

### Activity No. 8.1 - Securing Databases

**Name:** Efa, Christian  
Guevarra, Hans Angelo  
Mendoza, John Renzo  
Nicolas, Sean Julian  
Vinluan, Armando

**Date:** 21/10/2022

**Section:** CPE21S3

**Instructor:** Dr. Jonathan Vidal Taylar

#### Objectives:

This activity aims to secure databases using different methodologies.

#### Intended Learning Outcomes (ILOs):

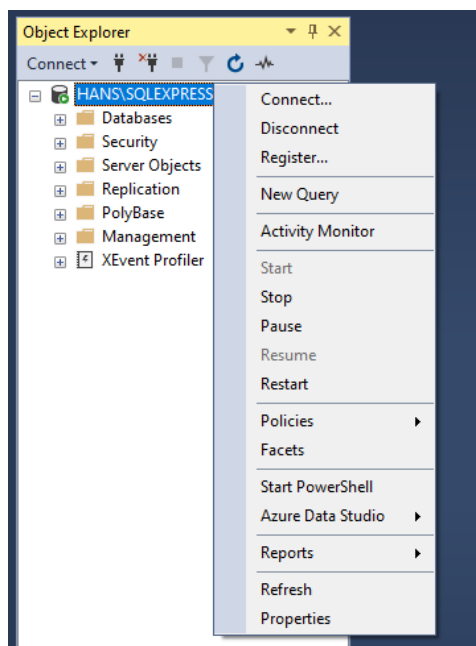
The students should be able to:

1. Configure authentication and authorization of database users.
2. Assign appropriate server and database roles to the users.

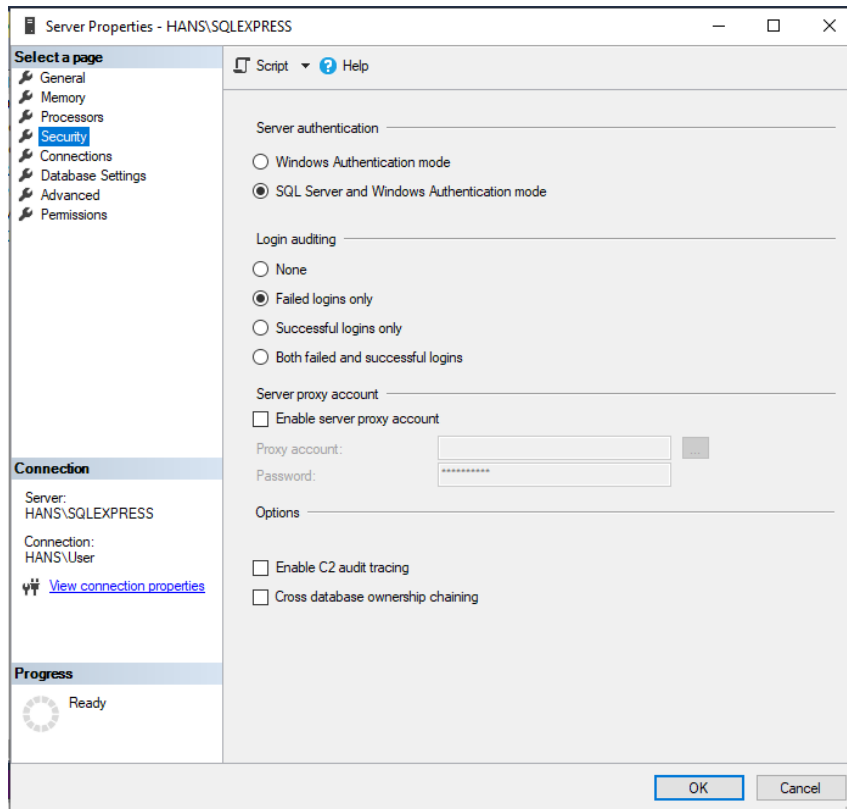
#### Database Output

#### To change security authentication mode

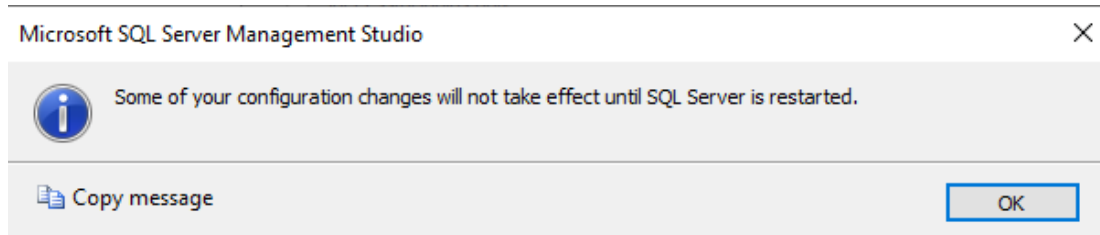
1. In SQL Server Management Studio Object Explorer, right-click the server, and then click Properties.



