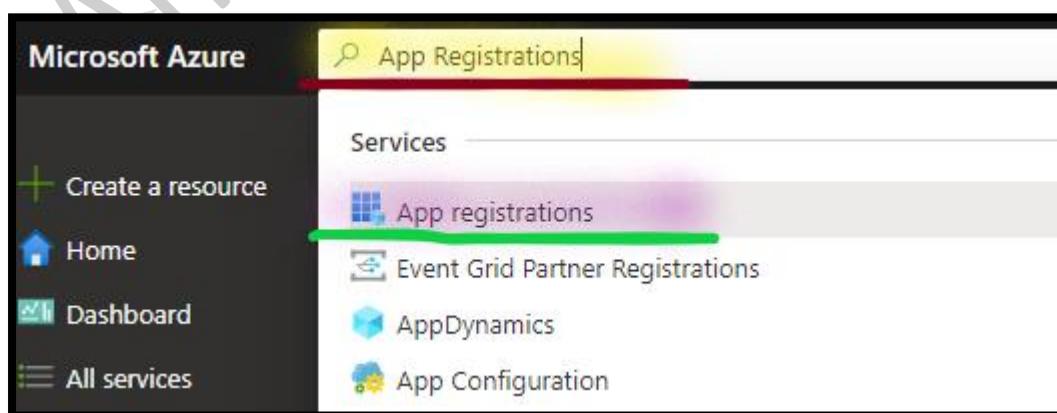
Task 4: Create an Application to use the Key Vault for Encryption

Step 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.



- 36.In the Azure portal, in the Search resources, services, and docs text box at the top of the Azure portal page, type **App Registrations**.



37. Select **App registrations**

38. Click **New registration** and **Configure**:

All applications Owned applications Applications from personal account

a. **Name:** Write **SQLApp**

b. **Redirect URI:** Dropdown and select **Web** and write <https://sqlapp>

Redirect URI (optional)
We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is changed later, but a value is required for most authentication scenarios.

Web https://sqlapp

c. Click **Register**.

39. On the **SQLApp** blade, Select **Overview**, identify the value of **Application (client) ID** and copy this in **Notepad**.

SQLApp

Search (Ctrl+ /)

Overview Delete Endpoints Preview features

Got a second? We would love your feedback on Microsoft identity platform (previously Azure AD for developer). →

Essentials

Display name	Supported account types
SQLApp	My organization only
Application (client) ID	Redirect URIs
72a38622-fa19-4772-bd49-45435800ce50	1 web, 0 spa, 0 public client
Directory (tenant) ID	Application ID URI
f593bc12-4ab1-4070-bd43-2ed2af36efee	Add an Application ID URI
Object ID	Managed application in local directory
f644b475-cd4e-4108-bace-d955466453ef	SQLApp

40.Click **Certificates and Secrets** under **Manage**

41.Click **New client secret** and **Configure**:

The screenshot shows the Azure portal's application configuration interface. On the left, a sidebar lists various management options like Overview, Quickstart, Integration assistant, etc. Under the 'Manage' section, 'Certificates & secrets' is highlighted with a red bar at the bottom. The main content area has two tabs: 'Certificates' (selected) and 'Client secrets'. Under 'Certificates', it says 'No certificates have been added for this application.' Under 'Client secrets', it defines what a client secret is and provides a 'New client secret' button.

a. **Description:** Write **Key1**

b. **Expires:** Select **1 year**

This is a modal dialog titled 'Add a client secret'. It has two fields: 'Description' containing 'Key1' and 'Expires' with 'In 1 year' selected from a radio button group. Other options in the group are 'In 2 years' and 'Never'.

c. Click **Add**

42.On the **SQLApp** blade, Select **Certificates & secrets** blade, identify the **value** of **Key1** and copy this in **Notepad**.

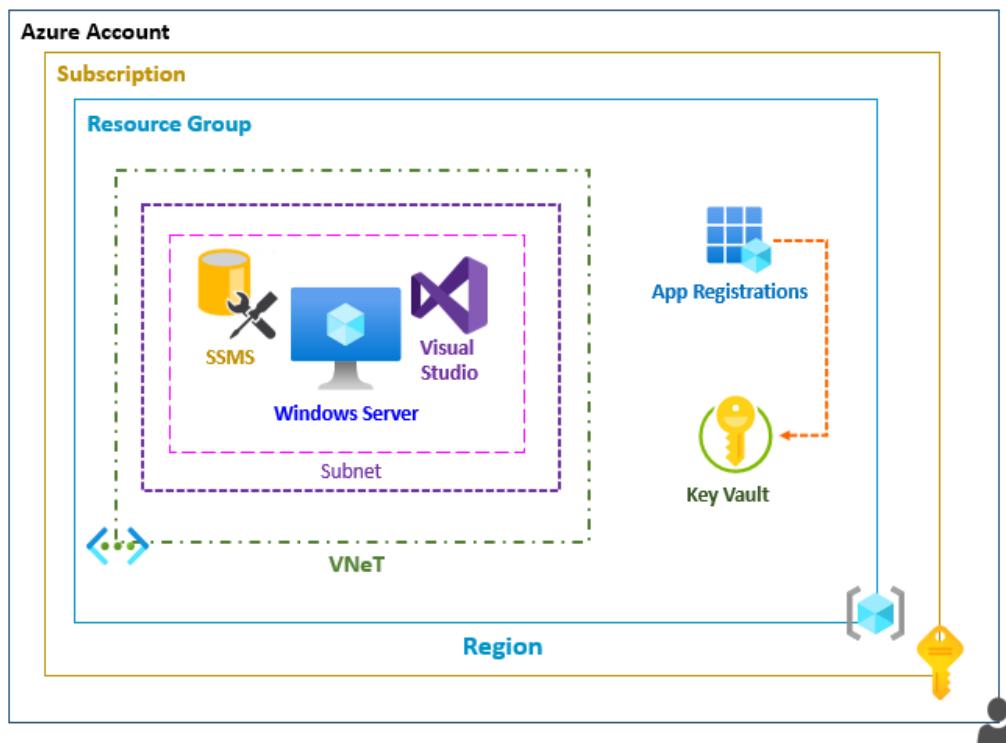
Client secrets

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.

+ New client secret

Description	Expires	Value
Key1	10/12/2021	Uw6pi6CB~~Bm549eJWGEhw2syjP0Q~o~u

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



43. From the **Azure Portal**, go to the left menu, select **resource group**

44. Open resource group **Az-204-13-01-RG**

45. Open Key Vault **lab204keyvault123**

46. Select **Access policies** under **settings**

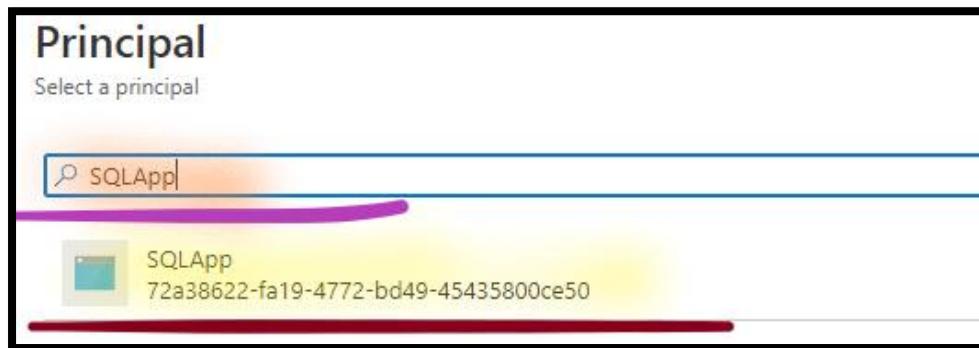
47. Select **Add access policy** and **Configure**:

- a. **Key permissions:** Dropdown and select **get, list, unwrapKey, wrapKey, verify, sign**.

Note: You can see the **6 permissions** selected.

b. **Select principal:** Click **None selected**.

i. On the **Principal** blade, search and select **SQLApp**.



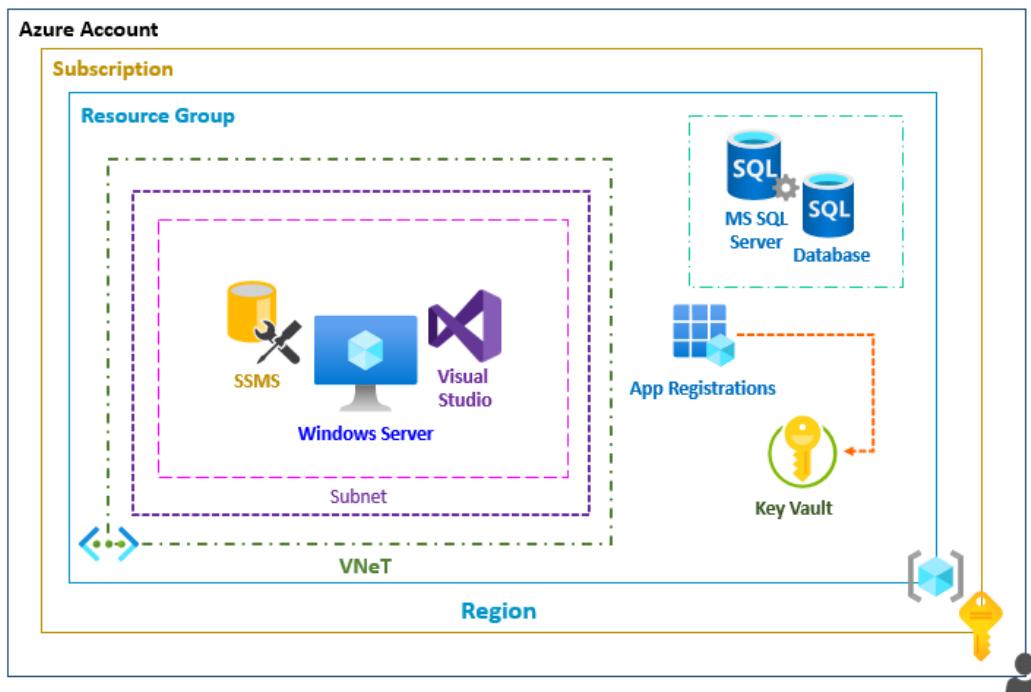
ii. Click **Select**

c. Select **Add**

48. Select **Save**

Task 5: Create Azure SQL Database

Step 1: Create Azure SQL Logical Server with SQL Database



49.Click the **Create resource** link in the left-hand navigation bar.

