

# Microsoft SQL server set up and the Entity Framework for Windows

YKUMAR 2021-02-07, updated 2023-02-11.

## Table of Contents

1. SQL server installation and set up .....	2
1.1. Download and install SQL server Express .....	2
TASK 1: Once the installation fully completes <i>successfully</i> take a screenshot. The screenshot should contain your name(s) course, section, Lab # and the Semester (you can use a docx with such text or open Notepad for it). .....	5
2. SQL Server Management Studio (SSMS) and Azure studio (by default) installation and set up .....	5
2.1. Download and install SSMS.....	5
Click on Install SSMS from the screen above, you can also do this from <a href="https://learn.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver16">https://learn.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver16</a> .....	5
TASK 2: Once the installation fully completes take a screenshot. The screenshot should contain your name(s) course, section, Lab # and the Semester and both apps open fully visible (you can use a docx with such text or open Notepad for it). .....	6
3. SQL Server Connection.....	6
3.1. Connect to the server.....	6
3.2. Create a Database and a Table.....	7
Right-click on Databases folder → New Database.....	7
3.3. Insert and Select the Data .....	8
TASK 3: Take a screenshot of your result. The table should have your (or your teammate username), the first recods / records of the table should display your name / all teammate names.....	10
3.3. Create a User. ....	10
3.4. Try to connect with the new User. ....	14
TASK 4: Take a screenshot of your result. The user should have your (your teammate username). ....	16
3.4. Try to connect with the new User from Azure Data Studio.....	19
TASK 5: Take a screenshot of your result. The user and table should have your (or your teammate username). .....	22
4. Connect with C# code .....	22
4.1. Create a new Console App in Visual Studio .....	22
4.2. Import Nugget Packages .....	23
4.3. Run the code and debug if needed. ....	23
5. Entity Framework Set up.....	26

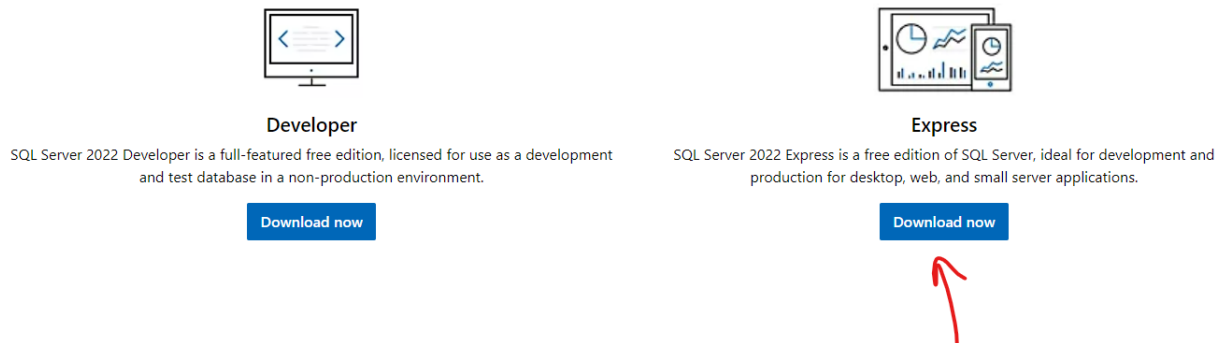
5.1. Create a new MVC Web App in Visual Studio.....	27
5.2. Add Nugget packages.....	28
5.3. Create a new Entity.....	29
5.4. Add new Controller and a View.....	30
TASK 7: Take a screenshot of your results from Entity Framework part (both autogenerated and custom code + the browser output). The MVC APP should have your (or your teammate username). Push your results to <i>GitHub</i> .....	
	33

# 1. SQL server installation and set up

## 1.1. Download and install SQL server Express.

Open link - <https://www.microsoft.com/en-us/sql-server/sql-server-downloads> and scroll down to install SQL server Express Edition.

Or, download a free specialized edition



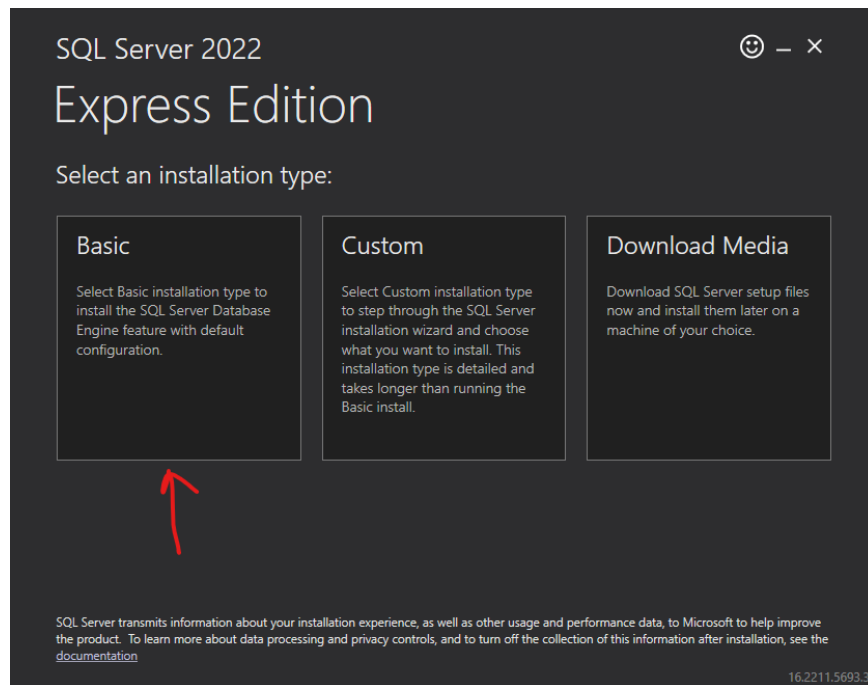
Click download now – the Installer should appear at the bottom of the screen. Right click on it - **Show in Folder** – Right click - **Run as administrator**.

1

Accept making changes on your device.

Choose the Basic Installation type:

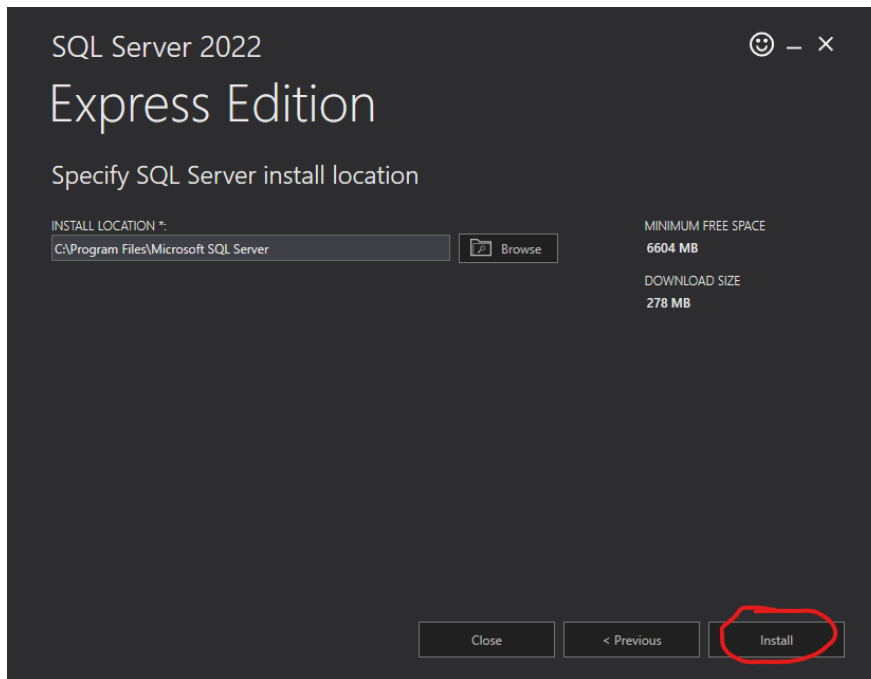
2



Accept the License Terms.

Choose the **default location (it is recommend to proceed with defaults)**. Click Install

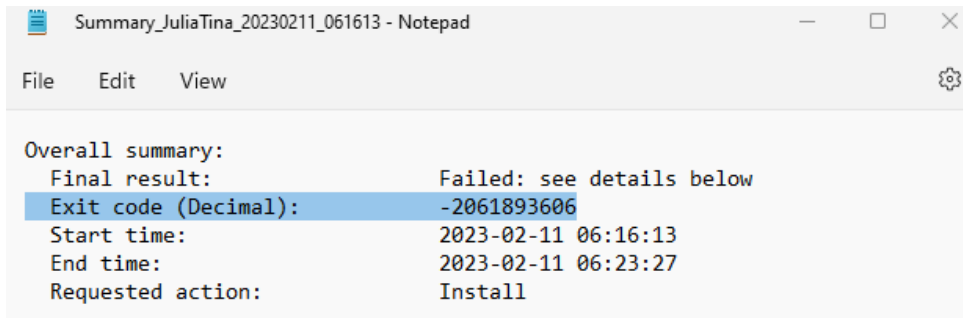
2



Click Install. Wait for download and installation to complete.

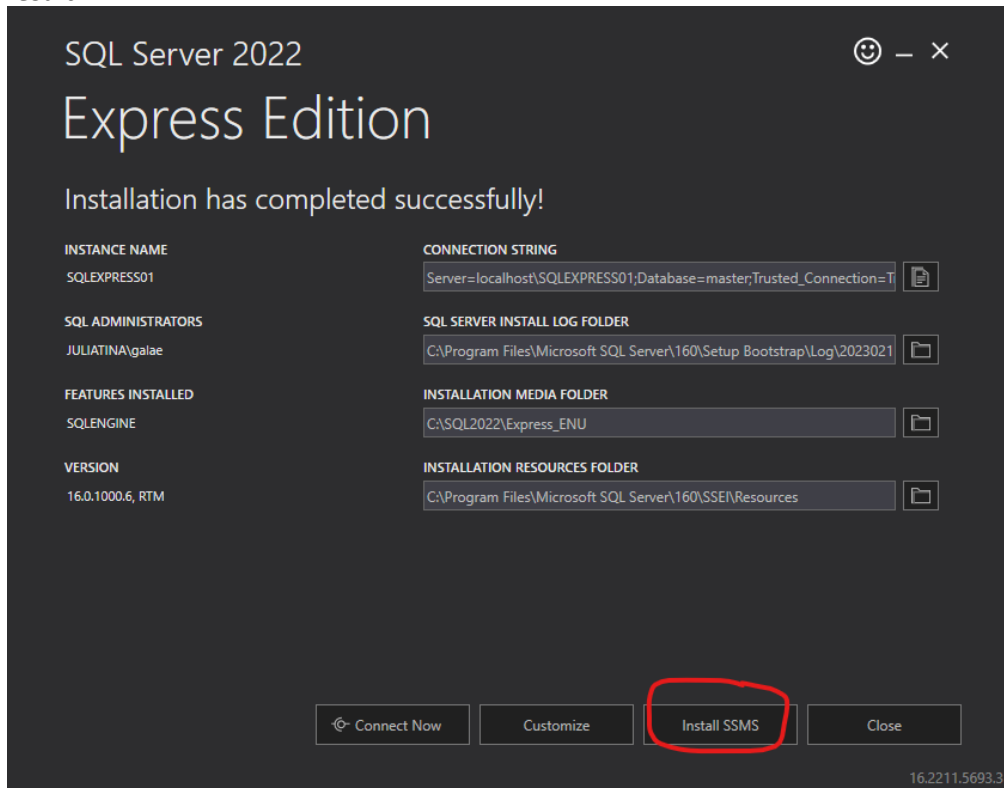
***In case of errors ONLY: In case of installation error, research on it before making a second attempt. In the folder you prompted to locate an error, find the Summary.txt file and / or Detail.txt, open and check. Example below:***

3



*find out the Exit code and Google it (the outcome might vary). Uninstall SQL server first in case of re-installation (Search for the app and choose to uninstall, sort Apps by date). Debug the issue until it is resolved. I had to perform the following on Command line as an admin and reboot:*  
<https://learn.microsoft.com/en-us/troubleshoot/sql/database-engine/database-file-operations/troubleshoot-os-4kb-disk-sector-size>

*REG ADD "HKLM\SYSTEM\CurrentControlSet\Services\stornvme\Parameters\Device" /v "ForcedPhysicalSectorSizeInBytes" /t REG\_MULTI\_SZ /d "\*" 4095" /f  
REG QUERY "HKLM\SYSTEM\CurrentControlSet\Services\stornvme\Parameters\Device" /v "ForcedPhysicalSectorSizeInBytes"  
Expected result:*



**TASK 1: Once the installation fully completes successfully take a screenshot. The screenshot should contain your name(s) course, section, Lab # and the Semester (you can use a docx with such text or open Notepad for it).**

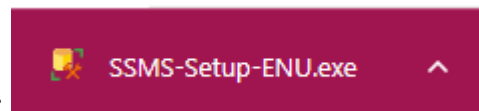
## 2. SQL Server Management Studio (SSMS) and Azure studio (by default) installation and set up

### 2.1. Download and install SSMS.

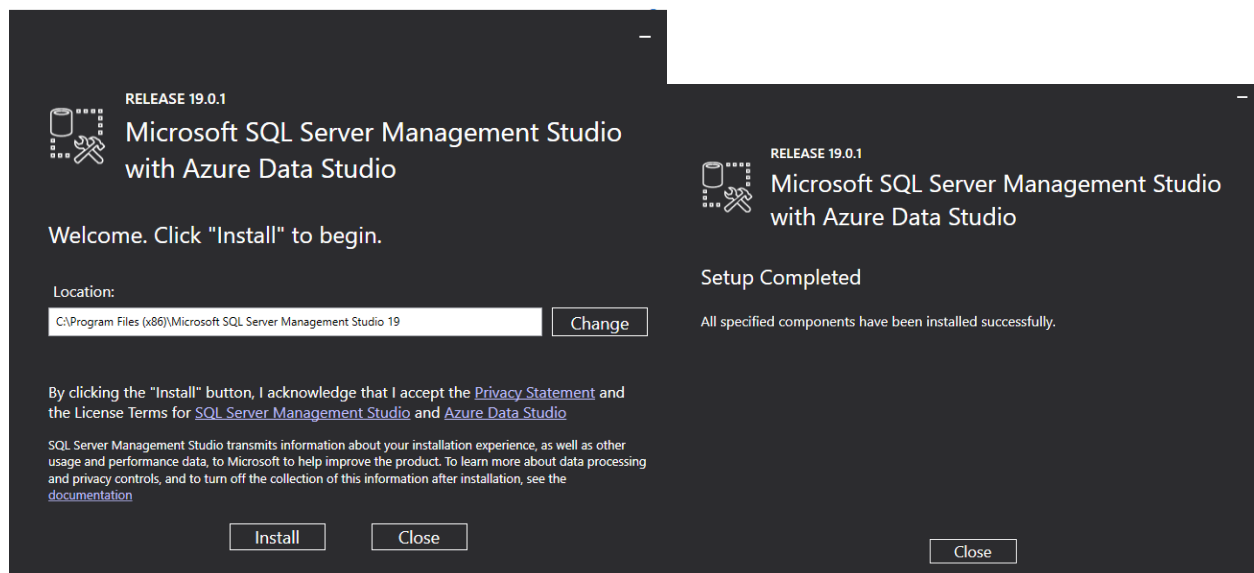
Click on **Install SSMS** from the screen above, you can also do this from <https://learn.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver16>.