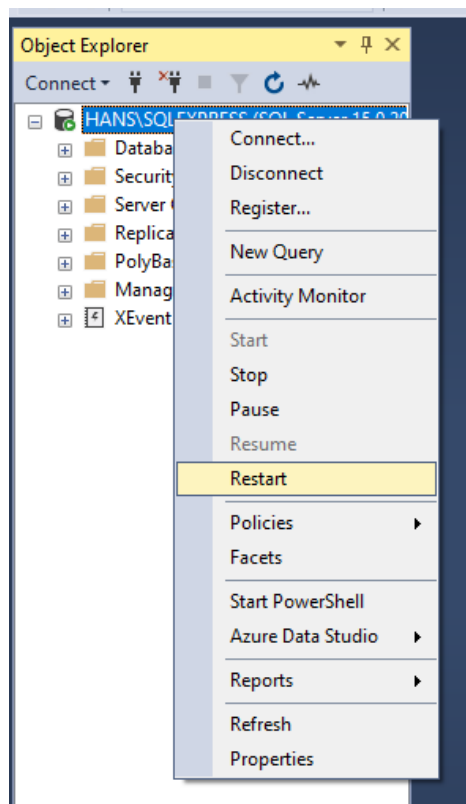
2. On the Security page, under Server authentication, select the new server authentication mode, and then click OK.



3. In the SQL Server Management Studio dialog box, click OK to acknowledge the requirement to restart SQL Server.

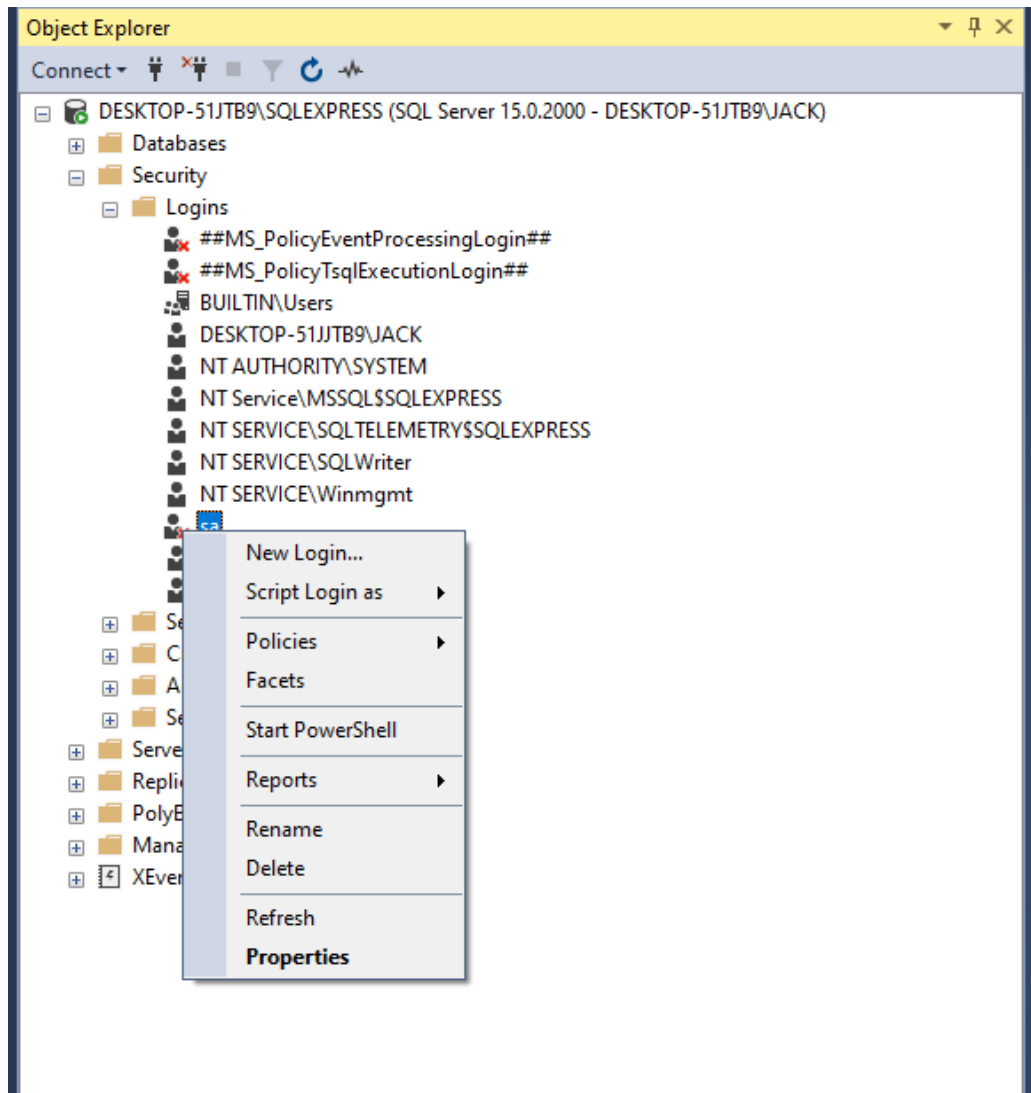


4. In Object Explorer, right-click your server, and then click Restart. If SQL Server Agent is running, it must also be restarted.