50.Search and Select the **SQL Database**

51.Click **Create**, & **configure**:

- a. **Subscription:** Select **default subscription**
- b. **Resource group:** Select **existing** resource group **Az-204-13-01-RG**
- c. **Database name:** Write **medical**

The screenshot shows the 'Create new' database configuration screen. Under 'Subscription', 'Azure Pass - Sponsorship' is selected. Under 'Resource group', 'Az-500-M03-01-RG' is selected, with a blue underline and a green checkmark indicating it's the active choice. In the 'Database details' section, the 'Database name' field contains 'medical'.

d. **Server:** Select **Create new** and **Configure**:

i. **Server name:** Provide server name **db-srv-123**

Note: Replace **123** to make the db server name unique.

- ii. **Server admin login:** Provide admin name **sqladmin**
- iii. **Password:** Provide password **Lab-Password**
- iv. **Confirm password:** Provide password **Lab-Password**
- v. **Location:** Select **East US**

Server name *

db-srv-123 .database.windows.net

Server admin login *

sqladmin

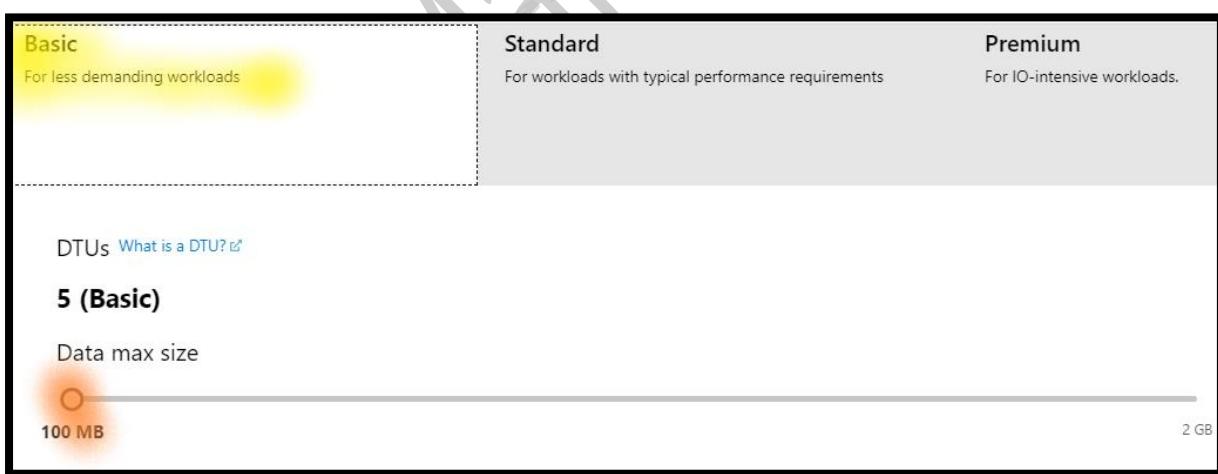
Password *

Confirm password *

Location *

(US) East US

- e. **Compute + Storage:** Click on **Configure database**
- Click on **Looking for basic, standard, premium**
 - Select **Basic**
 - Data max size:** Use slider to Select **minimum size**
 - Select **Apply**



- f. Click **Next: Networking**

Note: Leave all the options as default.

- g. Click **Next: Additional settings**

Note: Leave all the options as default.

h. Click **Next: Tags**

Note: Leave all the options as default.

i. Click **Next: Review + Create**

j. Click **Create**

Note: **Wait** until your deployment gets completed.

Step 2: Copy the Virtual Machine IP address

52. Go to the left side of the menu, select **virtual machines**.

53. Select & open the virtual machine **az204-13-vm** from the list.

54. On the right side of the page copy **Public IP Address**.

Step 3: Enable the Firewall Setting

55. From the Azure Portal, go to the left menu, select **resource group**

56. Open resource group **Az-204-13-01-RG**

57. Open SQL database **medical**

58. Select **Set server firewall**

a. **Rule name:** Write **az204-13-vm -rule**

b. **Start IP:** Write **Public IP Address az204-13-vm** virtual machine

c. **End IP:** Write **Public IP Address az204-13-vm** virtual machine

Client IP address	103.95.83.14	
Rule name	Start IP	End IP
az500-03-vm1-rule	52.147.201.169	52.147.201.169

d. Select **Save**

Step 4: Copy the Connection string

59. Select **Connection string** under **settings**

60. Select **ADO.NET**

61. Copy **ADO.NET Connection string** in the **NotePad**.



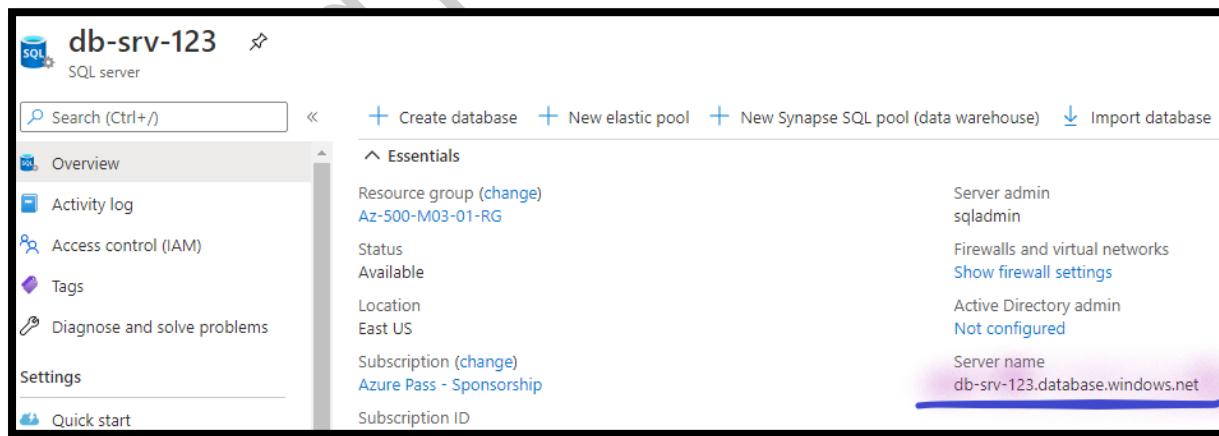
Step 5: Copy the SQL Server name

62. From the Azure Portal, go to the left menu, select **resource group**

63. Open resource group **Az-204-13-01-RG**

64. Open SQL server **db-srv-123**

65. Copy the **Server name** in the **NotePad**.



Step 6: Connect to Virtual Machine

- 66.In the local Desktop/ Laptop (Windows 10), right click on **Start & Run**.
67. In the open, write **mstsc**.
- 68.Enter in the **Public IP Address** of the Azure virtual machine, and then click **Connect**.
- 69.Enter the **Username** and **Password** of the Azure virtual machine and click **Ok**.