### Download SSMS

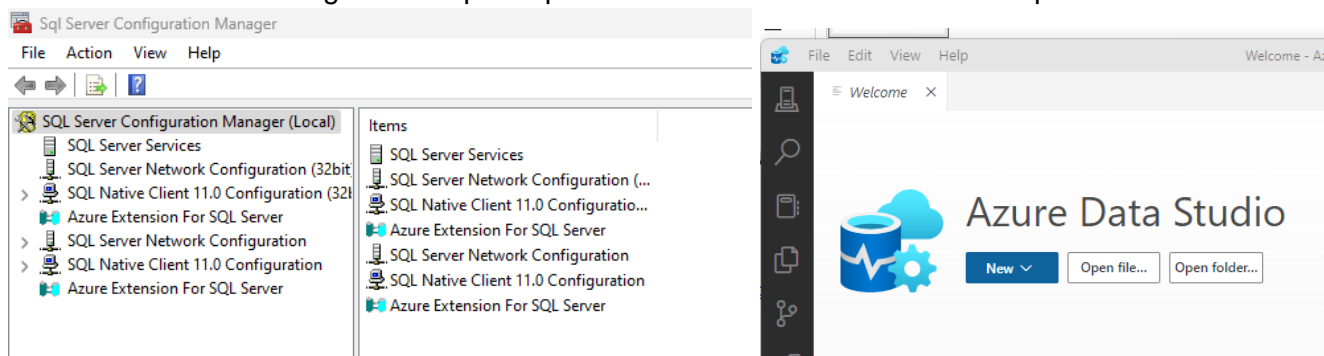
Download the link: [Free Download for SQL Server Management Studio \(SSMS\) 19.0.1](#)



Double-click on the installer:



Click **Install**. Allow changes. Attempt to open both SSMS and Azure Studio. Expected results:



**TASK 2:** Once the installation fully completes take a screenshot. The screenshot should contain your name(s) course, section, Lab # and the Semester and both apps open fully visible (you can use a docx with such text or open Notepad for it).

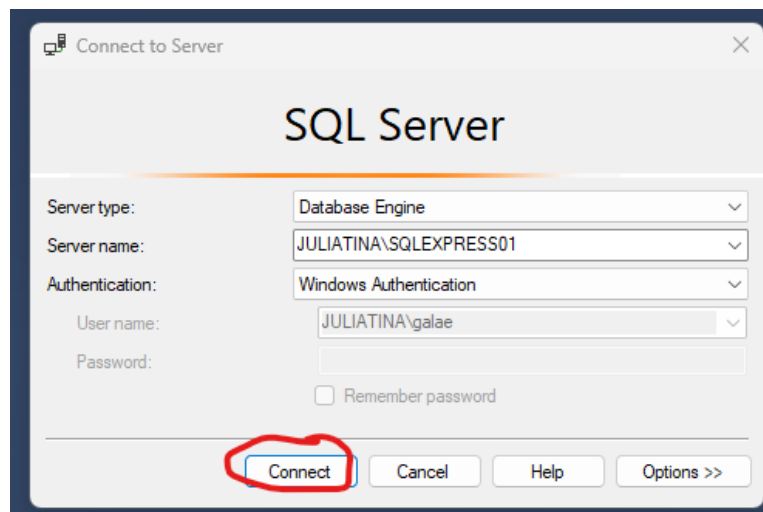
## 3. SQL Server Connection

### 3.1. Connect to the server.

Type and Find **SSMS** in search and Right-click – run as administrator. (close it if was already open)

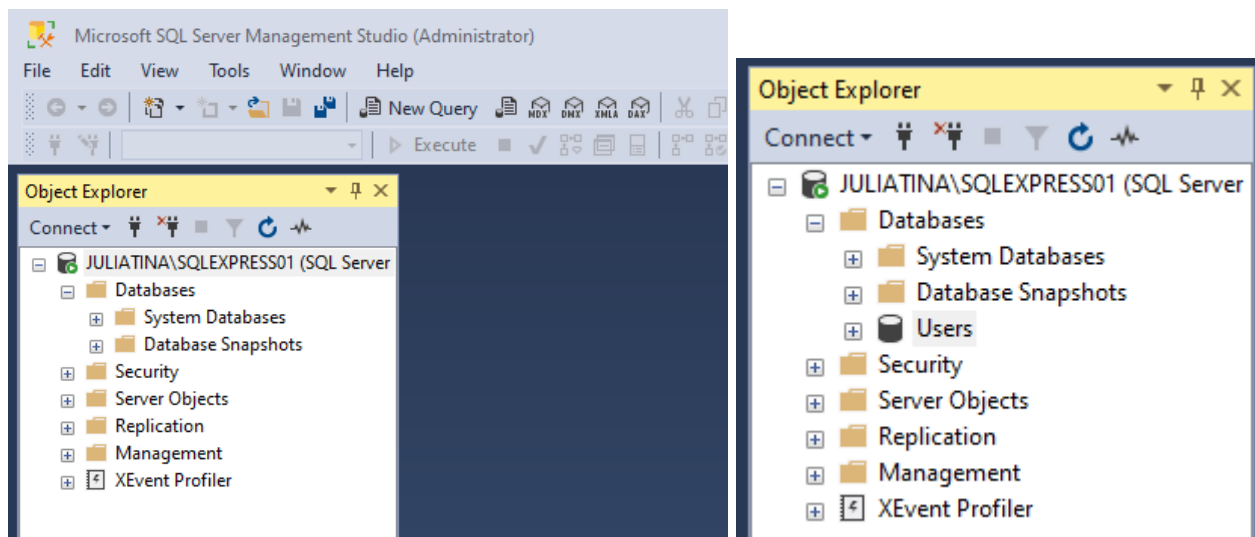


Allow changes. Right-click on the icon - pin into taskbar



Expected result:

Keep **defaults**, attempt to connect.

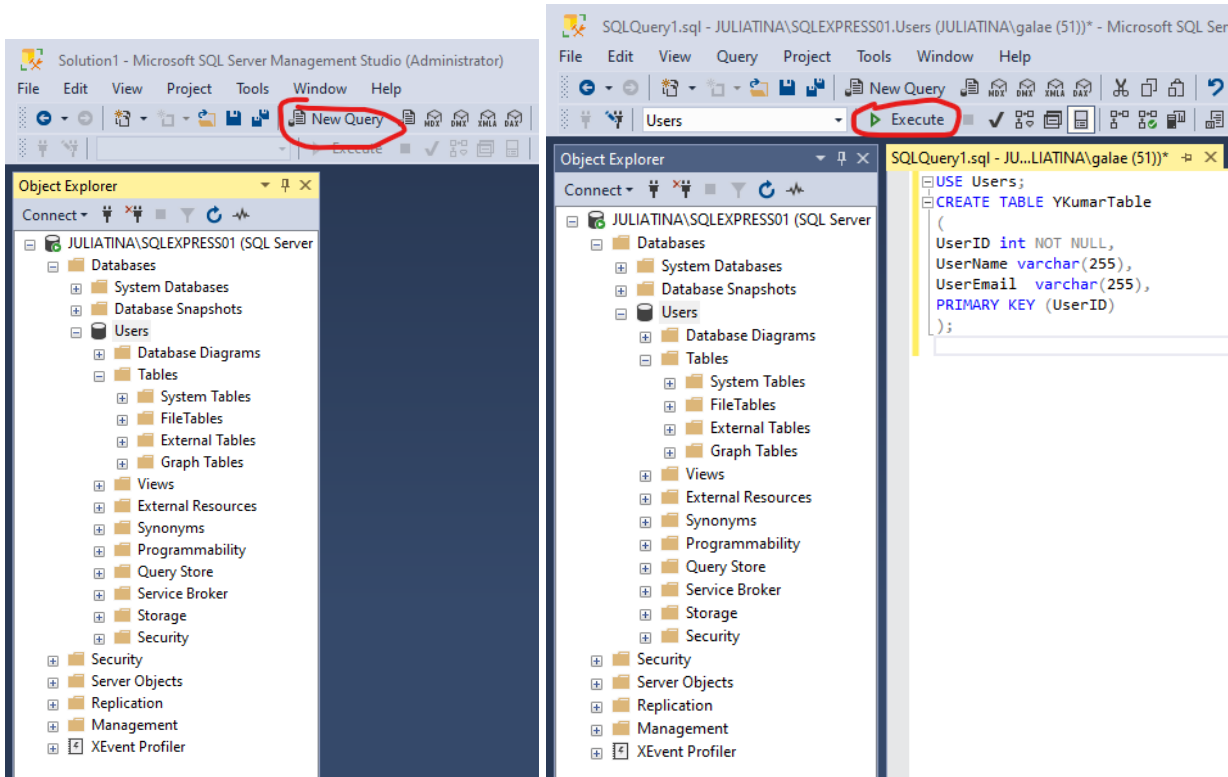


## 3.2. Create a Database and a Table.

**Right-click on Databases folder → New Database**

and add database **Users**. **Click OK.** (see result above).

**Start a new Query**



**USE Users;**

**CREATE TABLE YKumarTable**

**(**

**UserID int NOT NULL,**

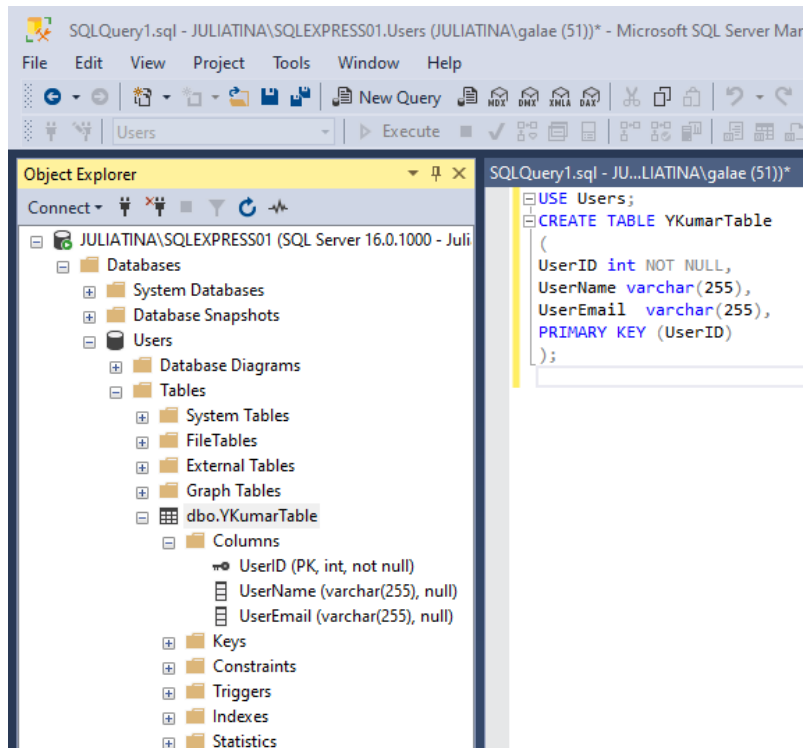
**UserName varchar(255),**

**UserEmail varchar(255),**

**PRIMARY KEY (UserID)**

**);**

Right-click, refresh Tables, Observe the result.



### 3.3. Insert and Select the Data

Re-write the previous query with

**USE Users;**

**--CREATE TABLE YKumarTable**

**--(**

**--UserID int NOT NULL,**

**--UserName varchar(255),**

**--UserEmail varchar(255),**

**--PRIMARY KEY (UserID)**

**--);**

**INSERT INTO YKumarTable**

**VALUES (1, 'ykumar', 'ykumar@kean.edu');**

**INSERT INTO YKumarTable**

**VALUES (2, 'mike', 'mke@kean.edu');**

**User your username.** Execute



```
USE Users;
--CREATE TABLE YKumarTable
--(
--UserID int NOT NULL,
--UserName varchar(255),
--UserEmail varchar(255),
--PRIMARY KEY (UserID)
--);

INSERT INTO YKumarTable
VALUES (1, 'ykumar', 'ykumar@kean.edu');
INSERT INTO YKumarTable
VALUES (2, 'mike', 'mke@kean.edu');
```

Observe the result:

Right click on the table name, Select top 1000 records:

The screenshot shows the SQL Server Management Studio interface. In the Object Explorer on the left, the 'Users' database is expanded, and 'dbo.YKumarTable' is selected. The main query window displays the following SQL query:

```
SELECT TOP (1000) [UserID]
, [UserName]
, [UserEmail]
FROM [Users].[dbo].[YKumarTable]
```

The Results window at the bottom shows the output of the query:

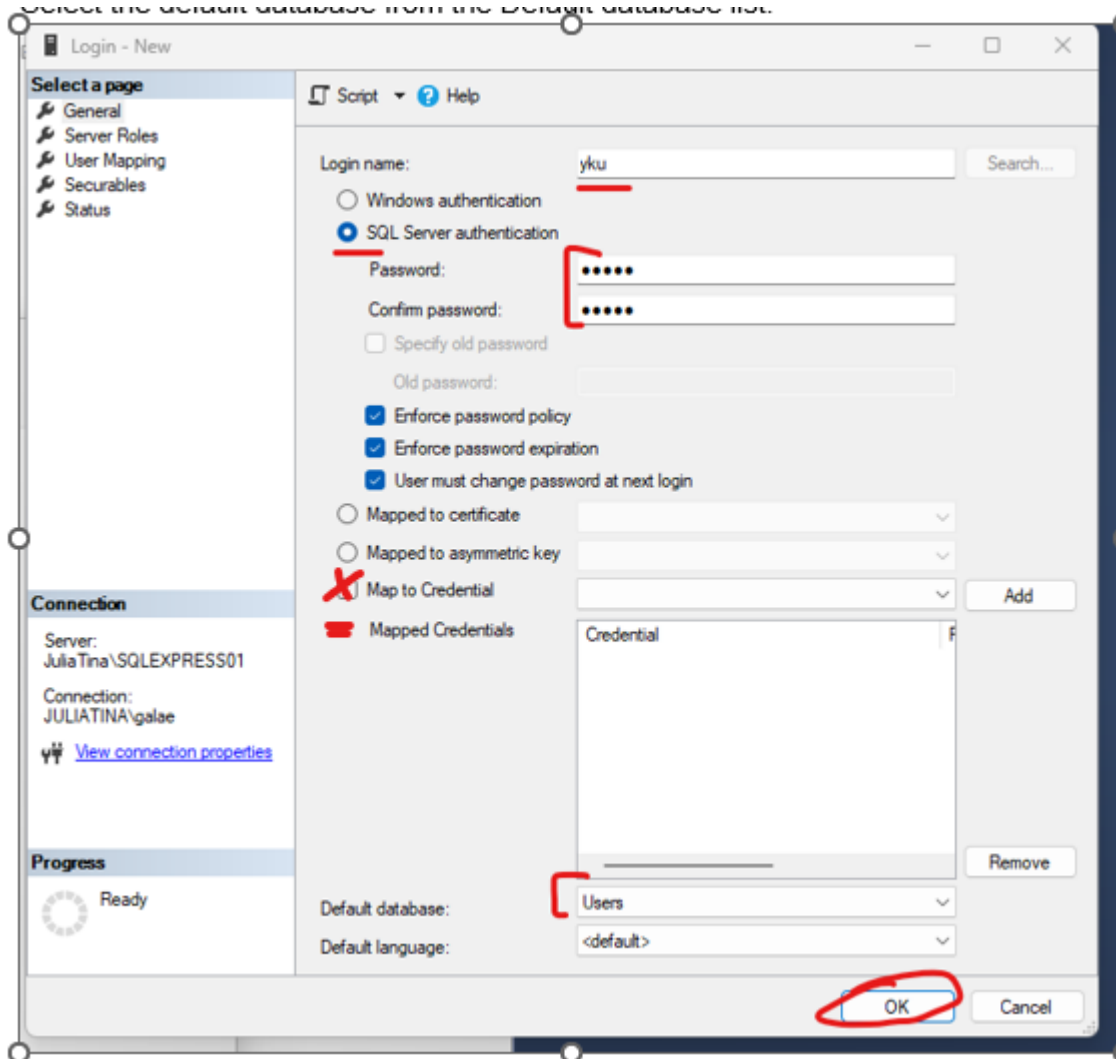
	UserID	UserName	UserEmail
1	1	ykumar	ykumar@kean.edu
2	2	mike	mke@kean.edu

**TASK 3: Take a screenshot of your result. The table should have your (or your teammate username), the first recods / records of the table should display your name / all teammate names.**

### 3.3. Create a User.

Reference: <https://www.microfocus.com/documentation/silk-test/200/en/silktestworkbench-help-en/SILKTEST-7FFBB86A-CREATINGNEWSQLSERVERADMINUSER-TSK.html>

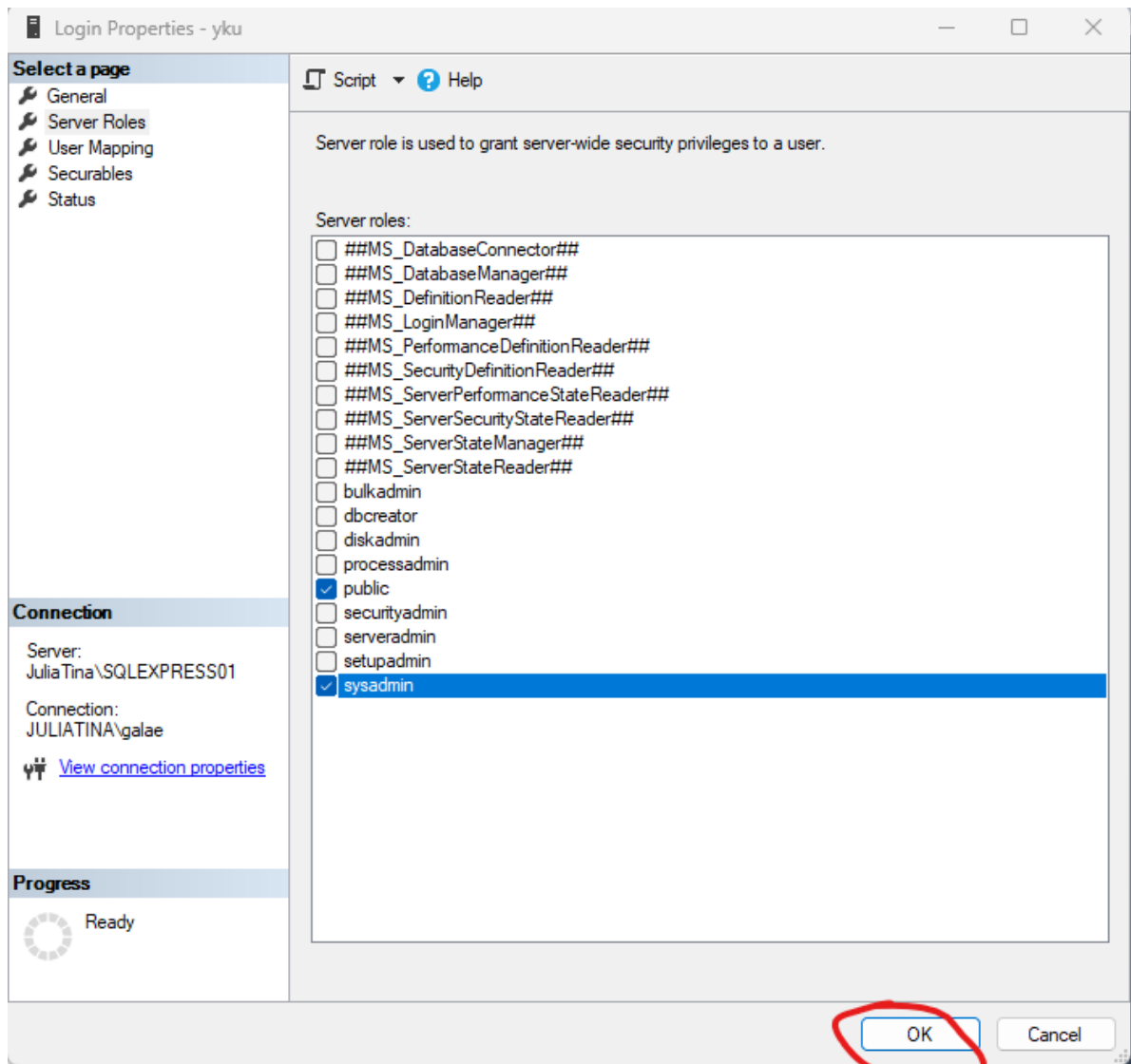
1. To create a new user with system administrator rights, perform the following steps:
2. In the Object Explorer of SQL Server Management Studio, navigate to the Security folder and expand it.
3. Right-click the Logins folder and choose New Login. The Login - New dialog box opens.
4. Select the **General** page, and then enter a user name in the Login name text box.
5. Select SQL Server Authentication and enter a password.
6. Select the default database from the Default database list.



Make sure the user is mapped to credentials

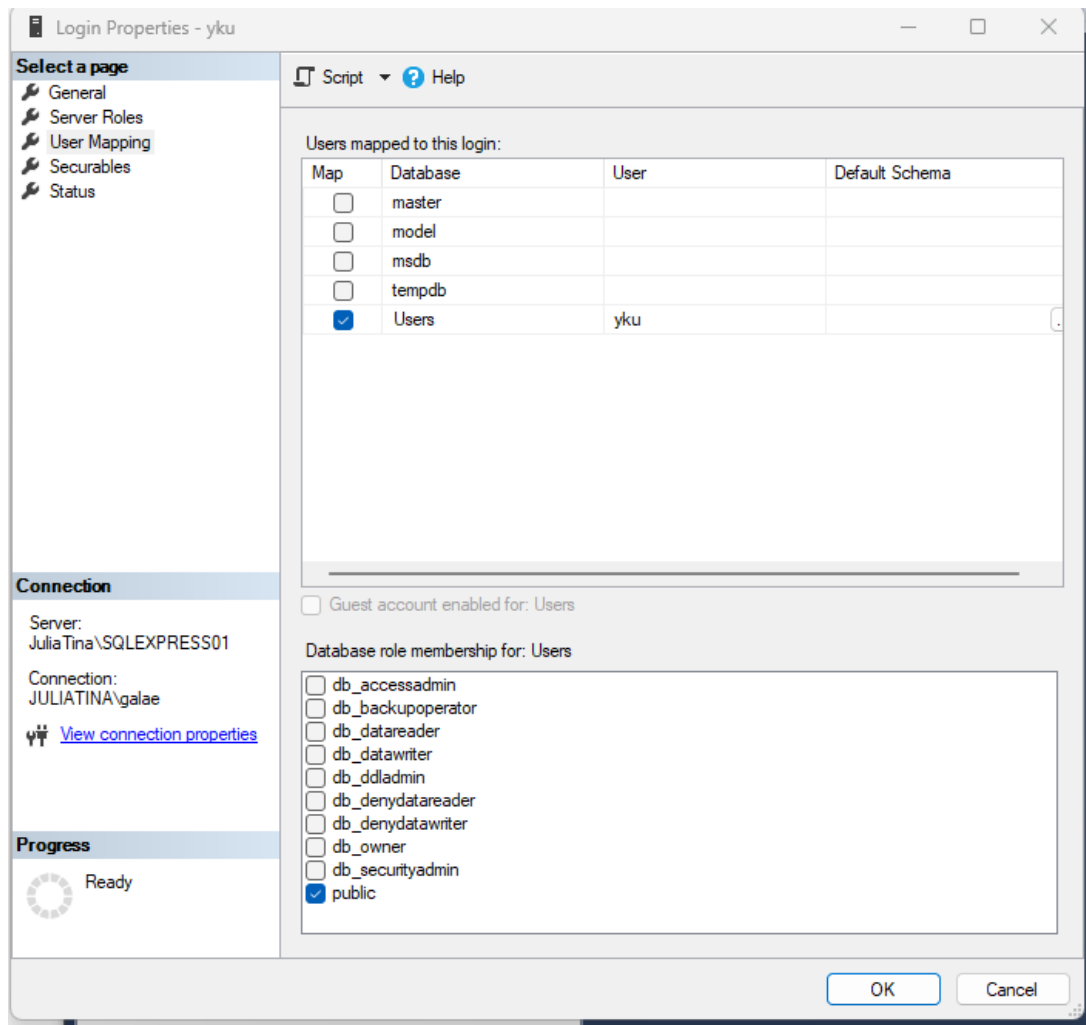
Right-click on the User -> Properties

7. Select the **Server Roles** page, and then check the sysadmin check box in the Server roles list.



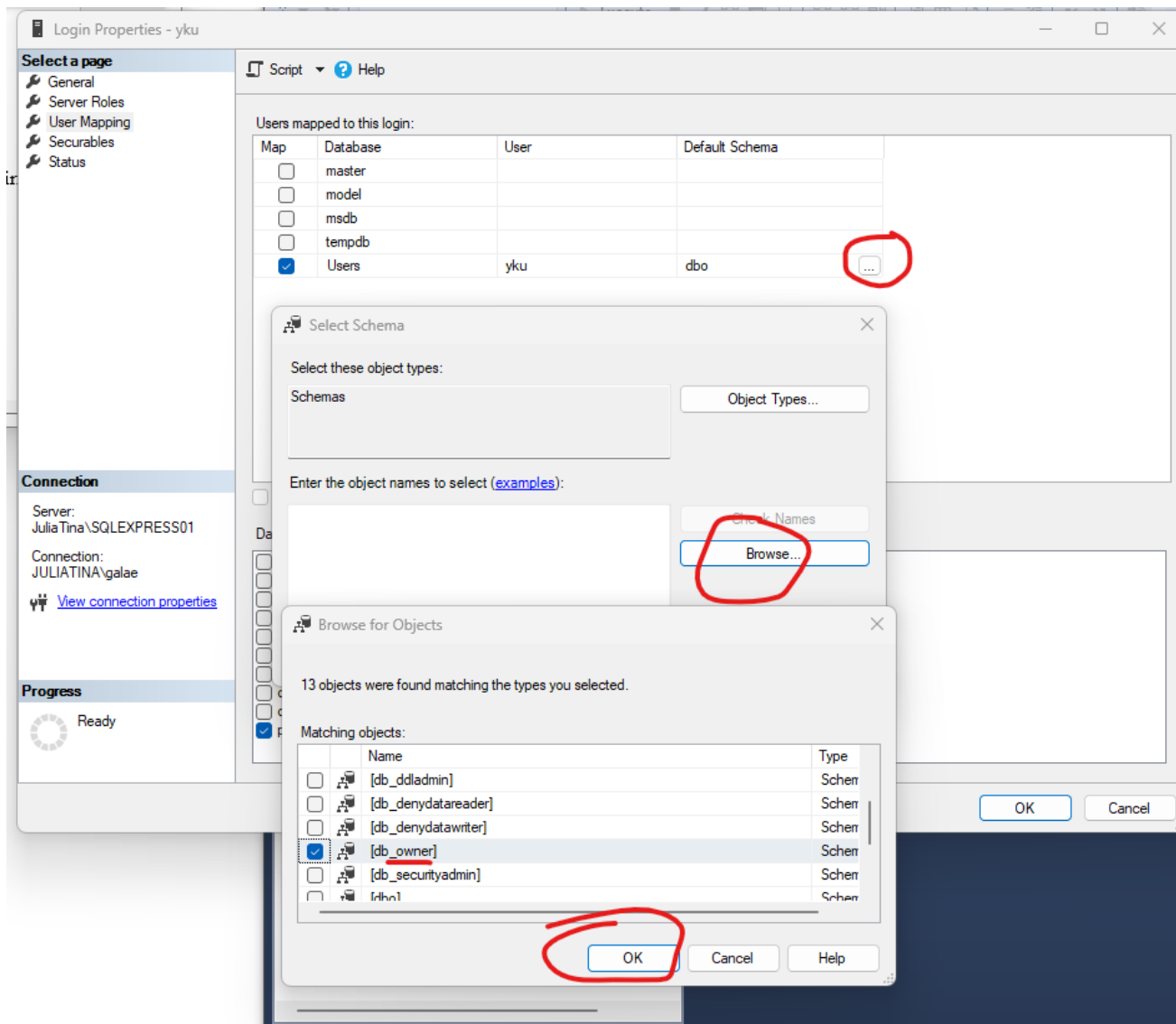
Right-click on Roles -> refresh, make sure the sysadmin is there?

8. Select the User Mapping page.
9. In the Map column, check the check box for the database that the new login can access. By default, the login name appears in the User column. Leave this value.

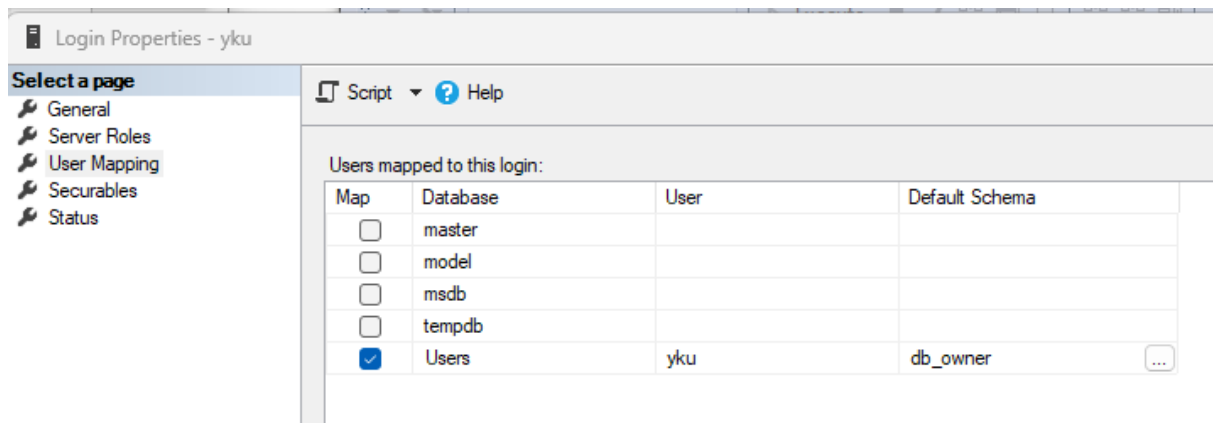


10. In the Database role membership for list, check the db\_owner check box.

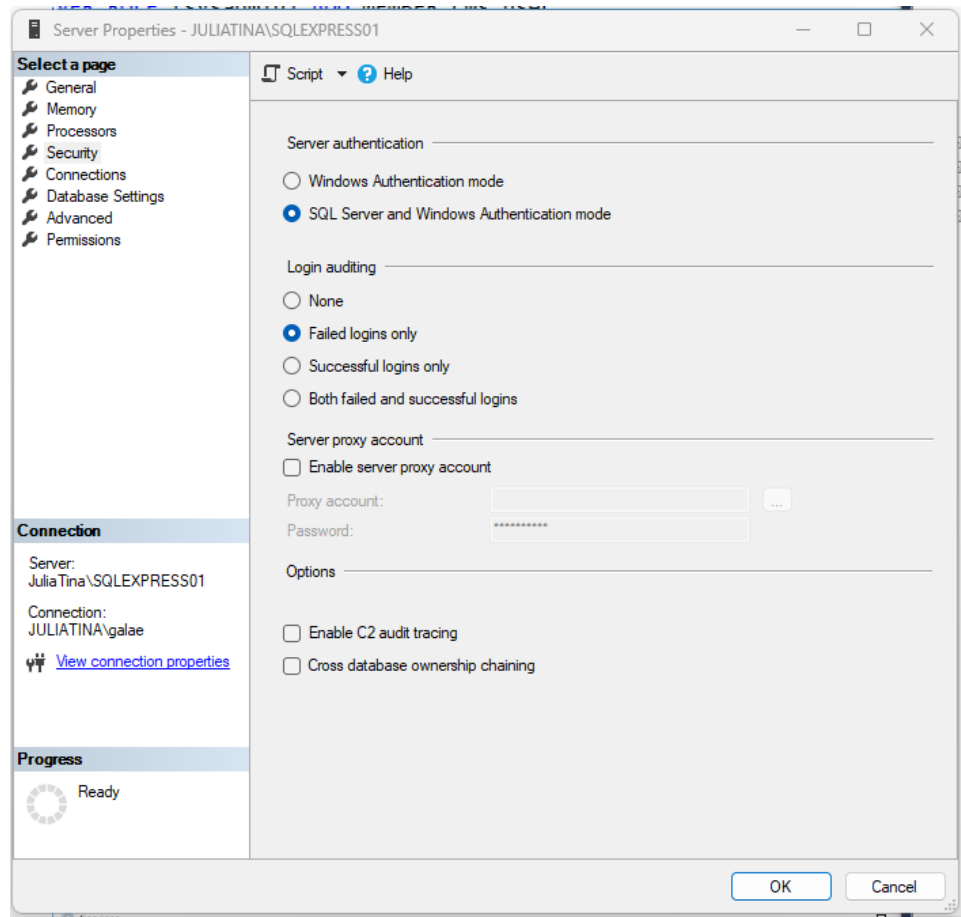
11. Click OK.