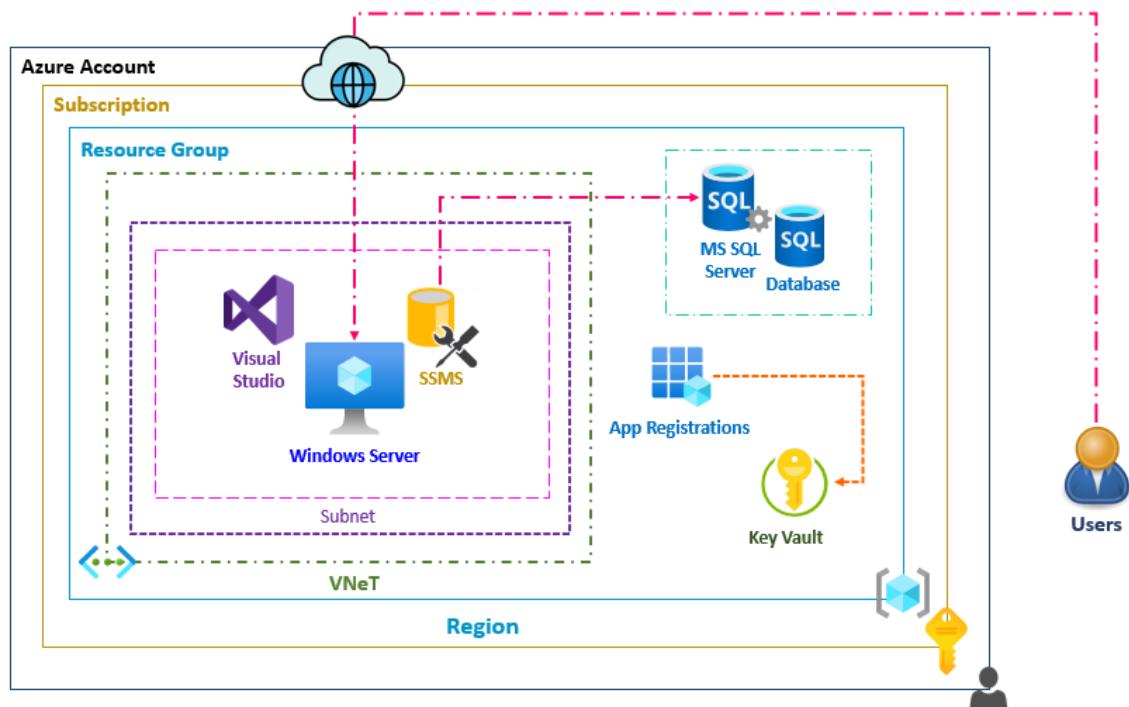


Info: When prompt for username & password, provide below details.

Username: **master**

Password: **Lab@password**

Step 7: Connect to SQL Server

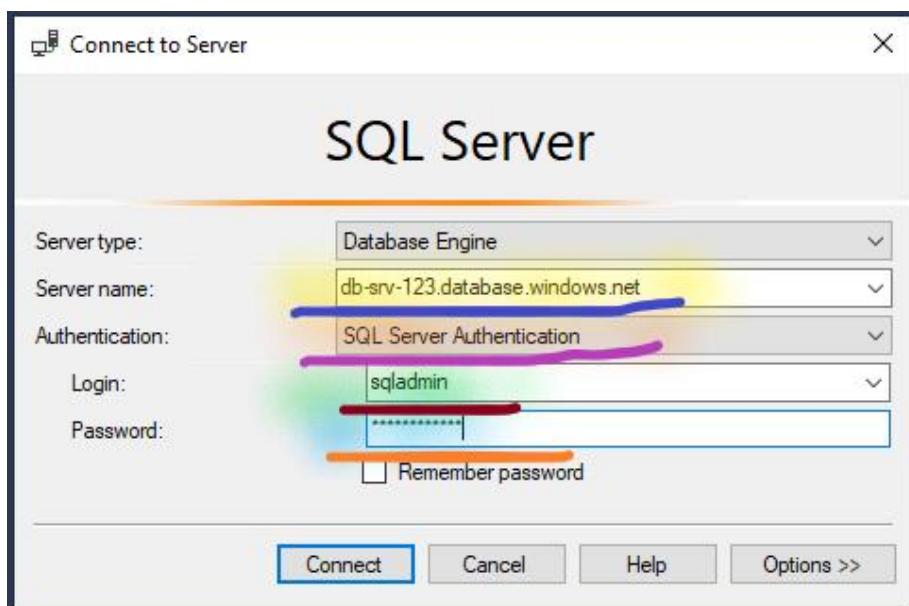


- 70.From the **az204-13-vm** virtual machine, Right click on **Start & Run**. In the open, **write**

**C:\Program Files (x86)\Microsoft SQL Server Management Studio
18\Common7\IDE\SSMS**

- a. **Server type:** Dropdown and Select **Database Engine**

- b. **Server name:** Copy the **db-srv-123 Server name**
- c. **Authentication:** Dropdown and Select **SQL server authentication**
 - i. **Server admin login:** Write **sqladmin**
 - ii. **Password:** Write **Lab-Password**



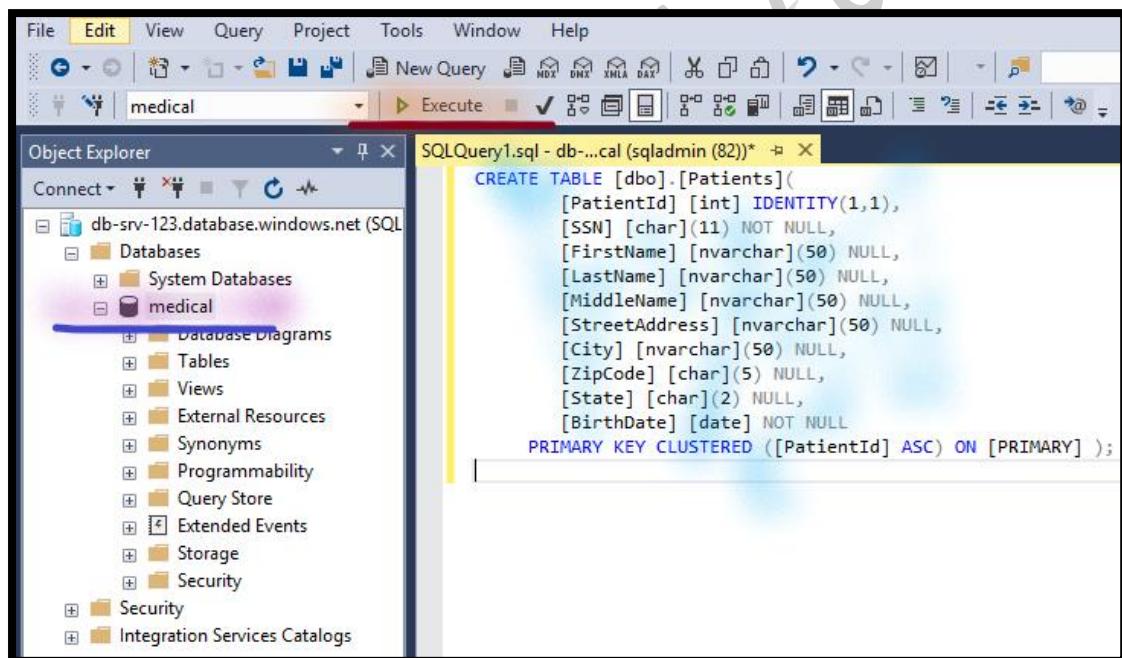
- d. Select **Connect**

Step 8: Create Table

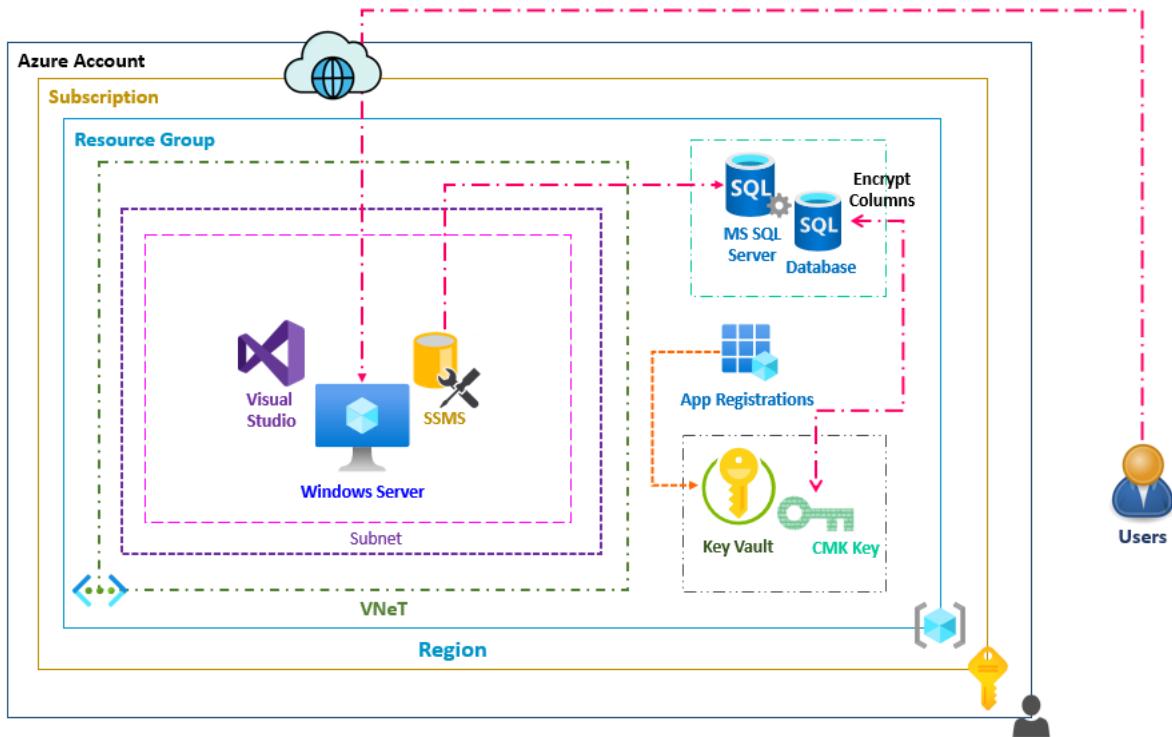
71. From the **SQL Management Studio (SSMS)**