Result



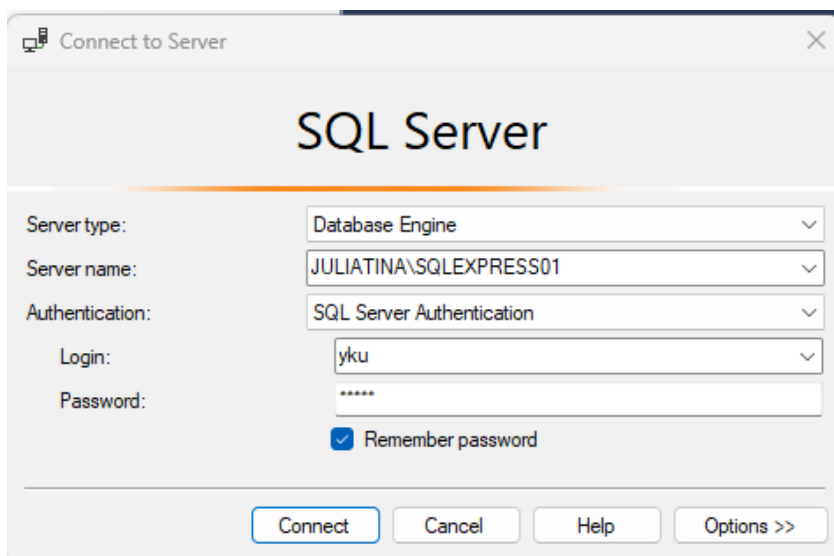
1. Using admin account right click on your server name (SQLEXPRESS01), go to properties
- 2 - Go to option **Security**
- 3 - Check the option "**SQL Server and Windows Authentication mode**"
- 4 - Click on the **Ok** button



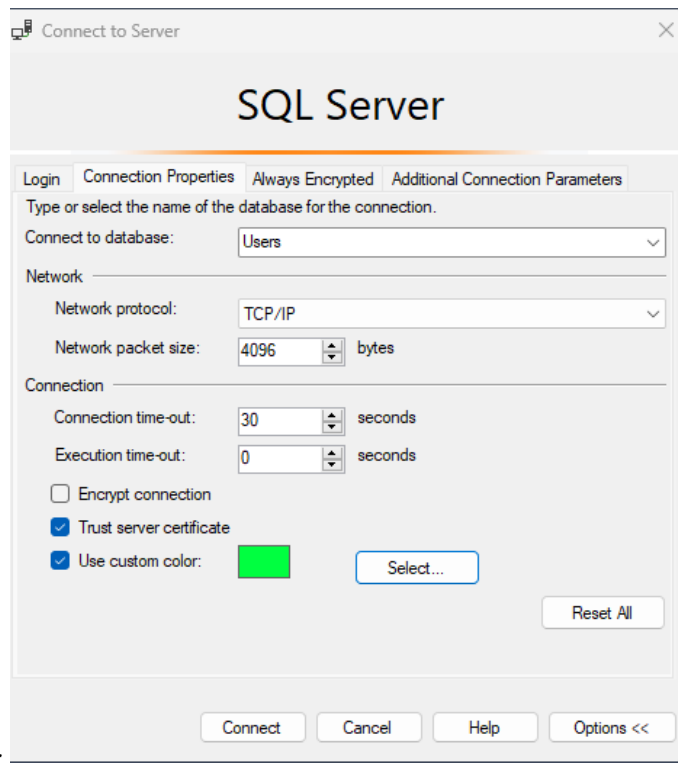
Restart the server

### 3.4. Try to connect with the new User.

Make services automatic, stop and start them. They should run

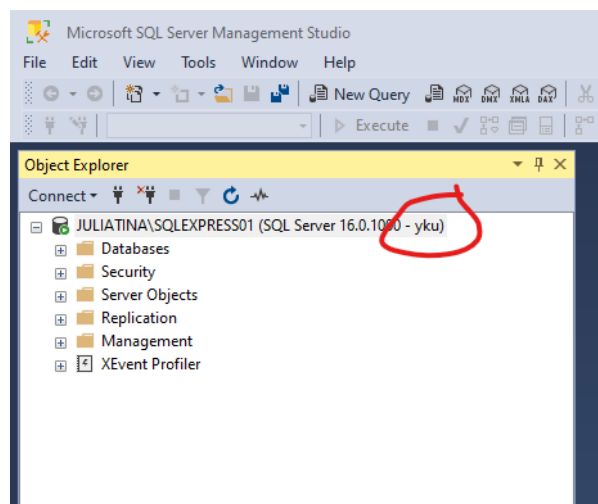


Go to **Options** ,



choose your favorite color

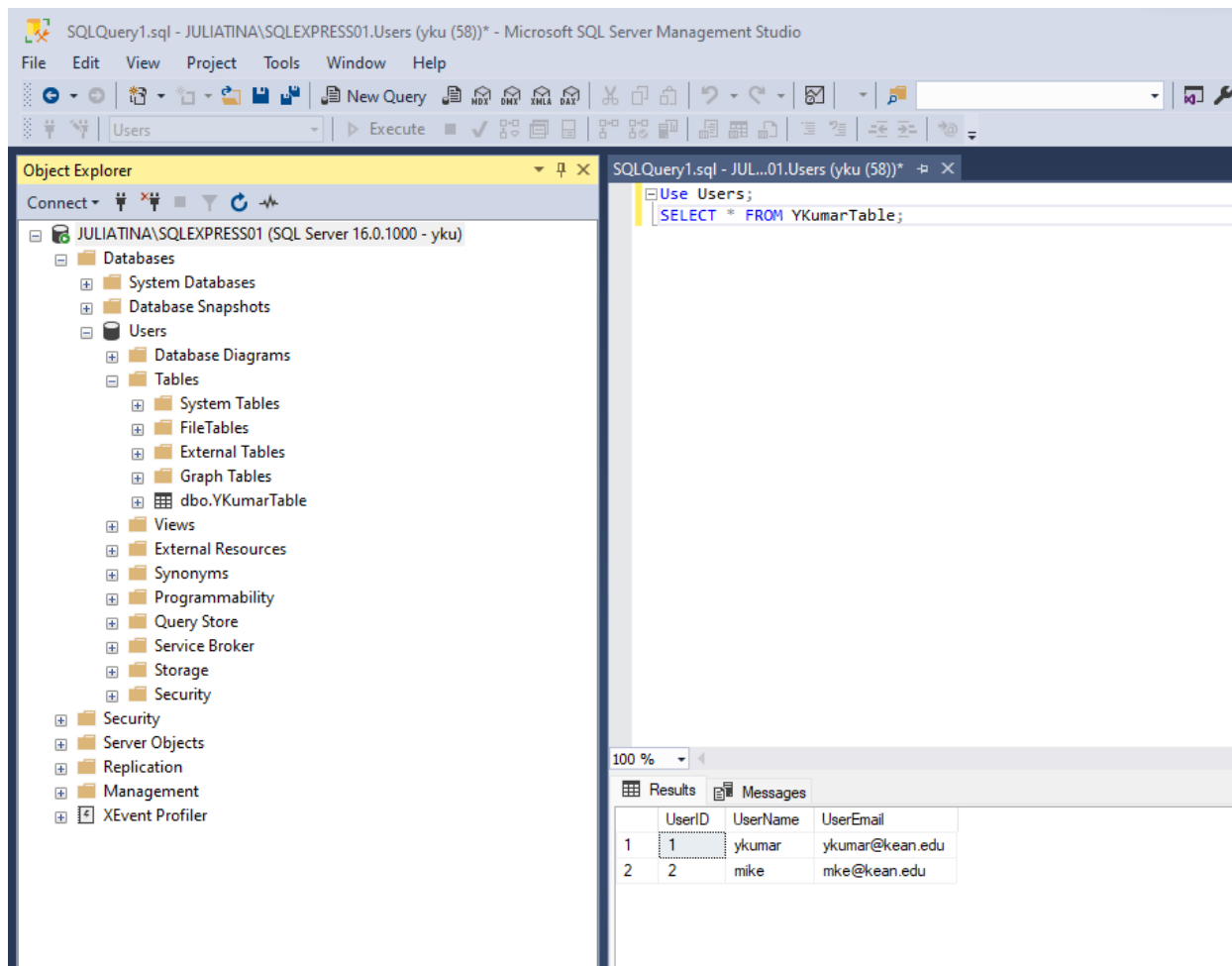
Click Connect



Test select

**Use Users;**

**SELECT \* FROM YKumarTable;**



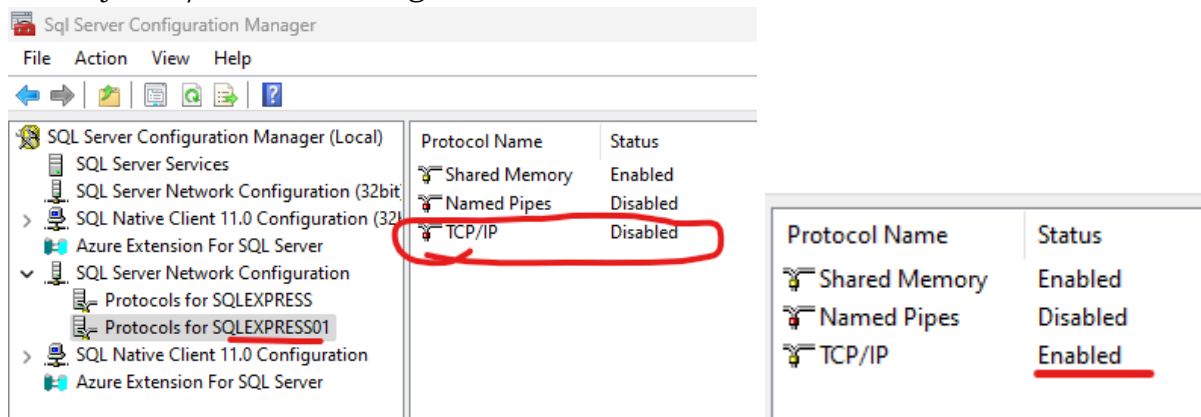
**TASK 4: Take a screenshot of your result. The user should have your (your teammate username).**

**Possible troubleshooting / in case of error:**

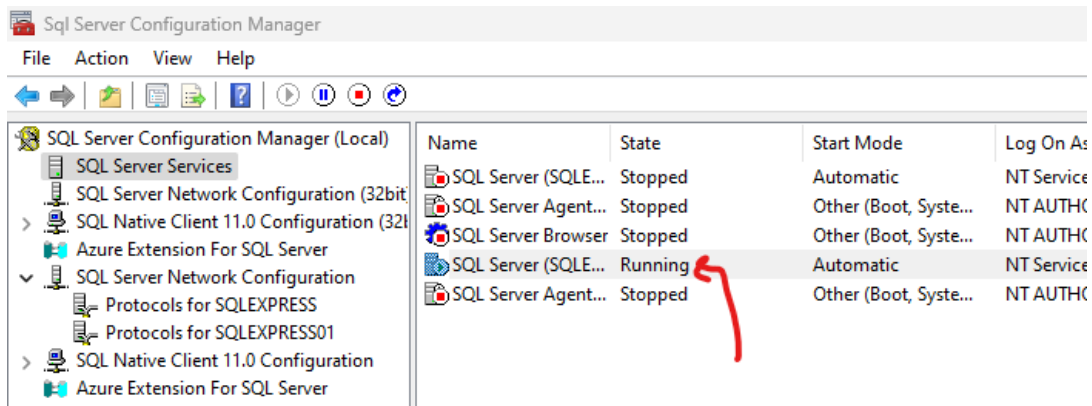
**Make sure that the server, Agent and Browser are all running, their start mode is automatic, TCP/IP is enabled.**

Close the studio, open **SSMS** as administrator.

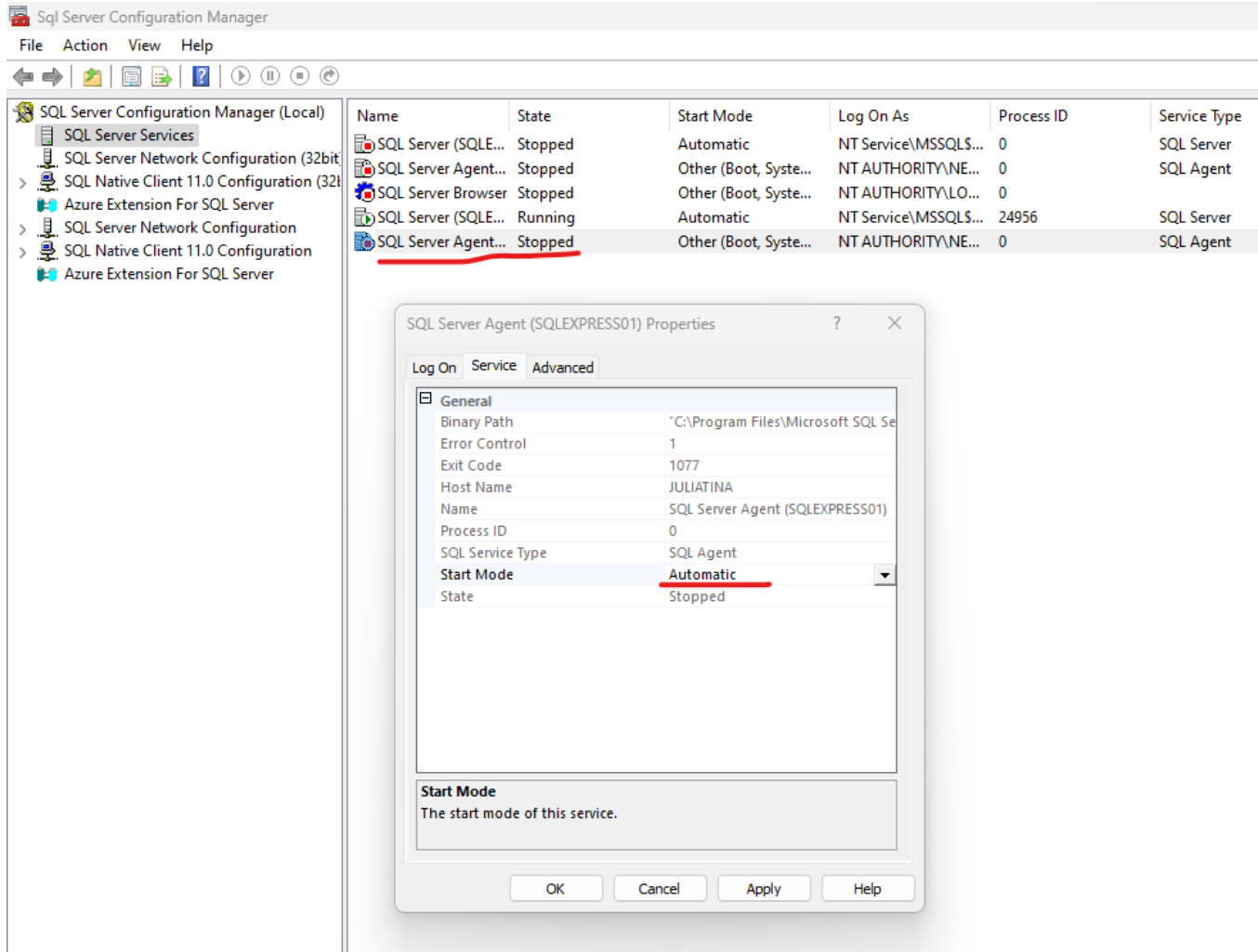
Initially TCP/IP disabled. Right-click - enable

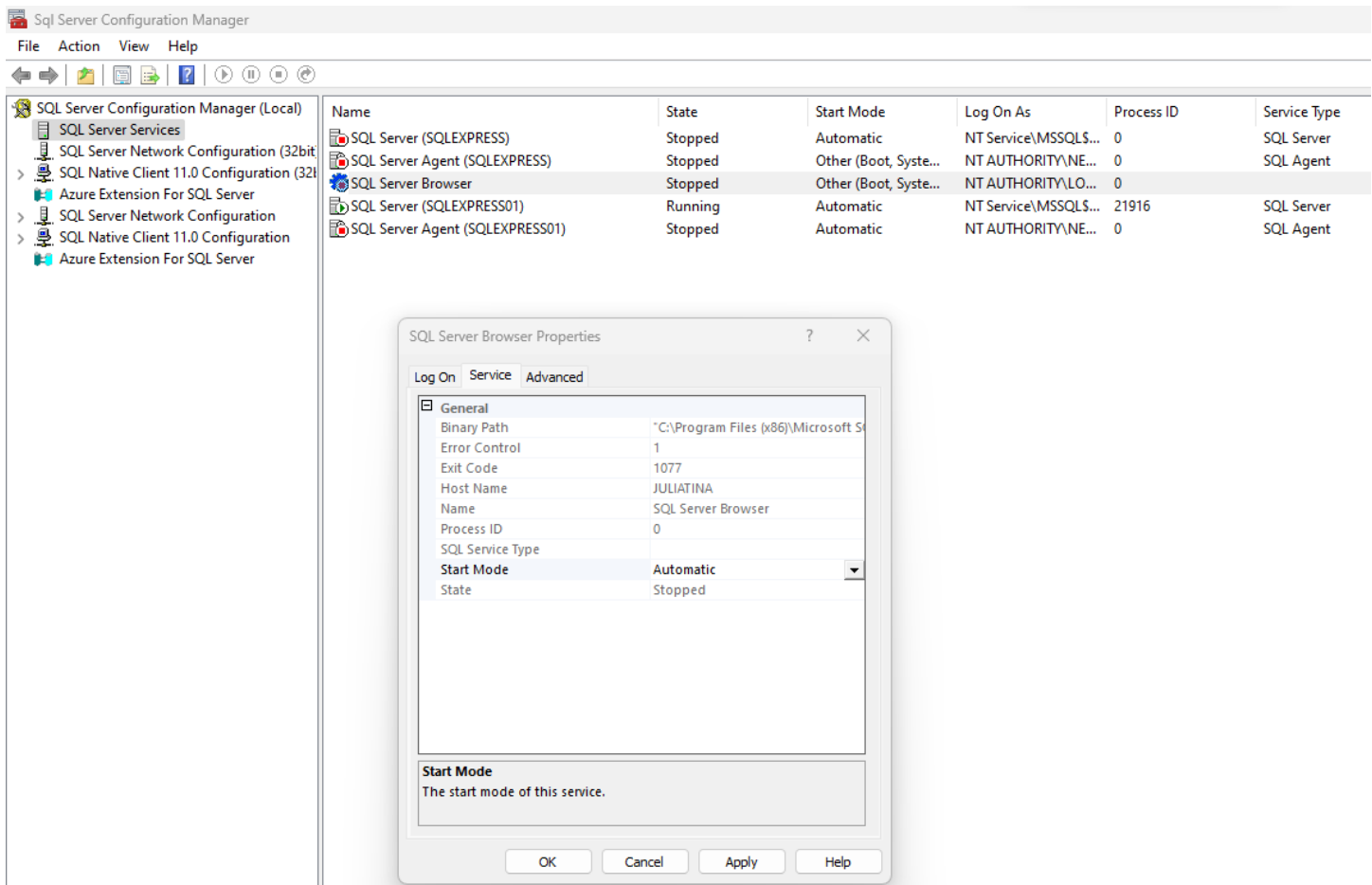






Find the server, **Start** or right-click - > **Stop**, then **Start**. You might need to change start mode to Automatic.



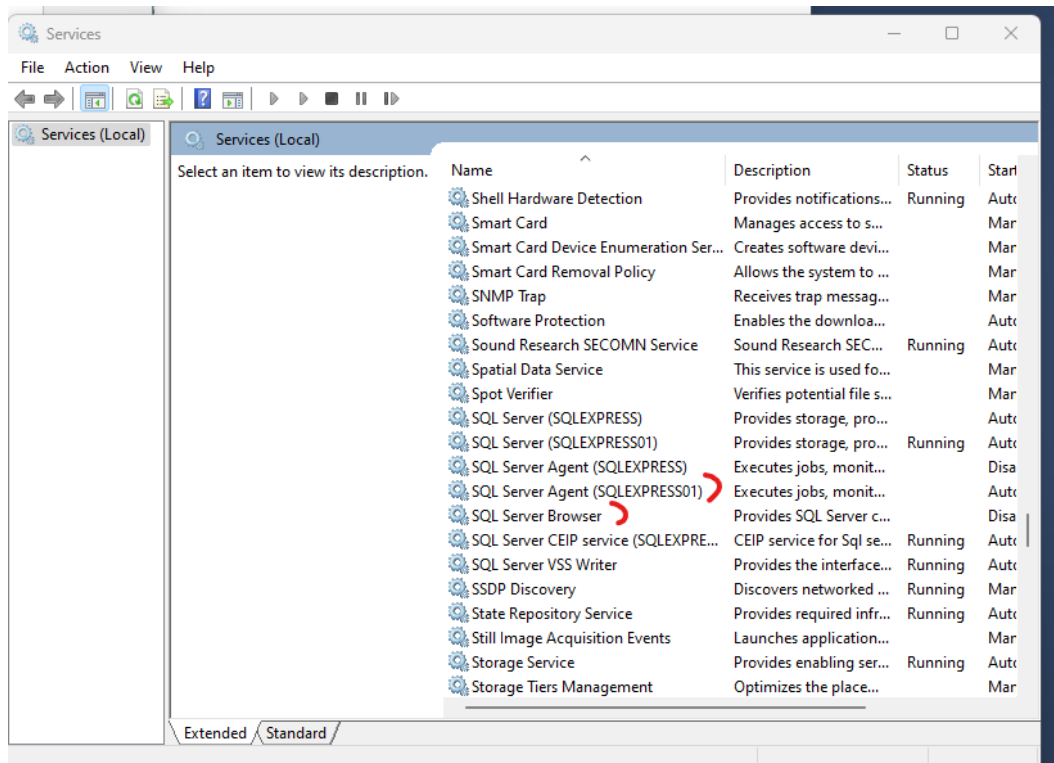


Reference: <https://learn.microsoft.com/en-us/sql/tools/configuration-manager/sql-server-browser-service?view=sql-server-ver16>

### If the server / agent or browser do not still start:

If it is disabled, go to **Control Panel->Administrative Tools->Services**, and look for the **SQL Server Agent**. Right-click, and select **Properties** From the **Startup Type** dropdown, change from **Disabled** to **Automatic**.

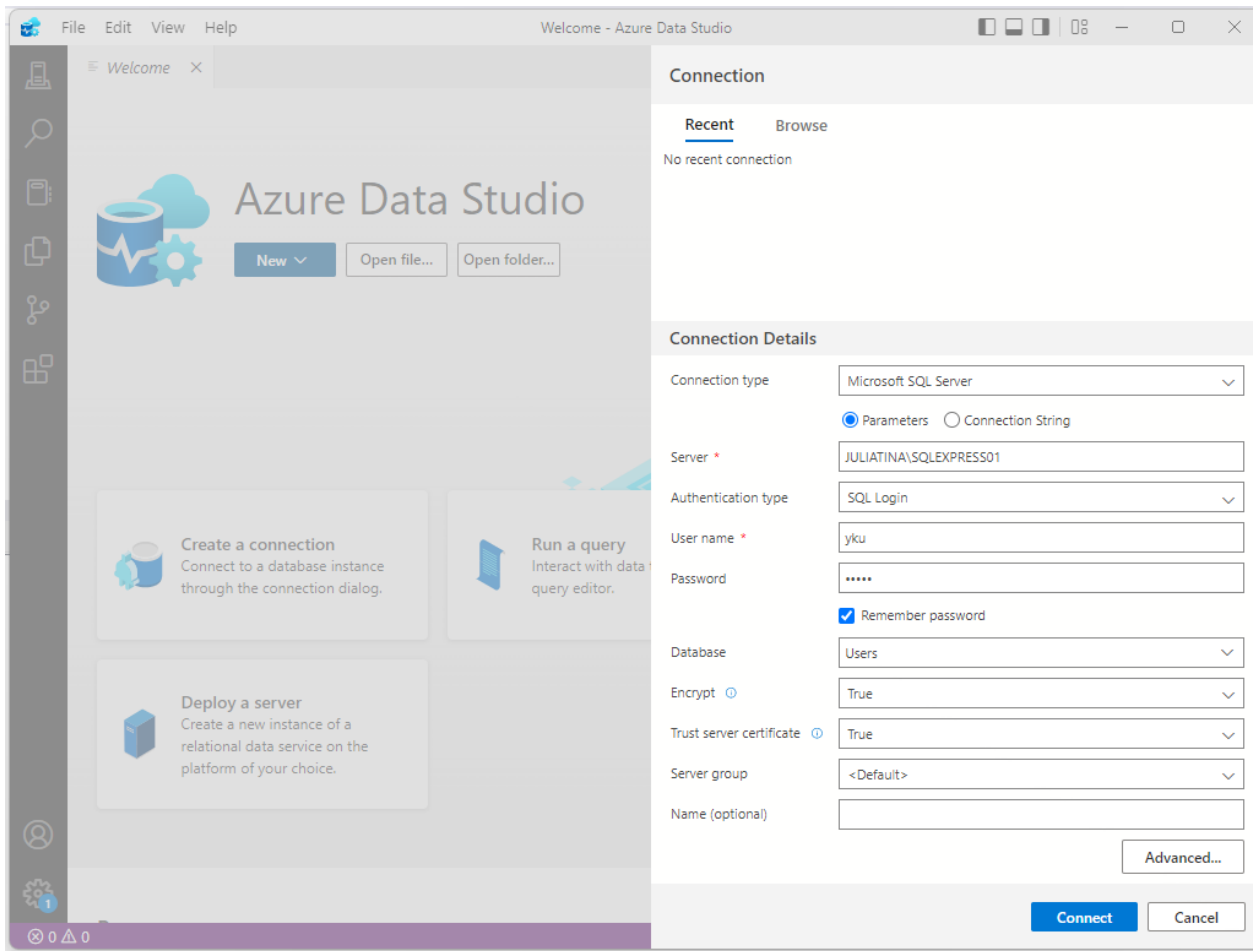
Search for "**Services**" on your computer

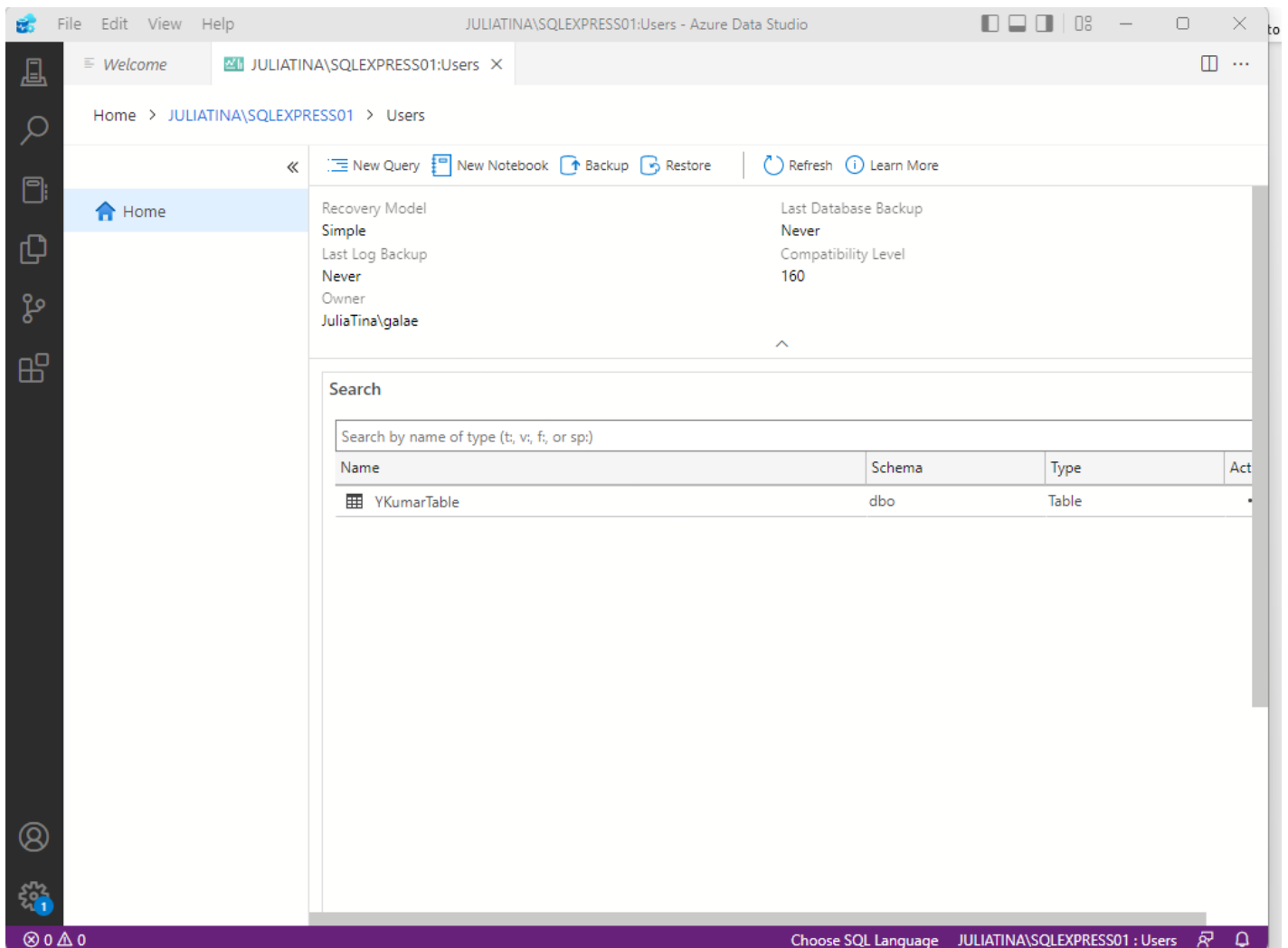


Make services automatic, stop and start them. They should run

### 3.4. Try to connect with the new User from Azure Data Studio.

Allow changes.