- a. Expand **Databases**
- b. **Right click** on **medical** database
- c. Select **New Query**
- d. Paste the below **code** into the query window and click **Execute** to 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] );
```



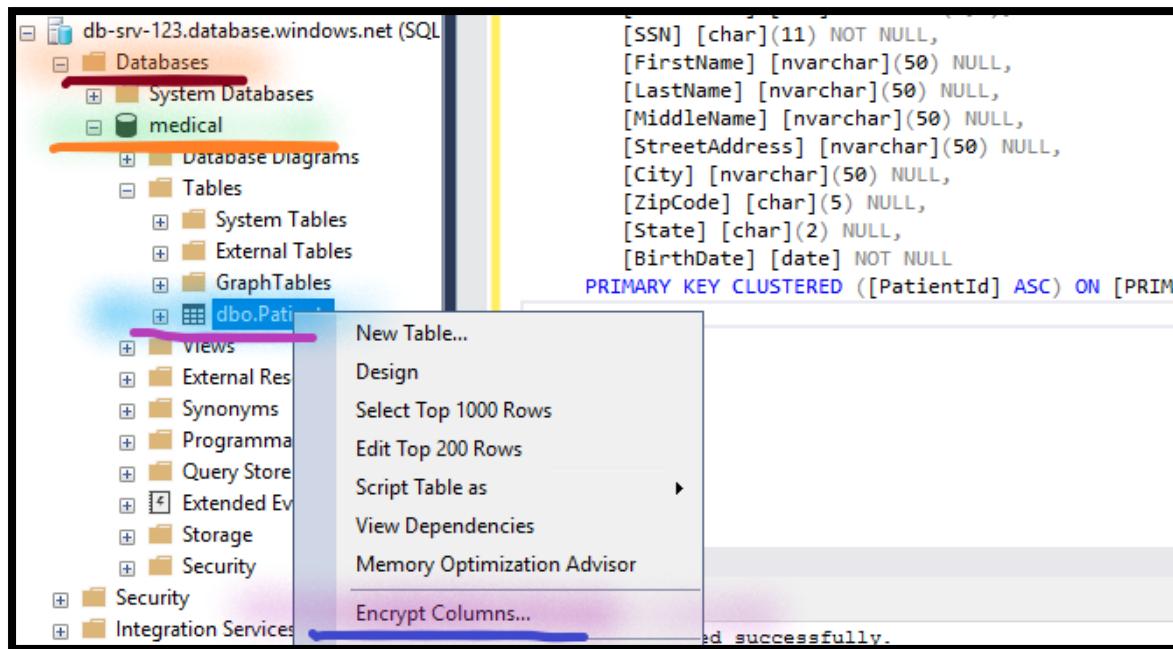
Step 9: Encrypt Columns



72. From the **SQL Management Studio (SSMS)**

- a. Expand **Databases**
- b. Select **medical** database
- c. Expand **Tables**
- d. **Right click** on **dbo.Patients** tables.
- e. Select **Encrypt columns**.

Note: This will initiate the **Always Encrypted** wizard displays.



- i. On the **Introduction page**, click **Next**.
- ii. On the **Column Selection page**:
 - **Name**: Select the **SSN**.
 - **Encryption type**: Dropdown and Select **Deterministic**.

Info: Deterministic Encryption always generates the same encrypted value for any given plaintext value.

- **Name**: Select the **Birthdate**.
- **Encryption type**: Dropdown and Select **Randomized**.

Info: Randomized encryption generates a different encrypted value for the same plaintext each time.

The screenshot shows the 'Column Selection' page. On the left, a sidebar lists 'Introduction', 'Column Selection' (which is selected), 'Master Key Configuration', 'Run Settings', 'Summary', and 'Results'. The main area has a search bar 'Search column name...' and a checkbox 'Apply one key to all checked columns: CEK_Auto1 (New)'. Below is a table with columns 'Name', 'State', 'Encryption Type', and 'Encryption Key'. The table shows the following rows:

Name	State	Encryption Type	Encryption Key
dbo.Patients			
PatientId			
SSN		Deterministic	CEK_Auto1 (New)
FirstName			
LastName			
MiddleName			
StreetAddress			
City			
ZipCode			
State			
BirthDate			

- Click **Next**.

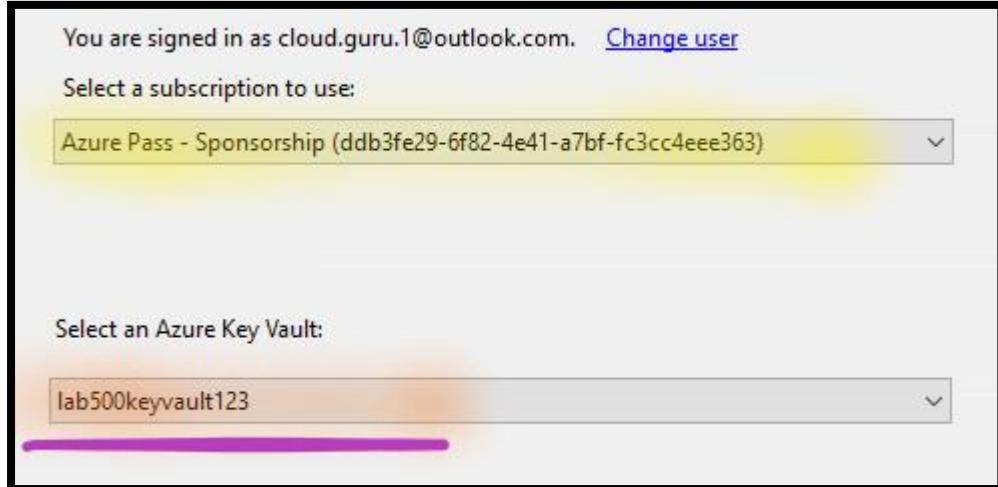
iii. On the **Master Key Configuration** page:

- Select **Azure Key Vault**.
- Click **Sign-in**.

The screenshot shows the 'Master Key Configuration' page. The sidebar includes 'Introduction', 'Column Selection' (selected), 'Master Key Configuration' (selected), 'Run Settings', 'Summary', and 'Results'. The main area contains a note: 'To generate a new column encryption key, a column master key must be selected to protect it. The column master key is stored outside of the database.' Below is a section 'Select column master key:' with a dropdown menu set to 'Auto generate column master key'. Under 'Select the key store provider', there are two options: 'Windows certificate store' (radio button unselected) and 'Azure Key Vault' (radio button selected). At the bottom, a message says 'You are not signed in to Microsoft Azure' with a 'Sign In...' button.

Note: When prompted, authenticate by using the **Azure Root account**.

- **Select a Subscription to use:** Dropdown and Select your default subscription.
- **Select an Azure Key Vault:** Dropdown and Select **lab500keyvault123** Key Vault.



- Click **Next**.

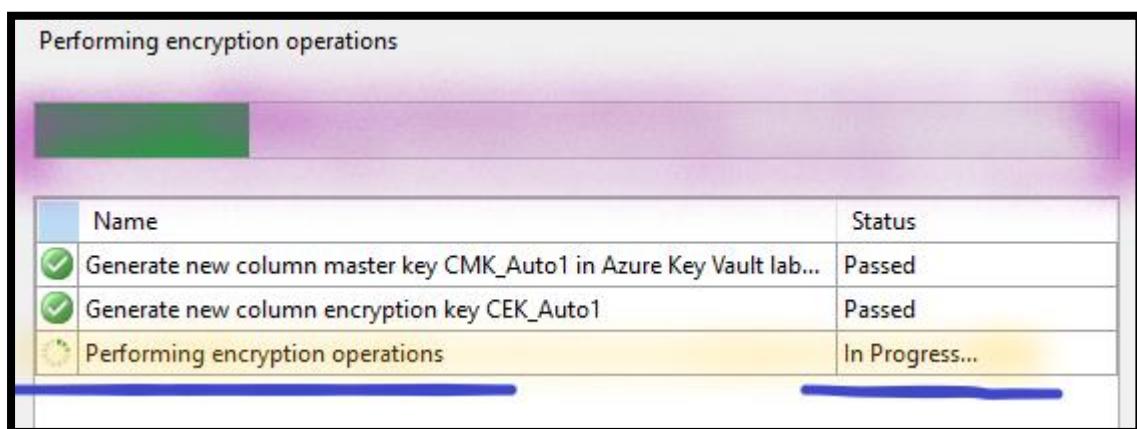
iv. On the **Run Settings page**:

- Click **Next**.

v. On the **Summary page**:

- Click **Finish**.

Note: To proceed with the encryption, when prompted, authenticate by using the **Azure Root account**.



Note: Wait till encryption process is complete.

Summary:	
Task	Details
Generate new column master key CMK_Auto1 in Azure Key Vault lab500keyva...	Passed
Generate new column encryption key CEK_Auto1	Passed
Performing encryption operations	Passed

vi. Click **Close**

Step 10: Verify the Encryption

73. From the **SQL Management Studio (SSMS)**

- Expand **Databases**
- Select **medical** database
- Expand **Security**
- Expand **Always encrypted keys**.

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

```

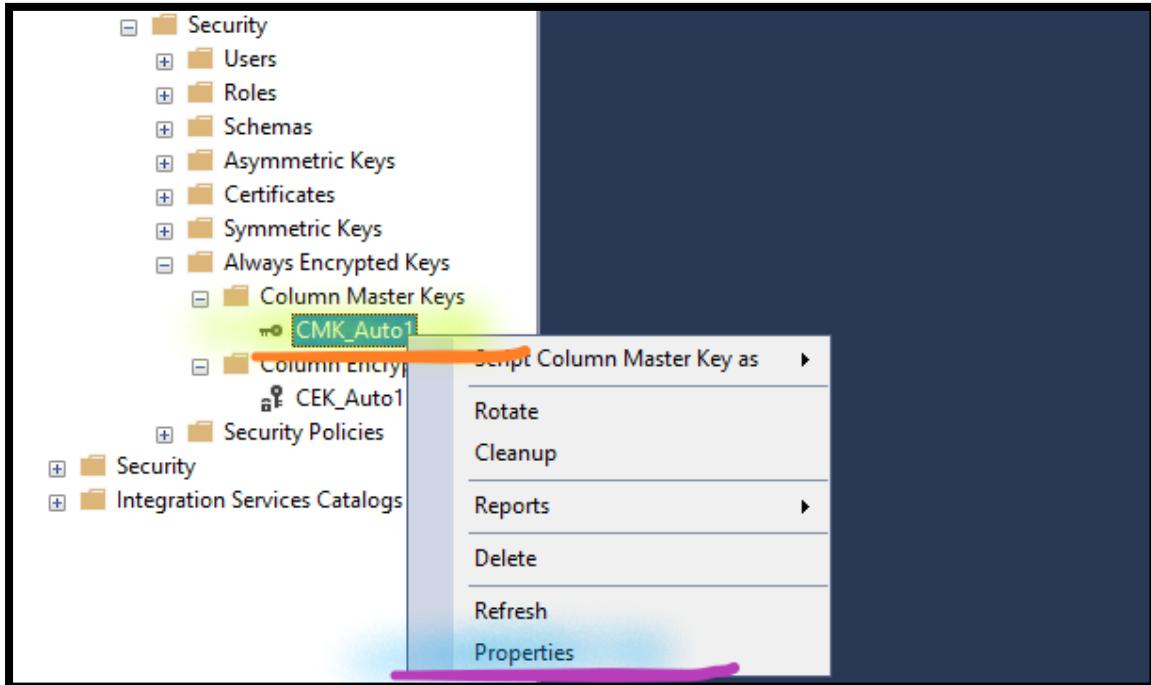
CREATE TABLE [dbo].[Patients]
[PatientId] [int] IDENTITY(1,1),
[SSN] [char](11) NOT NULL,
[FirstName] [nvarchar](50) NULL,
[LastName] [nvarchar](50) NULL,
[MiddleName] [nvarchar](50) NULL,
[StreetAddress] [nvarchar](50) NULL,
[City] [nvarchar](50) NULL,
[ZipCode] [char](5) NULL,
[State] [char](2) NULL,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

```

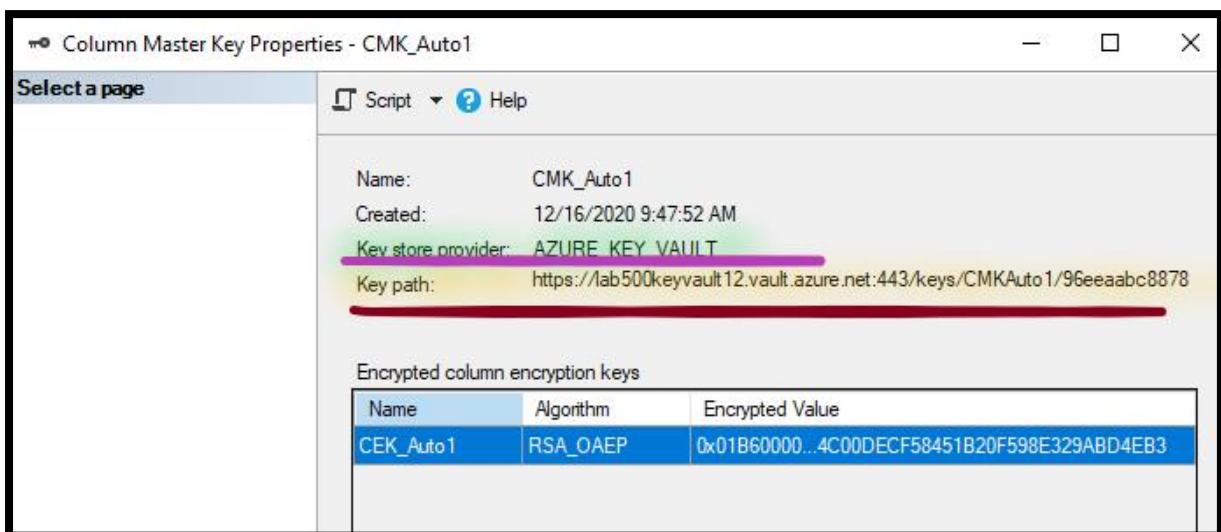
- Expand **Column Master Keys**.

i. Right-click on **CMK_Auto1** key.

ii. Select **Properties**.



Note: You can see the Key path is Azure key Vaults.



Info: A Column Encryption Key is used to protect and encrypt data in a column. A Column Master Key is used to protect the (one or more) column encryption keys. The information about the Column Master Key is stored in external key stores like Azure Key Vault.

Note: Minimize the SQL Server Management Studio. Don't close the SSMS Console.

Step 11: View the CMK Key

74.From the **Azure Portal**, go to the left menu, select **resource group**

75.Open resource group **Az-204-13-01-RG**

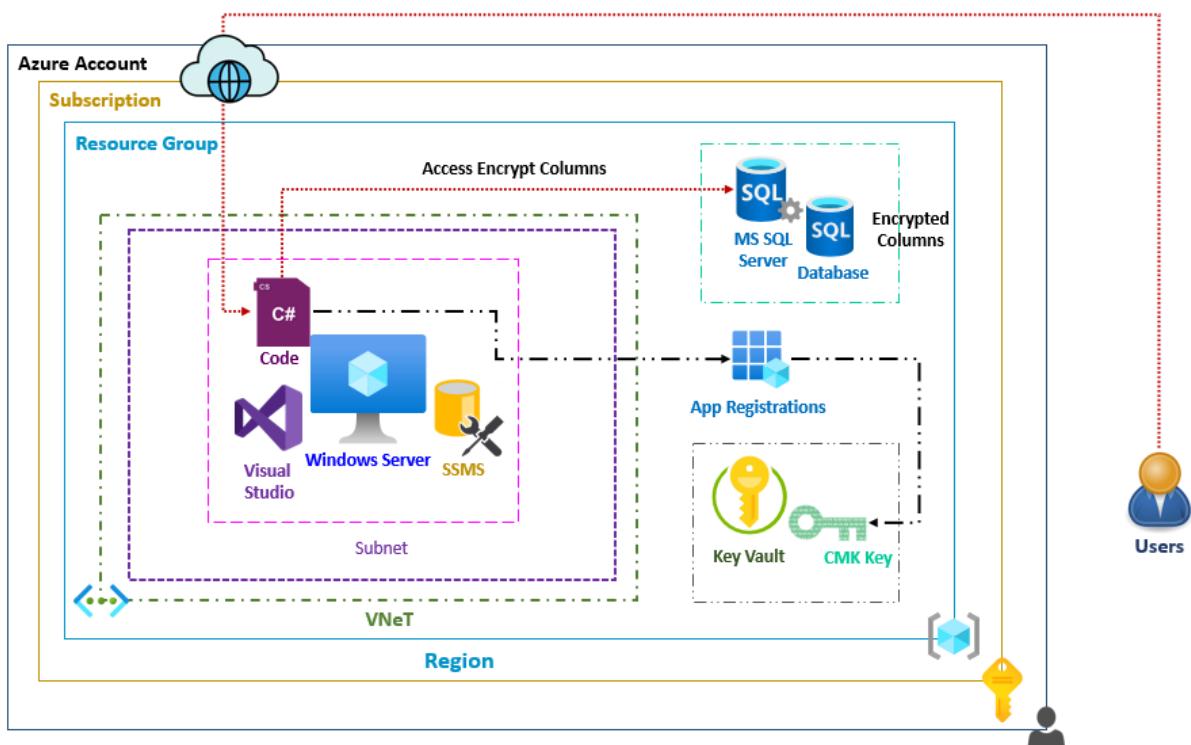
76.Open Key Vault **lab204keyvault123**

77.Select **Keys** under **settings**

Note: You can see the CMKAuto1 key.

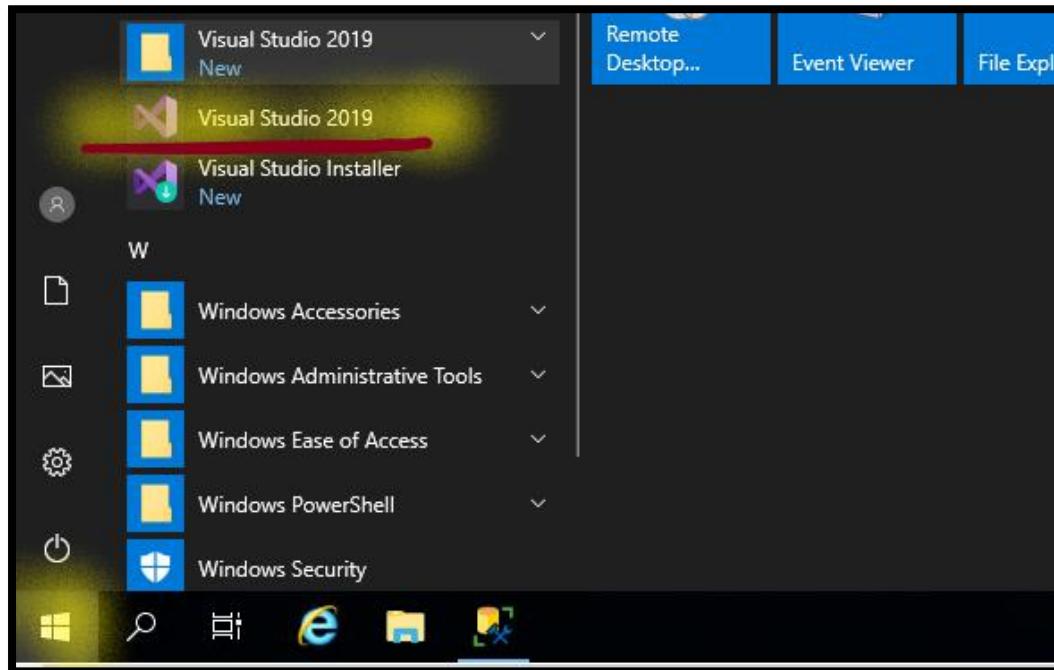
Name	Status
Az500LabKey	✓ Enabled
CMKAuto1	✓ Enabled

Task 5: Access Encrypted Columns



Step 1: Access Encrypted Columns from Application