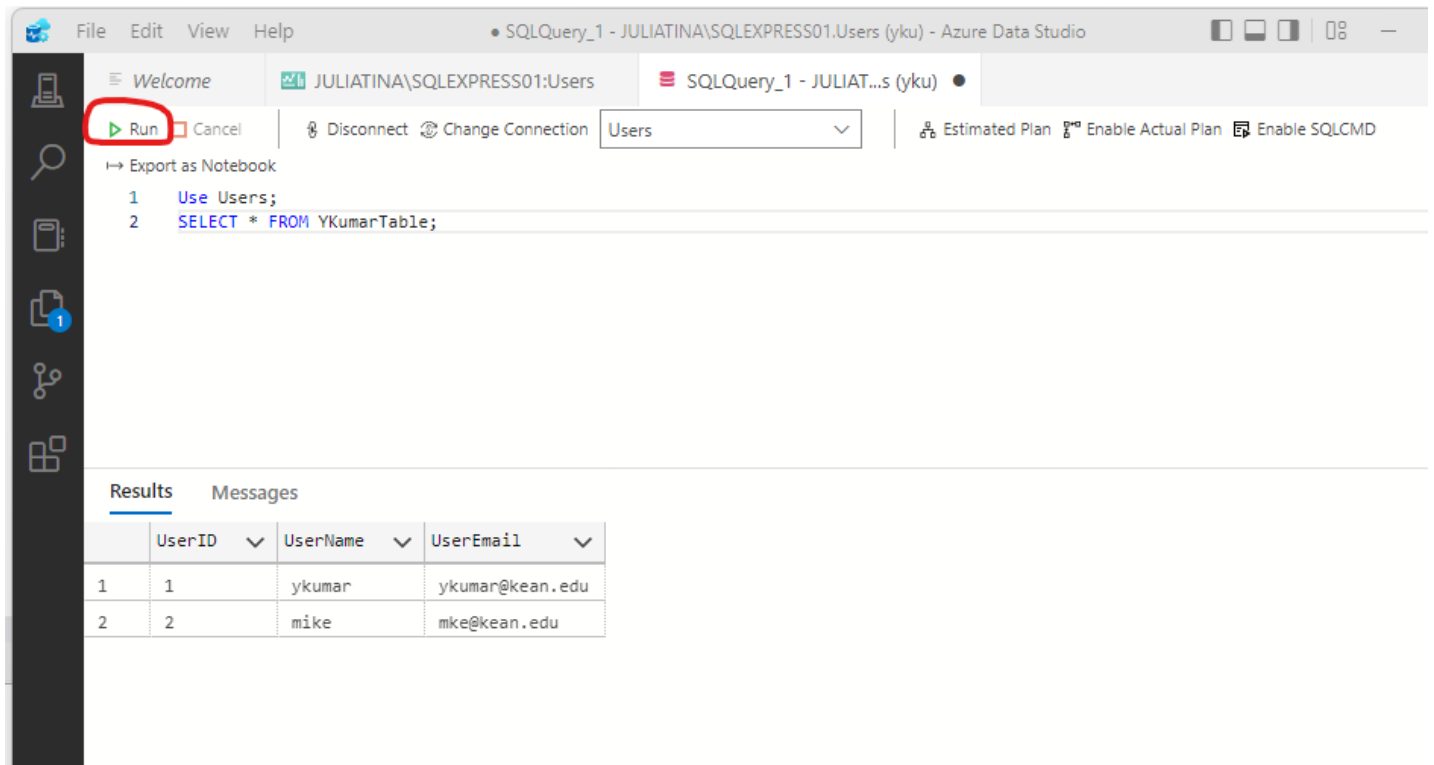




New Query ->Test Select

**Use Users;**

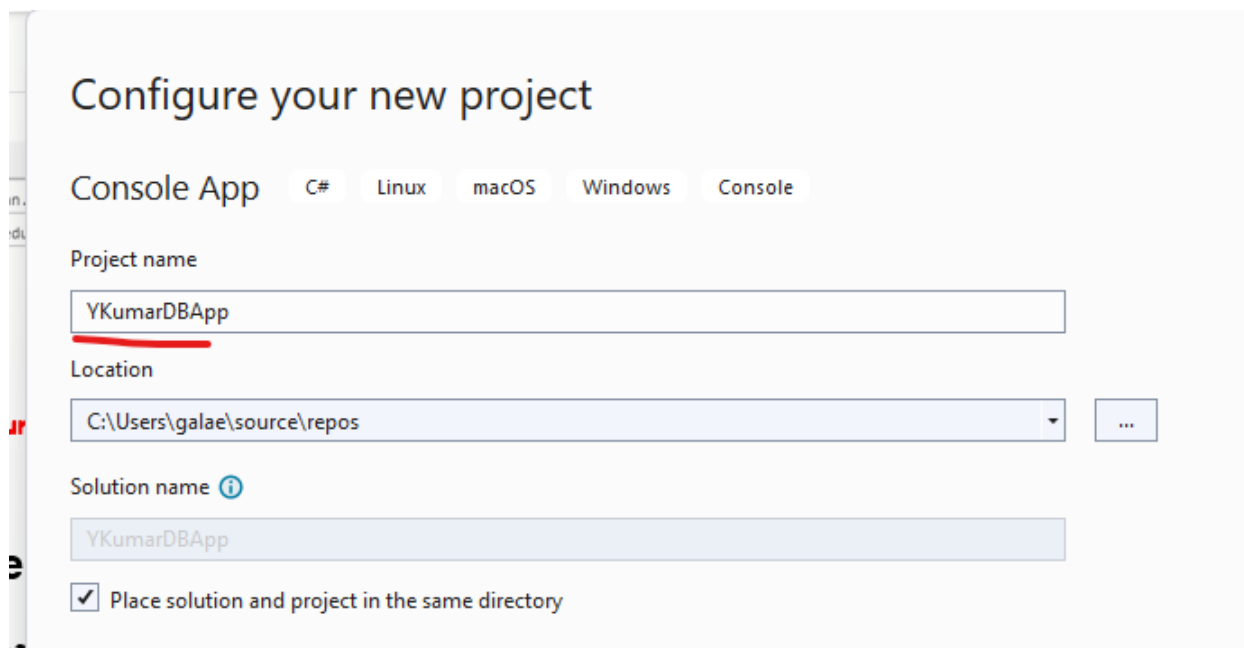
**SELECT \* FROM YKumarTable;**



**TASK 5: Take a screenshot of your result. The user and table should have your (or your teammate username).**

## 4. Connect with C# code

### 4.1. Create a new Console App in Visual Studio

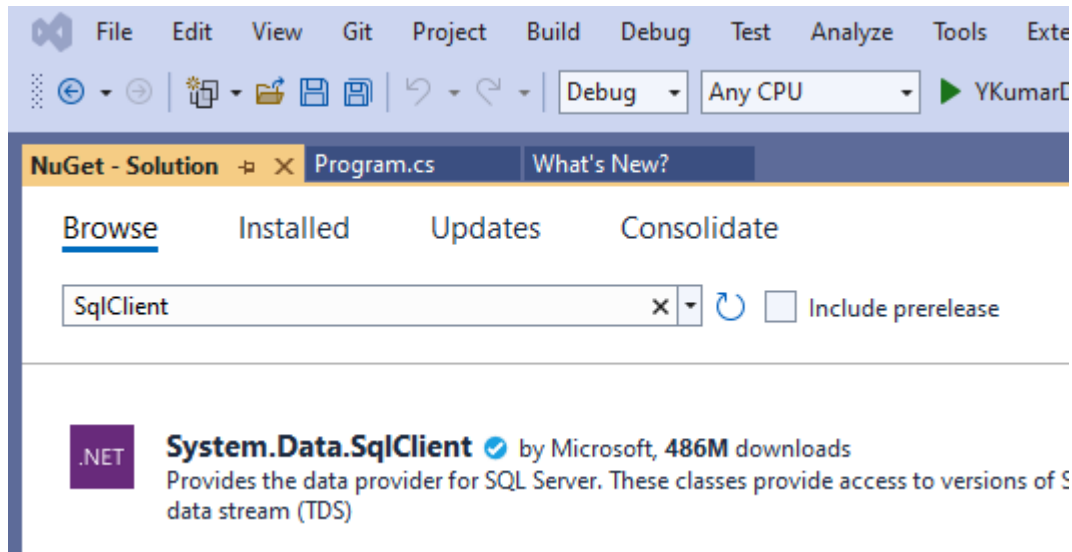


**Use your name.**

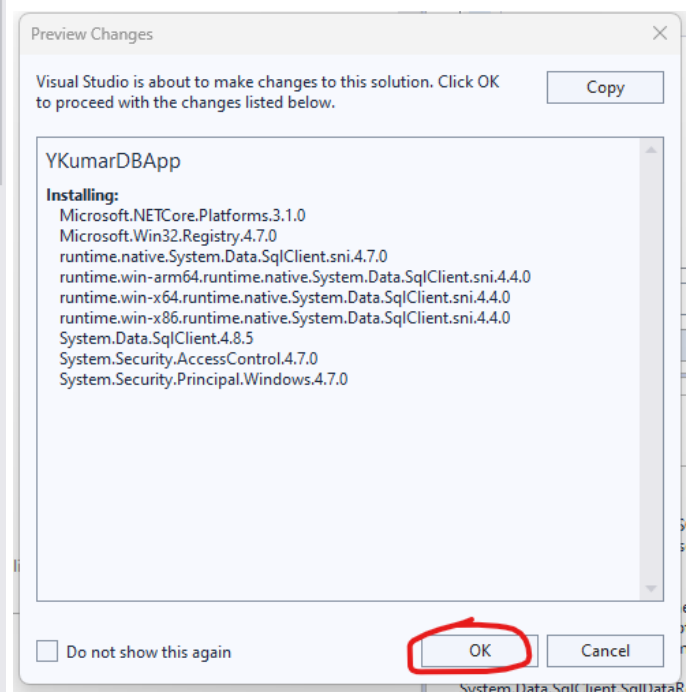
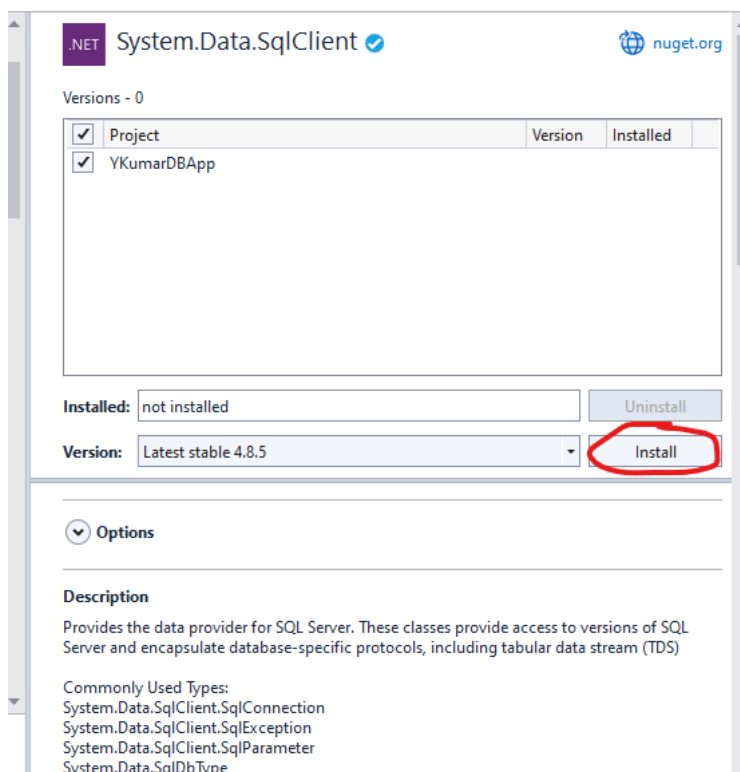
## 4.2. Import Nugget Packages

**Right-click on Solution Explorer** for your project – **Manage Nugget packages.**

Type the name of the assembly and Browse: **SqlClient**



**Choose the package, install.**

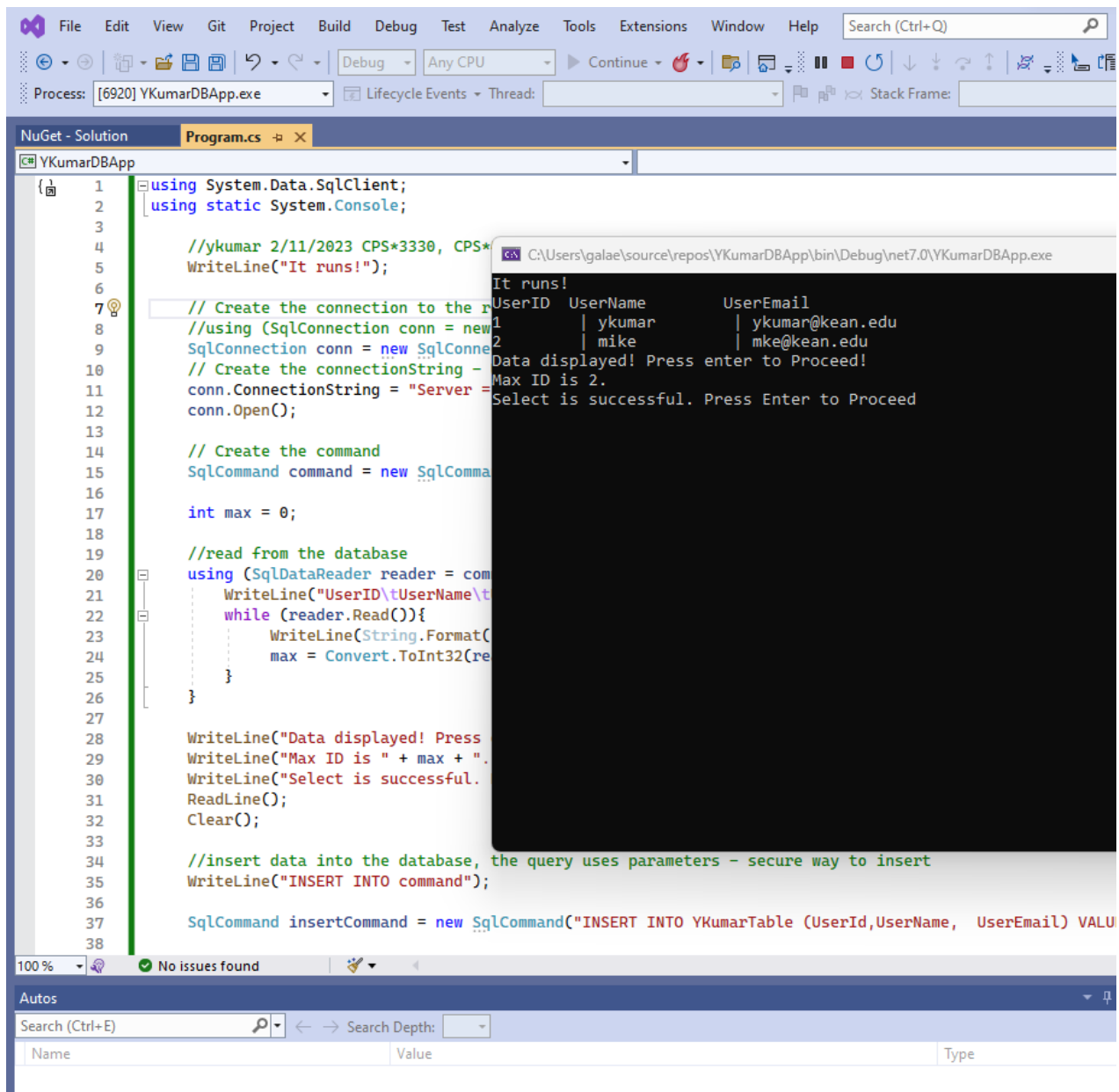


accept

## 4.3. Run the code and debug if needed.

**Copy-paste and run this code:** [http://eve.kean.edu/~ykumar/CPS3330\\_SP2023/SQLserver.txt](http://eve.kean.edu/~ykumar/CPS3330_SP2023/SQLserver.txt)