78.From the **az204-13-vm** virtual machine, Open **Visual Studio 2019**.

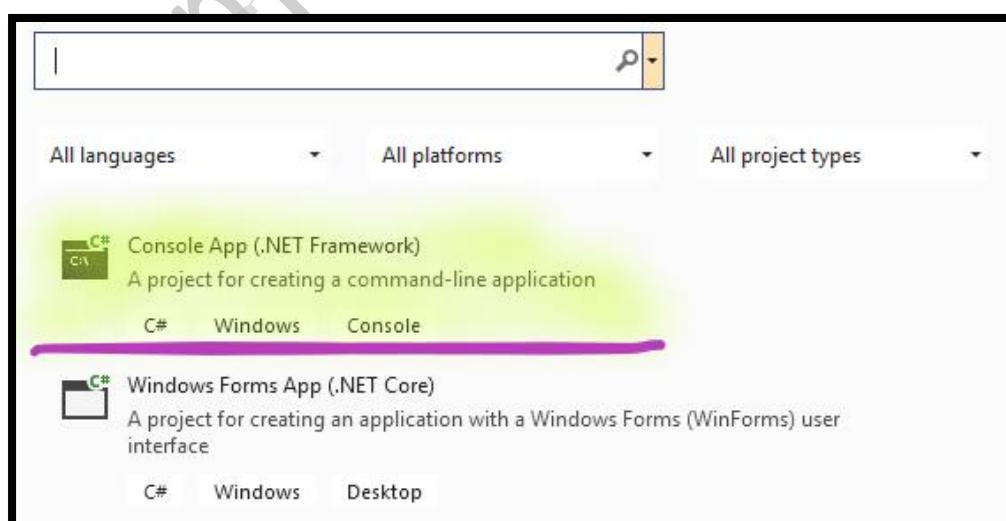


79.In the Welcome message, select "**Not now, may be later**".

80.Select **Start Visual Studio**

81.On the **Get started page**:

- a. Click **Create a new project**.
- b. In the list of **Project templates**:
 - i. Search for **Console App (.NET Framework)**.
 - ii. Click **Console App (.NET Framework)** for **C#**.



iii. Click **Next**.

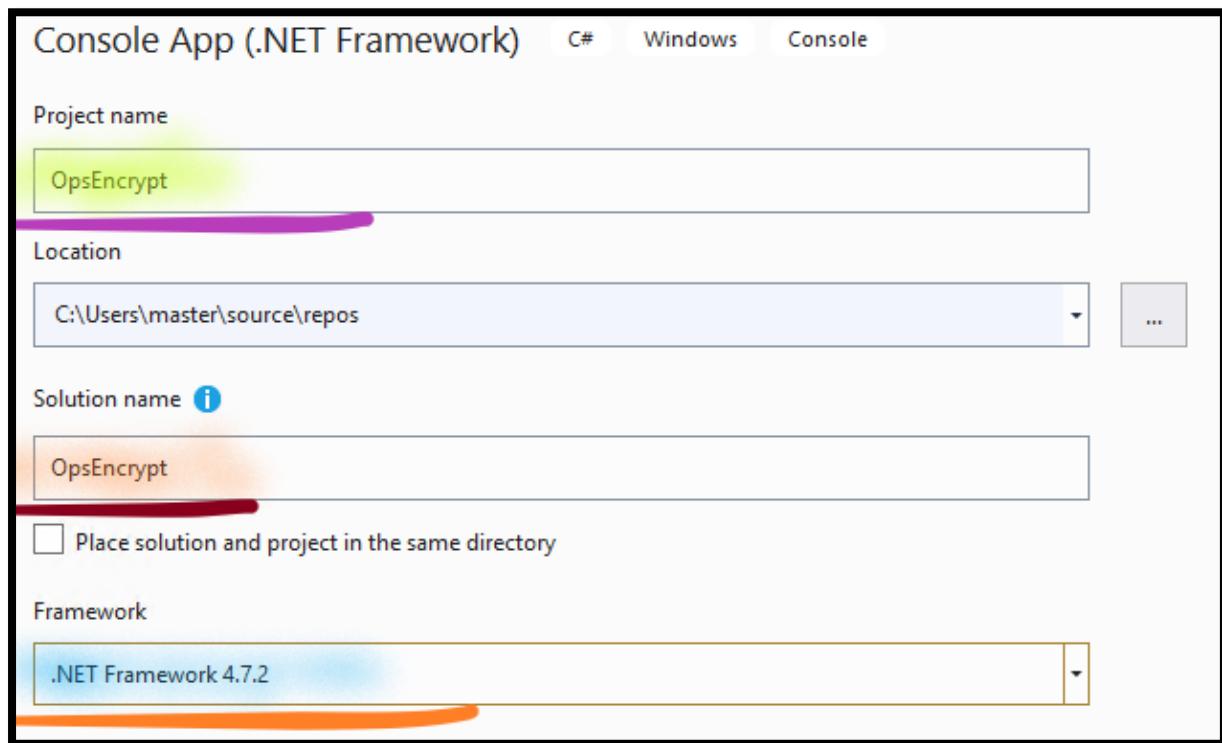
Note: If you missed to install **.NET desktop development** workload, then open Visual Studio, Select Tools and Select Get Tools and Features to install it.

82. On the **Configure your new project page**:

a. **Specify** the following settings:

- **Project name:** Write **OpsEncrypt**.
- **Solution name:** Write **OpsEncrypt**.
- **Framework:** Dropdown and Select **.NET Framework 4.7.2**.

Note: Leave other settings as default.

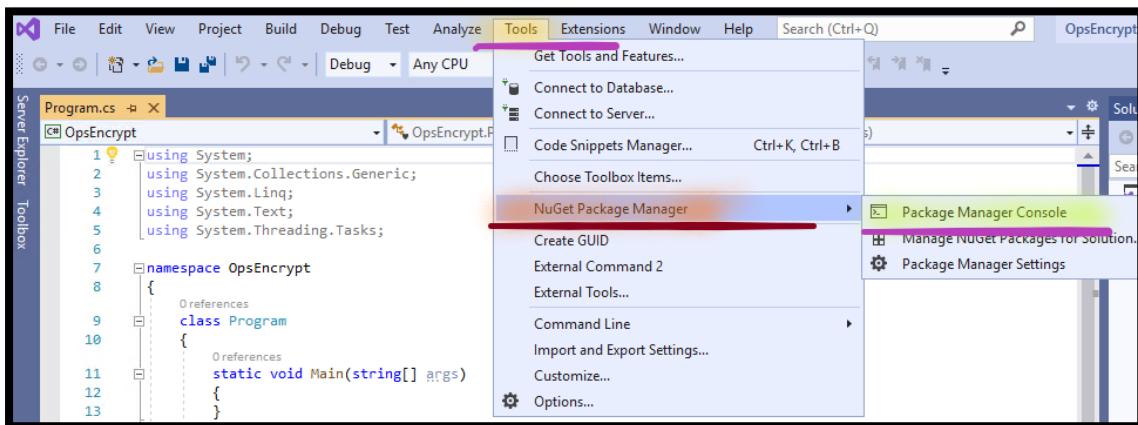
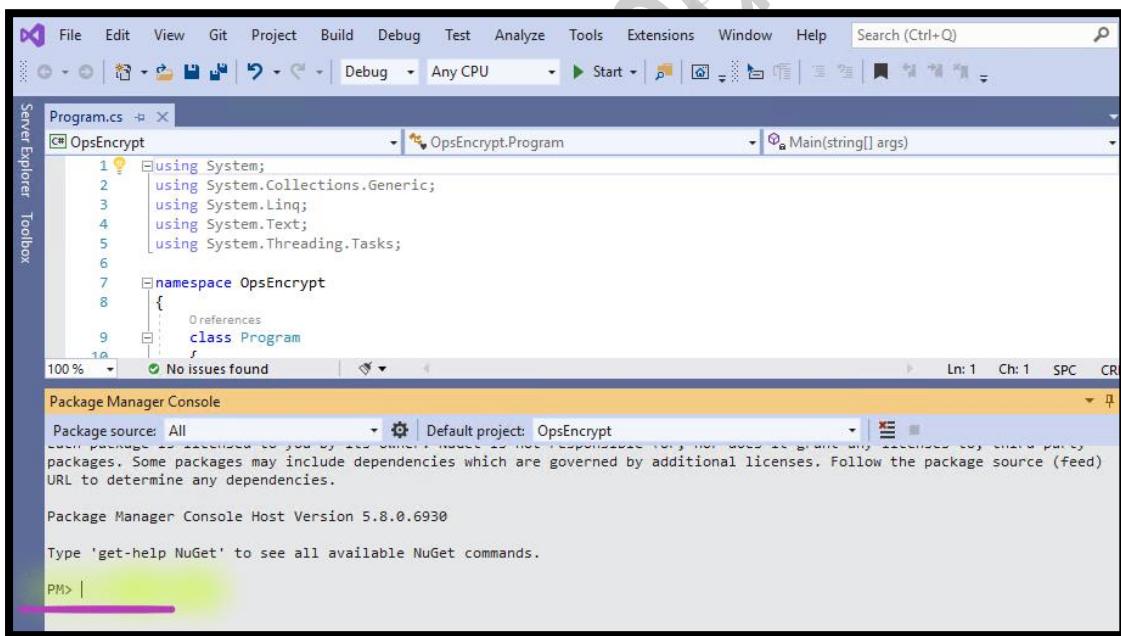


- Select **Create**.

Note: **Wait**, unless Project gets created.

83.In the Visual Studio console:

- a. Click the **Tools menu**.
- b. Click **NuGet Package Manager**.
- c. Click **Package Manager Console**.

**84.In the Package Manager Console pane, run the following one by one to install the required NuGet packages:**

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

```
Package Manager Console
Package source: All | Default project: OpsEncrypt
Each package is licensed to you by its owner. NuGet is not responsible for, nor does it grant any warranties about, any of its packages. Some packages may include dependencies which are governed by additional licenses. Follow the links in the license files to determine any dependencies.

Package Manager Console Host Version 5.7.0.6726

Type 'get-help NuGet' to see all available NuGet commands.

PM> Install-Package Microsoft.SqlServer.Management.AlwaysEncrypted.AzureKeyVaultProvider
Attempting to gather dependency information for package
'Microsoft.SqlServer.Management.AlwaysEncrypted.AzureKeyVaultProvider.2.4.0' with respect to project
'.NETFramework,Version=v4.7.2'
Gathering dependency information took 4.26 sec
```

Install-Package Microsoft.IdentityModel.Clients.ActiveDirectory

```
PM> Install-Package Microsoft.IdentityModel.Clients.ActiveDirectory
Attempting to gather dependency information for package 'Microsoft.IdentityModel.Clients.ActiveDirectory' with respect to project 'OpsEncrypt', targeting '.NETFramework,Version=v4.7.2'
Gathering dependency information took 329 ms
Attempting to resolve dependencies for package 'Microsoft.IdentityModel.Clients.ActiveDirectory.5.2.8' with the Dependency Status 'Lowest'
Resolving dependency information took 0 ms
Resolving actions to install package 'Microsoft.IdentityModel.Clients.ActiveDirectory.5.2.8'
Resolved actions to install package 'Microsoft.IdentityModel.Clients.ActiveDirectory.5.2.8'
```

85. Open the **program.cs** file in **Notepad**.

Note: **Program.cs** is provided with the Lab Manual.

- a. Replace the **<connection string noted earlier>** placeholder with the Azure SQL database **ADO.NET** connection string **you copied in the previous step**.
- b. Replace the **<client id noted earlier>** placeholder with the value of **Application (client) ID** of the registered **app you copied in the previous step**.
- c. Replace the **<key value noted earlier>** placeholder with the value of **Key1** of the registered app **you copied in the previous step**.
- d. **Save** the **program.cs** file

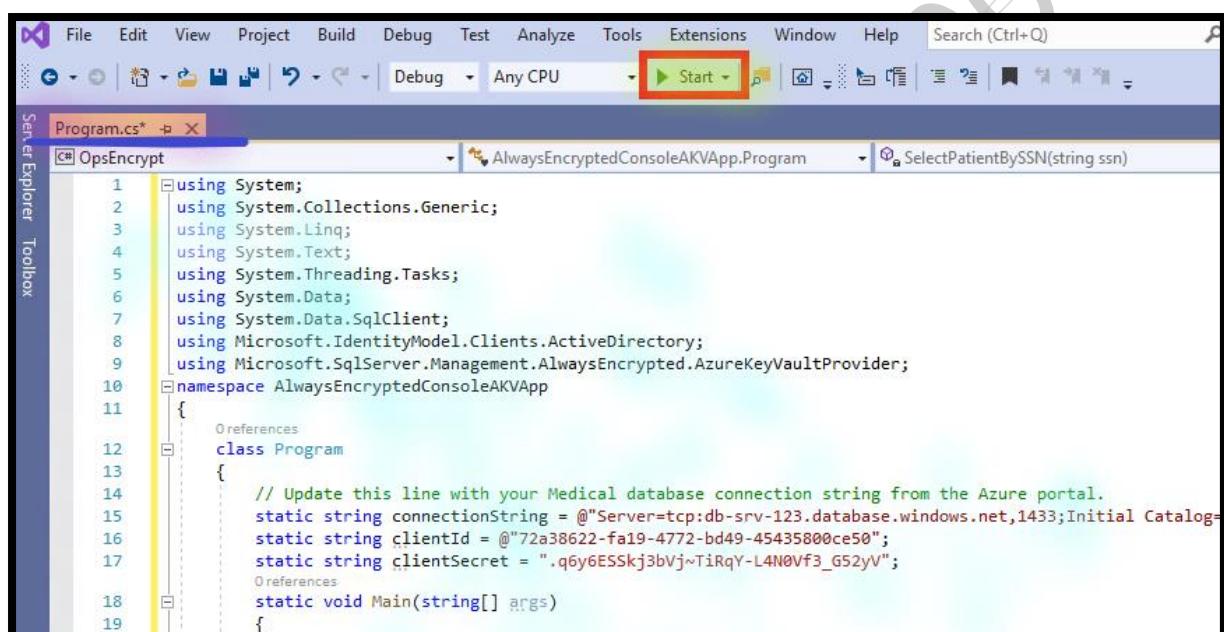
Note: Don't replace the start and end quote " ".

```

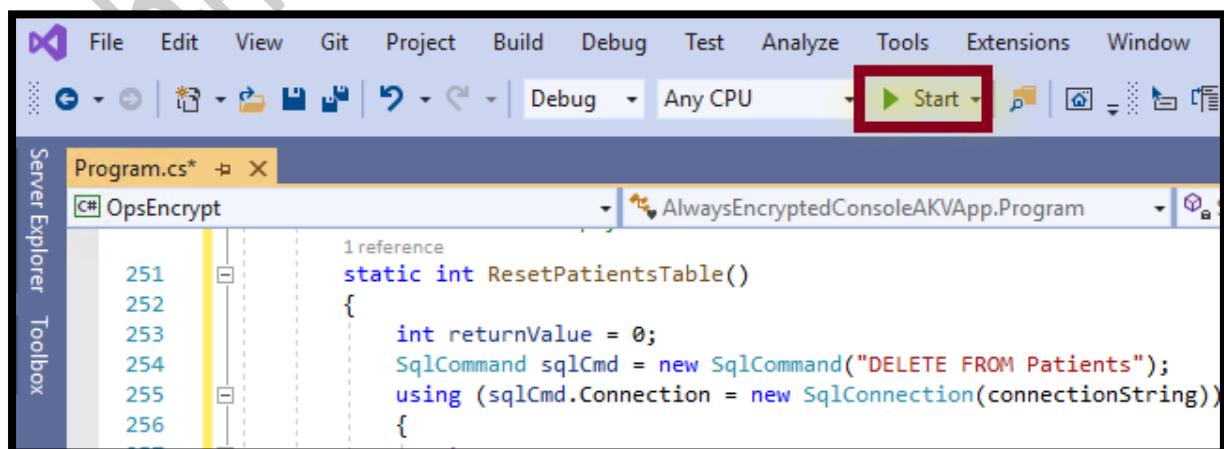
    {
        class Program
        {
            // Update this line with your Medical database connection string from the Azure portal.
            static string connectionString = @"Server=tcp:db-srv-123.database.windows.net,1433;Initial Catalog=medic;
            static string clientId = "72a38622-fa19-4772-bd49-45435800ce50";
            static string clientSecret = ".q6y6ESSkj3bVj-TiRqY-L4N0Vf3_G52yV";
            static void Main(string[] args)
            {

```

86.From the Visual Studio console, in the **Program.cs**, **replace** its content with the code.



87.Click the **Start, Start** button to **initiate the build** of the console application and start it.



88.The application will start a Command Prompt window. When **prompted** for password, write **Lab-Password** to connect to Azure SQL Database.