## Expected result:



```
using System.Data.SqlClient;
using static System.Console;

//y Kumar 2/11/2023 CPS*3330, CPS*
WriteLine("It runs!");

// Create the connection to the database
//using (SqlConnection conn = new SqlConnection(connectionString))
SqlConnection conn = new SqlConnection(connectionString);
// Create the connection string - server name, database name, user name, password
conn.ConnectionString = "Server=.;Database=YKumarDB;User ID=sa;Password=1qaz!@WSX;";
conn.Open();

// Create the command
SqlCommand command = new SqlCommand("SELECT * FROM YKumarTable", conn);

int max = 0;

//read from the database
using (SqlDataReader reader = command.ExecuteReader())
{
    WriteLine("UserID\tUserName\tUserEmail");
    while (reader.Read())
    {
        WriteLine(String.Format("{0}\t{1}\t{2}", reader.GetString(0), reader.GetString(1), reader.GetString(2)));
        max = Convert.ToInt32(reader.GetInt32(3));
    }
}

WriteLine("Data displayed! Press Enter to Proceed!");
WriteLine("Max ID is " + max + ".");
WriteLine("Select is successful. Press Enter to Proceed!");
ReadLine();
Clear();

//insert data into the database, the query uses parameters - secure way to insert
WriteLine("INSERT INTO command");

SqlCommand insertCommand = new SqlCommand("INSERT INTO YKumarTable (UserId,UserName, UserEmail) VALUES (@UserId,@UserName,@UserEmail)", conn);
insertCommand.Parameters.AddWithValue("@UserId", 1);
insertCommand.Parameters.AddWithValue("@UserName", "y Kumar");
insertCommand.Parameters.AddWithValue("@UserEmail", "y.kumar@kean.edu");
insertCommand.ExecuteNonQuery();

WriteLine("Commands executed! Total rows affected are " + insertCommand.ExecuteNonQuery() + ".");
WriteLine("Insert is successful. Press Enter to Proceed!");
ReadLine();
Clear();
```

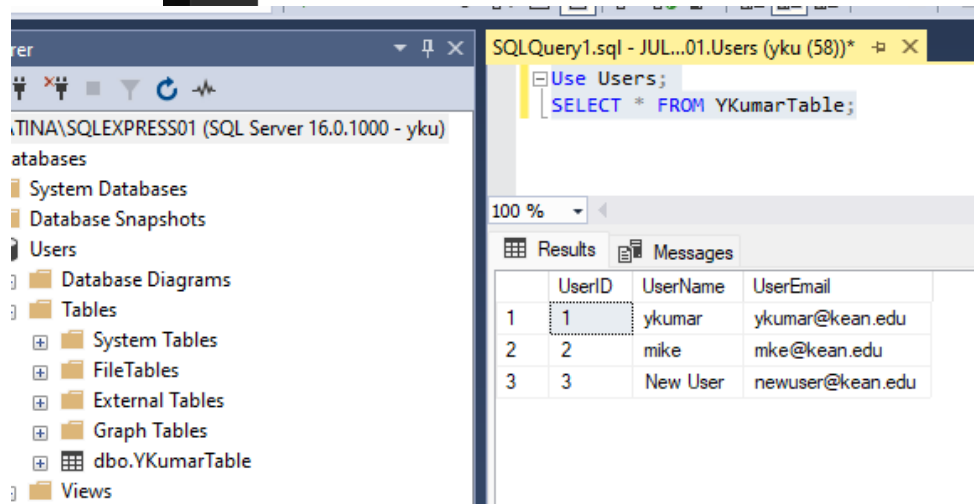
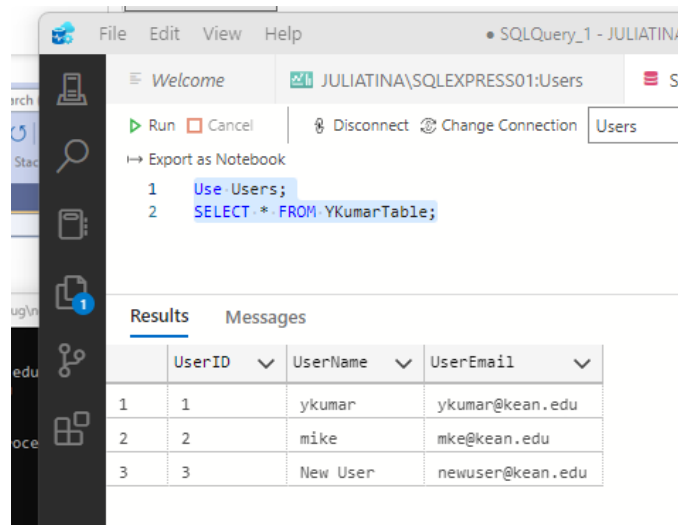
```
C:\Users\galae\source\repos\YKumarDBApp\bin\Debug\net7.0\YKumarDBApp.exe
INSERT INTO command
Commands executed! Total rows affected are 1
Insert is successful. Press Enter to Proceed
```

press Enter

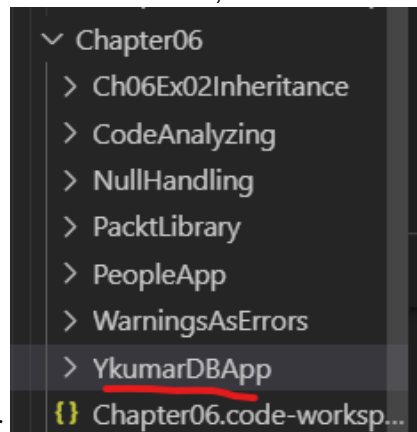
```
C:\Users\galae\source\repos\YKumarDBApp\bin\Debug\net7.0\YKumarDBApp.exe
Now the error trial! Press Enter to Complete.
All done! Good Job!
```

Check changes in the Database:





See, if the code runs in VSC:



Create new folder right-click, open in integrated terminal,

```
PS C:\Users\galae\Repos\cs11dotnet7\vscode\Chapter06\YkumarDBApp> dotnet new console
```

Stand on Program.cs in the created project, right-click, open in integrated terminal. Type the commands

```
dotnet tool install --global dotnet-ef
dotnet add package Microsoft.EntityFrameworkCore.SqlServer
dotnet add package Microsoft.EntityFrameworkCore.Design
dotnet add package Microsoft.EntityFrameworkCore.Tools
```

```
PS C:\Users\galae\Repos\cs11dotnet7\vscode\Chapter06\YkumarDBApp> dotnet tool install --global dotnet-ef
You can invoke the tool using the following command: dotnet-ef
Tool 'dotnet-ef' (version '7.0.2') was successfully installed.
PS C:\Users\galae\Repos\cs11dotnet7\vscode\Chapter06\YkumarDBApp> dotnet add package Microsoft.EntityFrameworkCore.SqlServer
Determining projects to restore...
Writing C:\Users\galae\AppData\Local\Temp\tmp41EE.tmp
...X 500...First check will use the default trust store located by .NET
```

Try the code:

We need to slightly update 2 lines of code (underlined below)

```
1 using Microsoft.Data.SqlClient;
2 using static System.Console;
3
4 //ykumar 2/11/2023 CPS*3330, CPS*4981 - ADD YOUR information here
5 WriteLine("It runs!");
6
7 // Create the connection to the resource!
8 //using (SqlConnection conn = new SqlConnection())
9 SqlConnection conn = new SqlConnection();
10 // Create the connectionString - USE YOUR CREDENTIALS
11 conn.ConnectionString = "Server = JULIATINA\\SQLEXPRESS01; Database = Users; Trusted_Connection=True;Trust Server Certificate=true";
12 conn.Open();
13
14 // Create the command
15 SqlCommand command = new SqlCommand("SELECT * FROM YkumarTable", conn);
16
17 int max = 0;
18
19 //read from the database
20 using (SqlDataReader reader = command.ExecuteReader()){
21     WriteLine("UserID\tUserName\tUserEmail");
22     while (reader.Read()){
23         WriteLine(String.Format("{0} \t | {1} \t | {2}", reader[0], reader[1], reader[2]));
24         max = Convert.ToInt32(reader[0].ToString());//yk
25     }
26 }
```

```
at Program.<Main>$(String[] args) in C:\Users\galae\Repos\cs11dotnet7\vscode\Chapter06\YkumarDBApp\Program.cs:line 12
ClientConnectionId:c0c84fcf-95af-45df-be28-916fe388a019
Error: Number: -2146893019,State:0,Class:20
PS C:\Users\galae\Repos\cs11dotnet7\vscode\Chapter06\YkumarDBApp> dotnet run
It runs!
UserID  UserName      UserEmail
1       ykumar        ykumar@kean.edu
2       mike          mke@kean.edu
3       New User     newuser@kean.edu
Data displayed! Press enter to Proceed!
Max ID is 3.
Select is successful. Press Enter to Proceed
```

TASK 6: Take a screenshot of your results from both IDEs (VSC and Visual studio) and from the Database. The table should have your (or your teammate username).

## 5. Entity Framework Set up

Entity Framework (EF) Core is a lightweight, extensible, open source and cross-platform version of the popular Entity Framework data access technology.

EF Core can serve as an object-relational mapper (O/RM), which:

Enables .NET developers to work with a database using .NET objects.

Eliminates the need for most of the data-access code that typically needs to be written.

Reference: <https://learn.microsoft.com/en-us/ef/core/>

## Development Approaches with Entity Framework

There are three different approaches you can use while developing your application using Entity Framework:

1. Database-First
2. Code-First
3. Model-First

### Database-First Approach

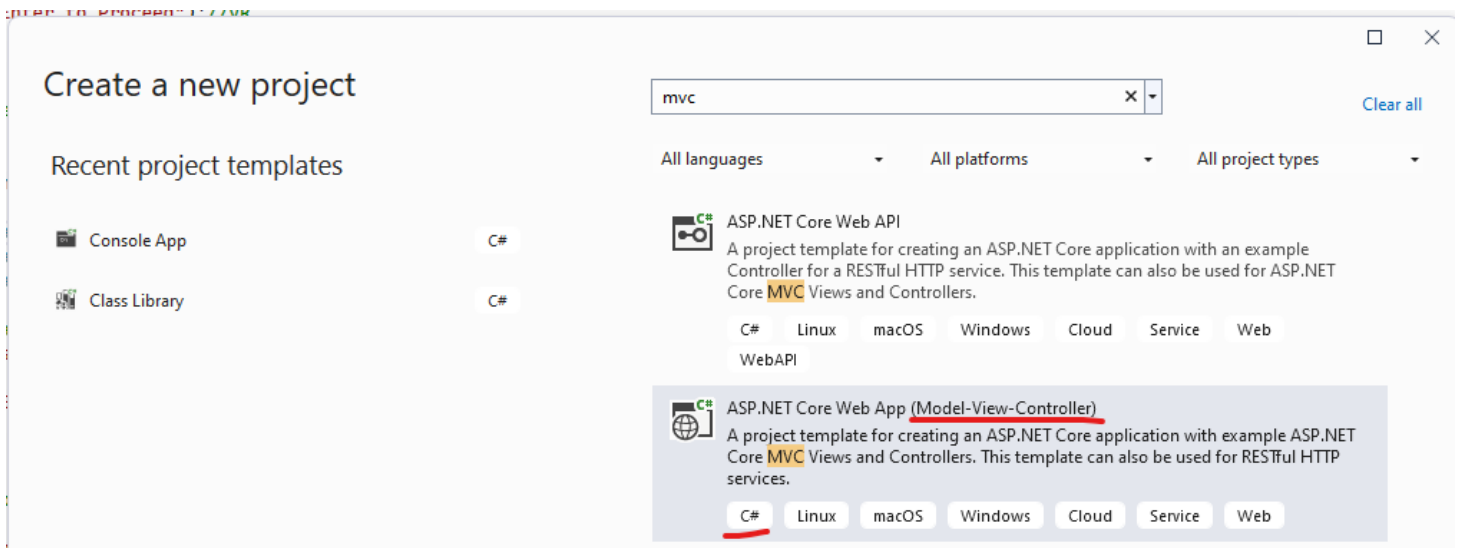
In the database-first development approach, you generate the context and entities for the existing database using EDM wizard integrated in Visual Studio or executing EF commands.



In simple words you are creating OOP code (entities with their properties) automatically based on Database data

**Pretty much it will automatically create an OOP code for you if you have a database or will create a database for you if you have an OOP code**

### 5.1. Create a new MVC Web App in Visual Studio



**Use your name to name the app.**

ENTER TO PROCEED" 17/1/2024

## Additional information

ASP.NET Core Web App (Model-View-Controller) C# Linux macOS Windows

Framework ⓘ  
[.NET 7.0 (Standard Term Support)]

Authentication type ⓘ  
[None]


☒ Configure for HTTPS ⓘ  
☐ Enable Docker ⓘ

Docker OS ⓘ  
[Linux]

☒ Do not use top-level statements ⓘ

Run the project, the authentication / SSLmessage below might be displayed: say Yes

Trust ASP.NET Core SSL Certificate X

 This project is configured to use SSL. To avoid SSL warnings in the browser you can choose to trust the self-signed certificate that ASP.NET Core has generated.

Would you like to trust the ASP.NET Core SSL certificate?