```
C:\Users\master\source\repos\OpsEncrypt\OpsEncrypt\bin\Debug\OpsEncrypt.exe
Signed in as: 72a38622-fa19-4772-bd49-45435800ce50
Original connection string copied from the Azure portal:
Server=tcp:db-srv-123.database.windows.net,1433;Initial Catalog=medical;Persist Se
lsword={your_password};MultipleActiveResultSets=False;Encrypt=True;TrustServerCerti
Updated connection string with Always Encrypted enabled:
Data Source=tcp:db-srv-123.database.windows.net,1433;Initial Catalog=medical;Persi
n;Password={your_password};MultipleActiveResultSets=False;Connect Timeout=30;Encry
column Encryption Setting=Enabled

Enter server password:
Lab-Password

Adding sample patient data to the database...
All the records currently in the Patients table:
Orlando Gee    SSN: 999-99-0001        Birthdate: 1/4/1964 12:00:00 AM
Keith Harris    SSN: 999-99-0002        Birthdate: 6/20/1977 12:00:00 AM
Donna Carreras  SSN: 999-99-0003        Birthdate: 2/9/1973 12:00:00 AM
Janet Gates     SSN: 999-99-0004        Birthdate: 8/31/1985 12:00:00 AM
Lucy Harrington SSN: 999-99-0005        Birthdate: 5/6/1993 12:00:00 AM

Now lets locate records by searching the encrypted SSN column.
Please enter a valid SSN (ex. 999-99-0003):
```

Step 2: Access the Data Securely

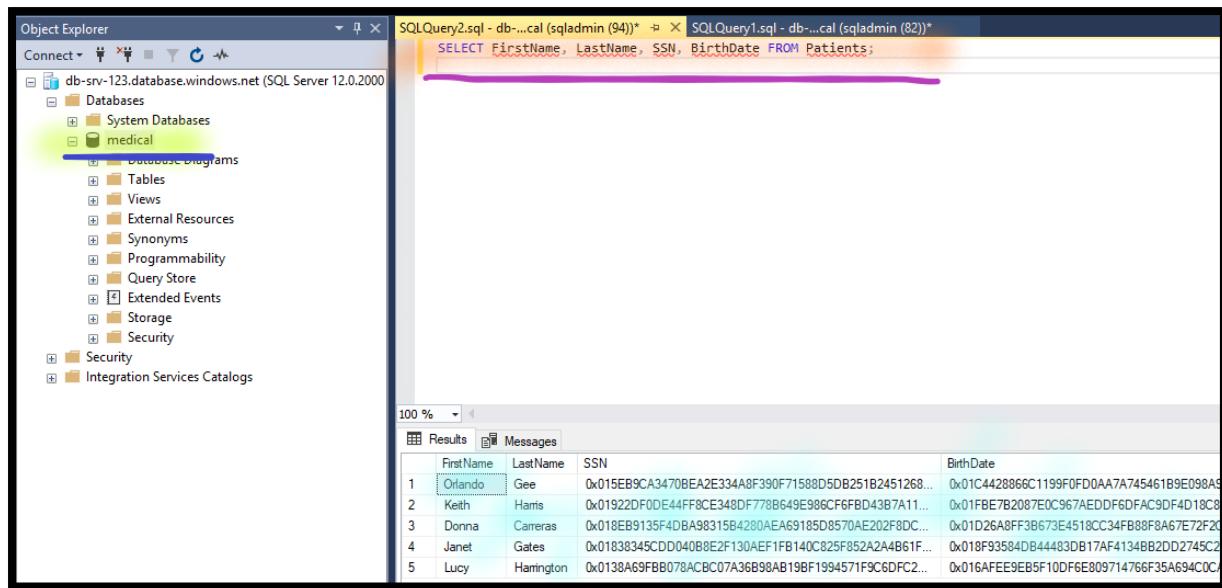
89.**Switch** to the SQL Management Studio console.

90.From the **SQL Management Studio (SSMS)**

- a. Expand **Databases**
- b. **Right-click** the **medical** database
- c. Click **New query**
- d. **Copy** the below query to verify that the data that loaded into the database from the console app is encrypted.

```
SELECT FirstName, LastName, SSN, BirthDate FROM Patients;
```

- e. Click **Execute**



Note: You can see the Encrypted, SSN and BirthDate data.

91. **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**, copy the following and press the Enter key:

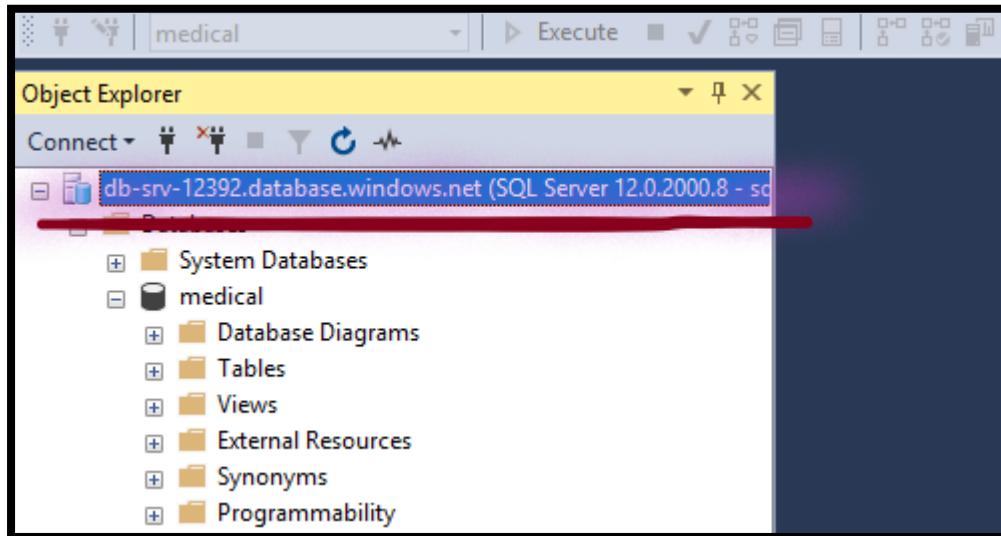
```
999-99-0003
```

```
Please enter a valid SSN (ex. 999-99-0003):  
999-99-0003  
Patient found with SSN = 999-99-0003  
Donna Carreras  SSN: 999-99-0003          Birthdate: 2/9/1973 12:00:00 AM  
Press Enter to exit...
```

Note: Verify that the data returned by the query is not encrypted.

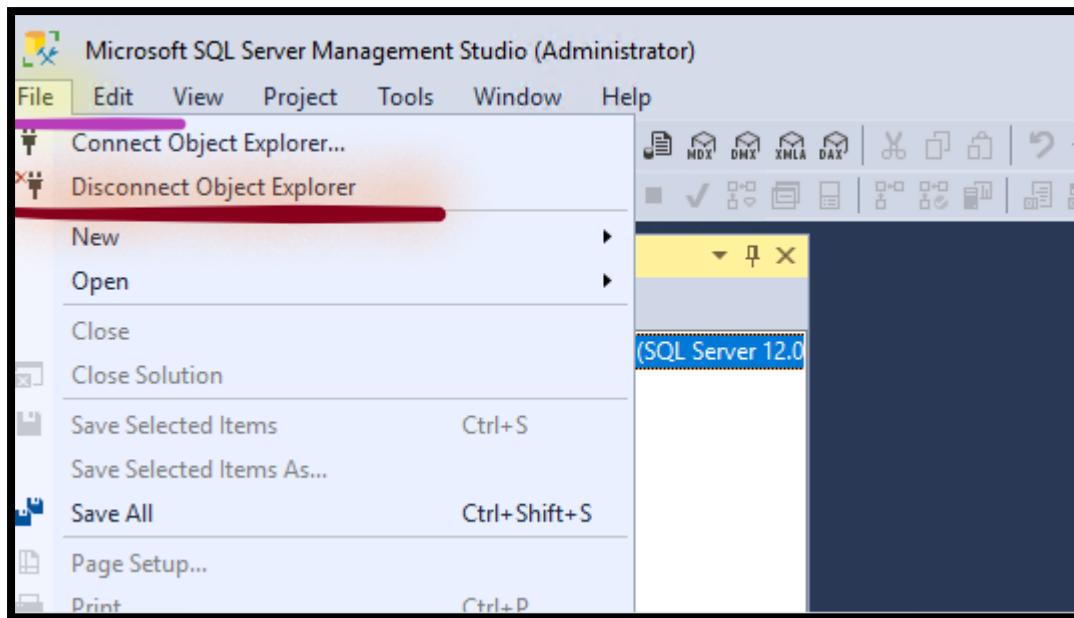
Step 3: Disconnect SQL Database

92. Select **SQL Server** from the **SQL Management Studio console**.



93. Select **File** from the **SQL Management Studio console**.

94. Select **Disconnect object explorer**



Task 6: Delete Environment

Step 1: Delete Resource Group

1. Delete **Az-204-13-01-RG** resource groups

© No part of this manual, may be reproduced in whole or in part in any manner without the permission of the copyright owner.