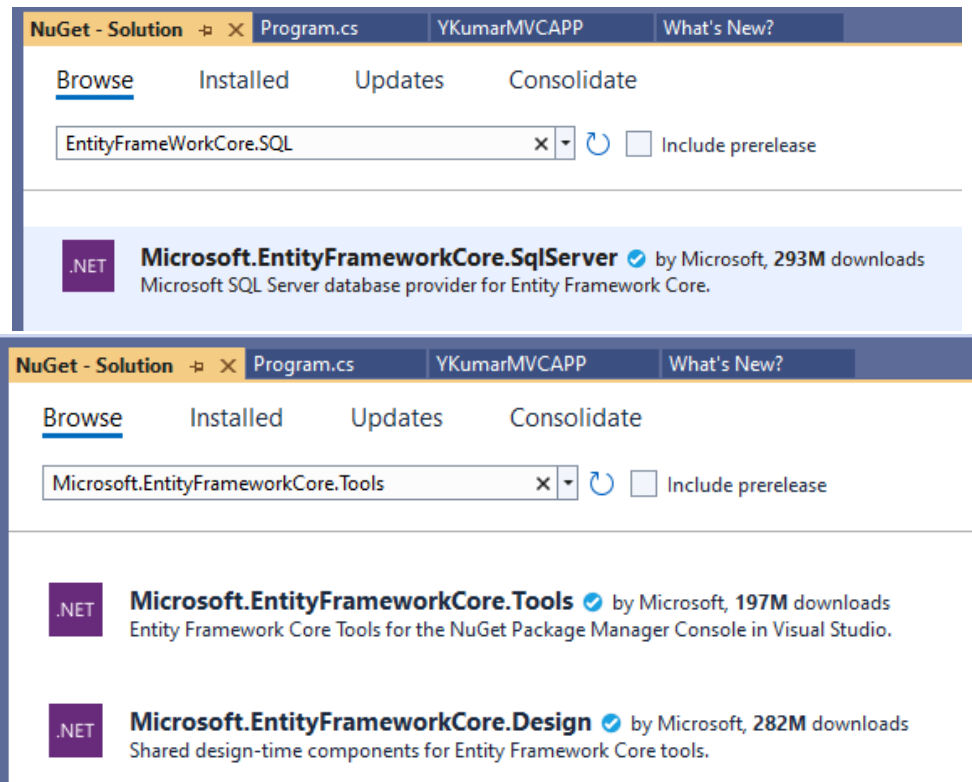
☐ Don't ask me again



## 5.2. Add Nugget packages.

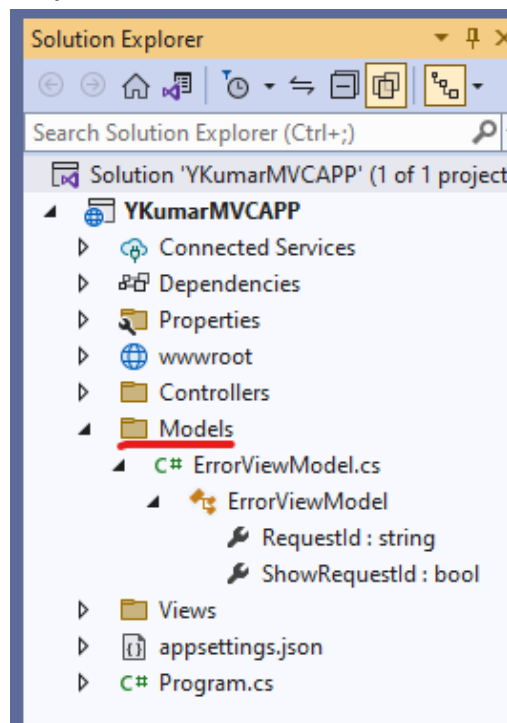
**Right-click on Solution Explorer** for your project – **Manage Nugget packages.**

Browse for **Entity Framework Core SQLServer**, install, accept.  
And **Microsoft.EntityFrameworkCore.Tools**, and **.Design** install, accept

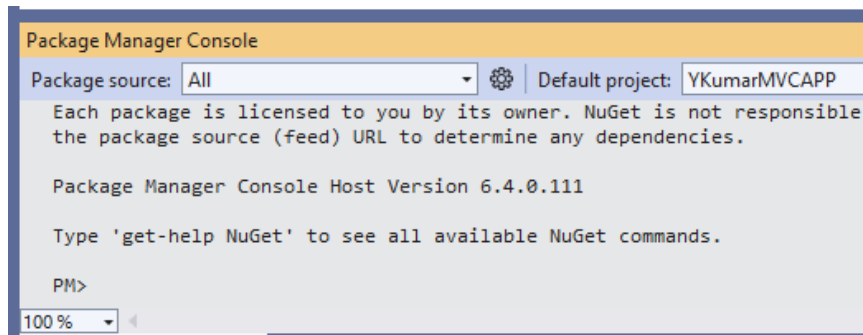


### 5.3. Create a new Entity.

Observe the model's folders in Solution Explorer



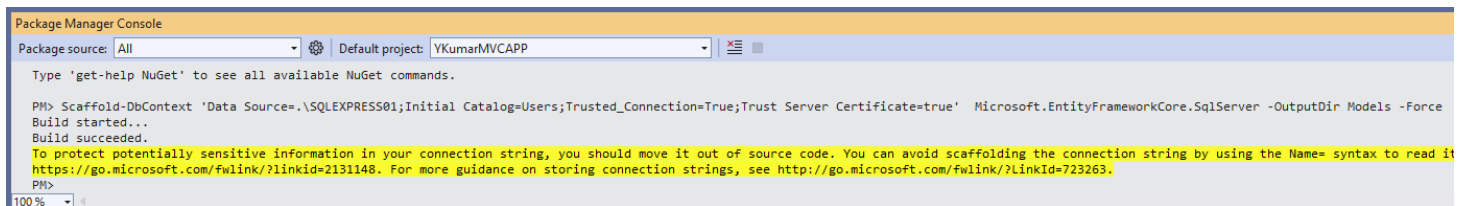
Go to **Tools** → **Nugget Package Manager** → **Package Manager Console**



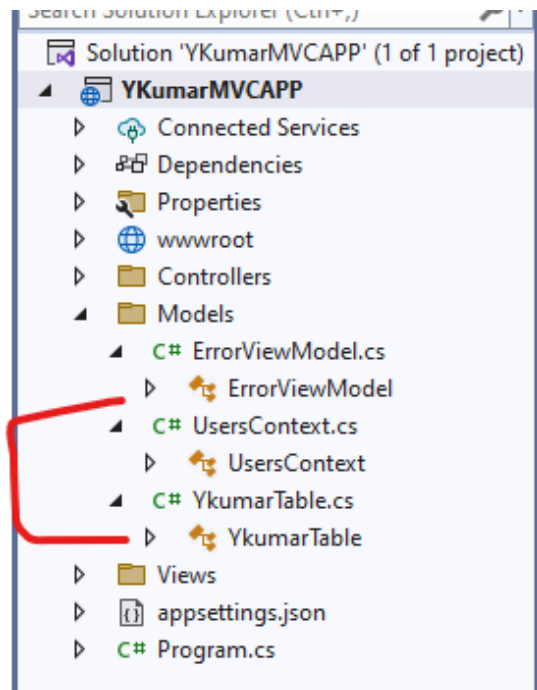
Copy-paste there the command below as is:

**Scaffold-DbContext 'Data Source=.\SQLEXPRESS01;Initial Catalog=Users;Trusted\_Connection=True;Trust Server Certificate=true' Microsoft.EntityFrameworkCore.SqlServer -OutputDir Models -Force**

Press Enter



Observe the changes in the Model folder.

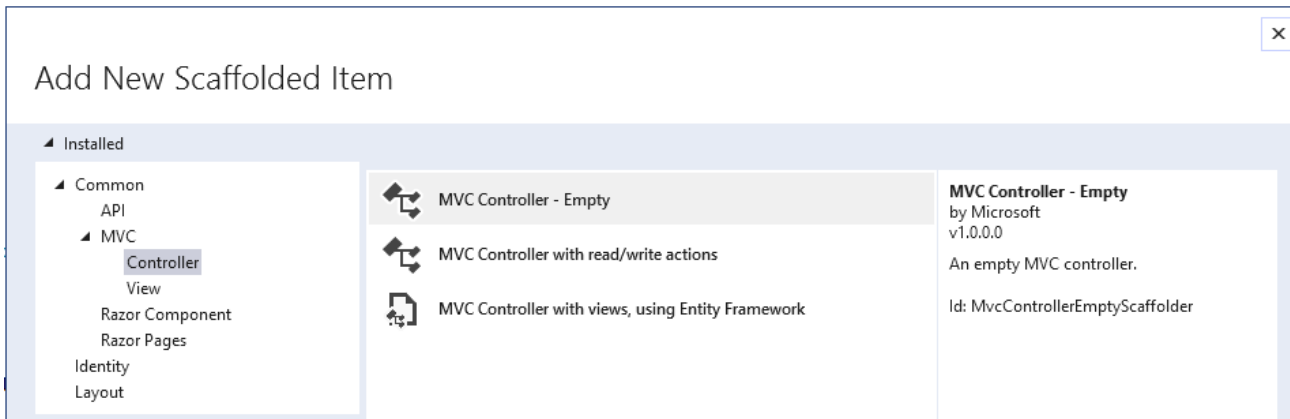


**Take a screenshot.**

## 5.4. Add new Controller and a View.

Right-click on **Controllers** folder and **Add Controller**

> **Controller** → **MVC controller – empty**, call it **UsersController.cs**

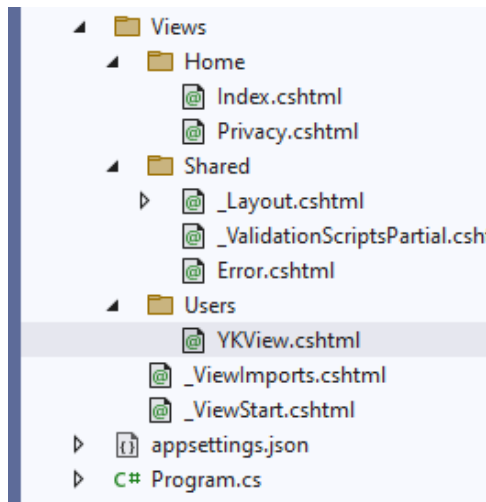


Make the following code adjustments:

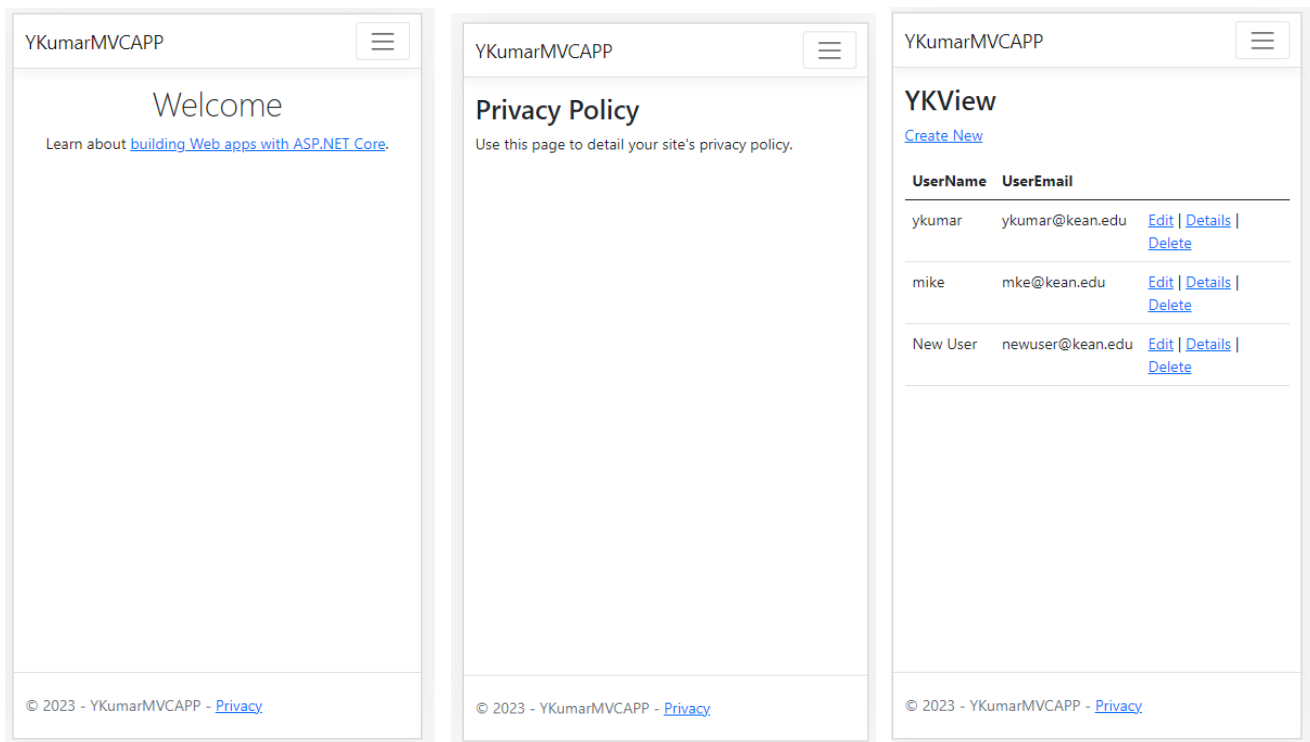
```
Privacy.cshtml | YkumarTable.cs | YKView.cshtml | HomeController.cs | Us...
YKumarMVCAPP
{
1  using Microsoft.AspNetCore.Mvc;
2  using YKumarMVCAPP.Models;
3
4  namespace YKumarMVCAPP.Controllers
5  {
6      0 references
       public class UsersController : Controller
7      {
8          0 references
       UsersController uc = new();
9          0 references
       public IActionResult YKView()
10         {
11             return View(uc.YkumarTables.ToList());
12         }
13     }
14 }
15 }
```

Right-click on **IActionResult** on line 9 in the Controller → **Add View**

Observe the View folder:

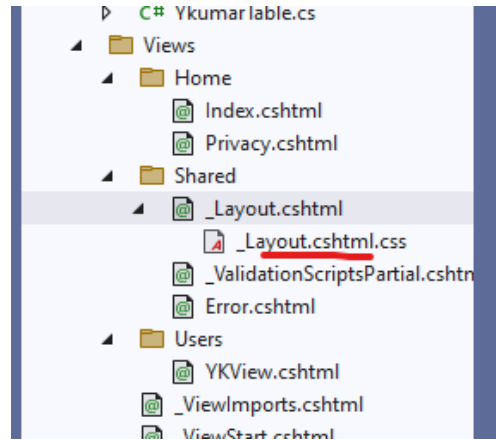


Run the app:



We will add users to the Navbar...  
Navigate to Layout\_cshtml.ss





The code appeared:

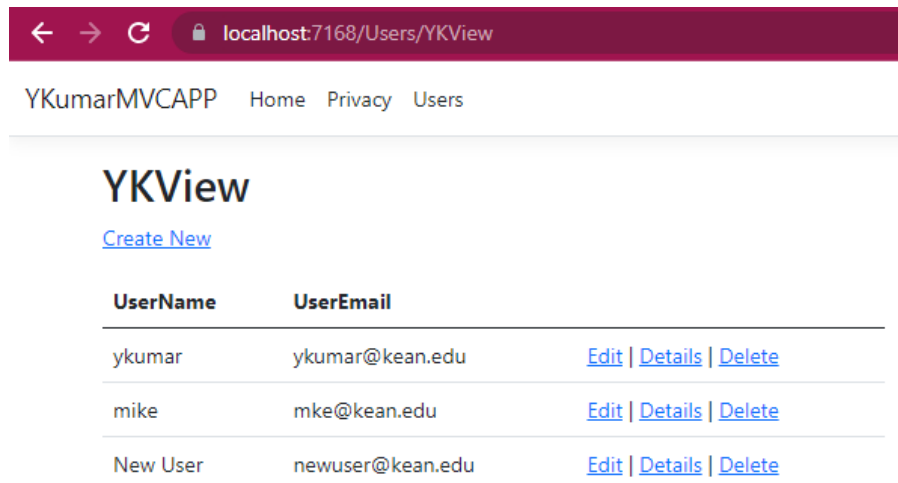
```

11 <body>
12 <header>
13 <nav class="navbar navbar-expand-sm navbar-toggleable-sm navbar-light bg-white border-bottom box-shadow mb-3">
14 <div class="container-fluid">
15 <a class="navbar-brand" asp-area="" asp-controller="Home" asp-action="Index">YKumarMVCAPP</a>
16 <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target=".navbar-collapse" aria-controls="navbarSupportedContent"
17     aria-expanded="false" aria-label="Toggle navigation">
18 <span class="navbar-toggler-icon"></span>
19 </button>
20 <div class="navbar-collapse collapse d-sm-inline-flex justify-content-between">
21 <ul class="navbar-nav flex-grow-1">
22 <li class="nav-item">
23 <a class="nav-link text-dark" asp-area="" asp-controller="Home" asp-action="Index">Home</a>
24 </li>
25 <li class="nav-item">
26 <a class="nav-link text-dark" asp-area="" asp-controller="Home" asp-action="Privacy">Privacy</a>
27 </li>
28 </ul>
29 </div>
30 </div>

```

We will add a link to our page:

**<li class="nav-item"><a class="nav-link text-dark" asp-area="" asp-controller="Users" asp-action="YKView">Users</a></li>**



**TASK 7: Take a screenshot of your results from Entity Framework part (both autogenerated and custom code + the browser output). The MVC APP should have your (or your teammate username). Push your results to GitHub.**