## To enable the sa login

Step 1. In Object Explorer, expand Security, expand Logins, right-click sa, and then click Properties.



Step 2. On the General page, you might have to create and confirm a password for the login.

**Login Properties - sa**

Select a page

- General
- Server Roles
- User Mapping
- Status

Script Help

Login name: sa Search...

☐ Windows authentication

☒ SQL Server authentication

Password: .....

Confirm password: .....

☐ Specify old password

Old password: .....

☒ Enforce password policy

☐ Enforce password expiration

☐ User must change password at next login

☐ Mapped to certificate

☐ Mapped to asymmetric key

☐ Map to Credential

Add

Credential	Provider
------------	----------

Remove

Default database: master

Default language: English - us\_english

OK Cancel

**Connection**

Server: DESKTOP-51JTB9\SQLEXPRES

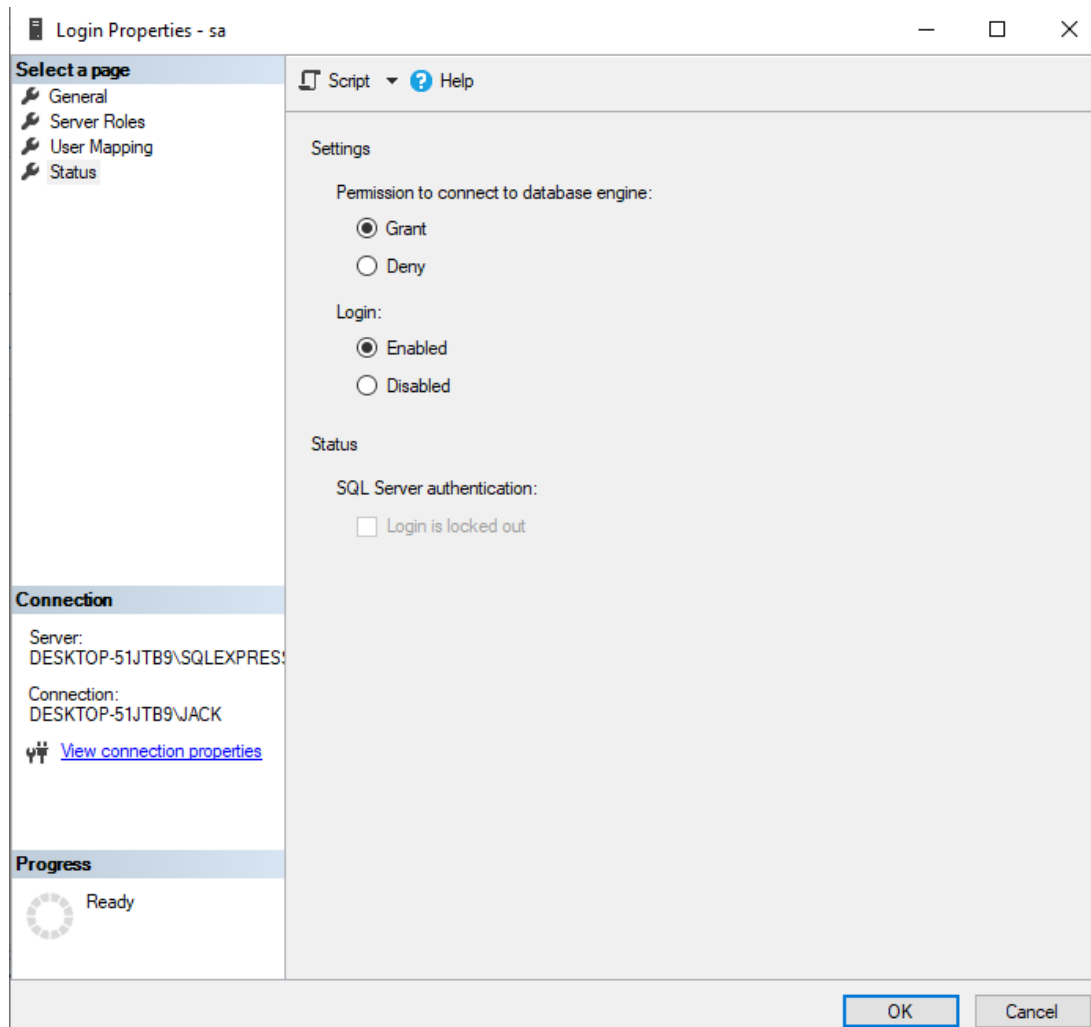
Connection: DESKTOP-51JTB9\JACK

[View connection properties](#)

**Progress**

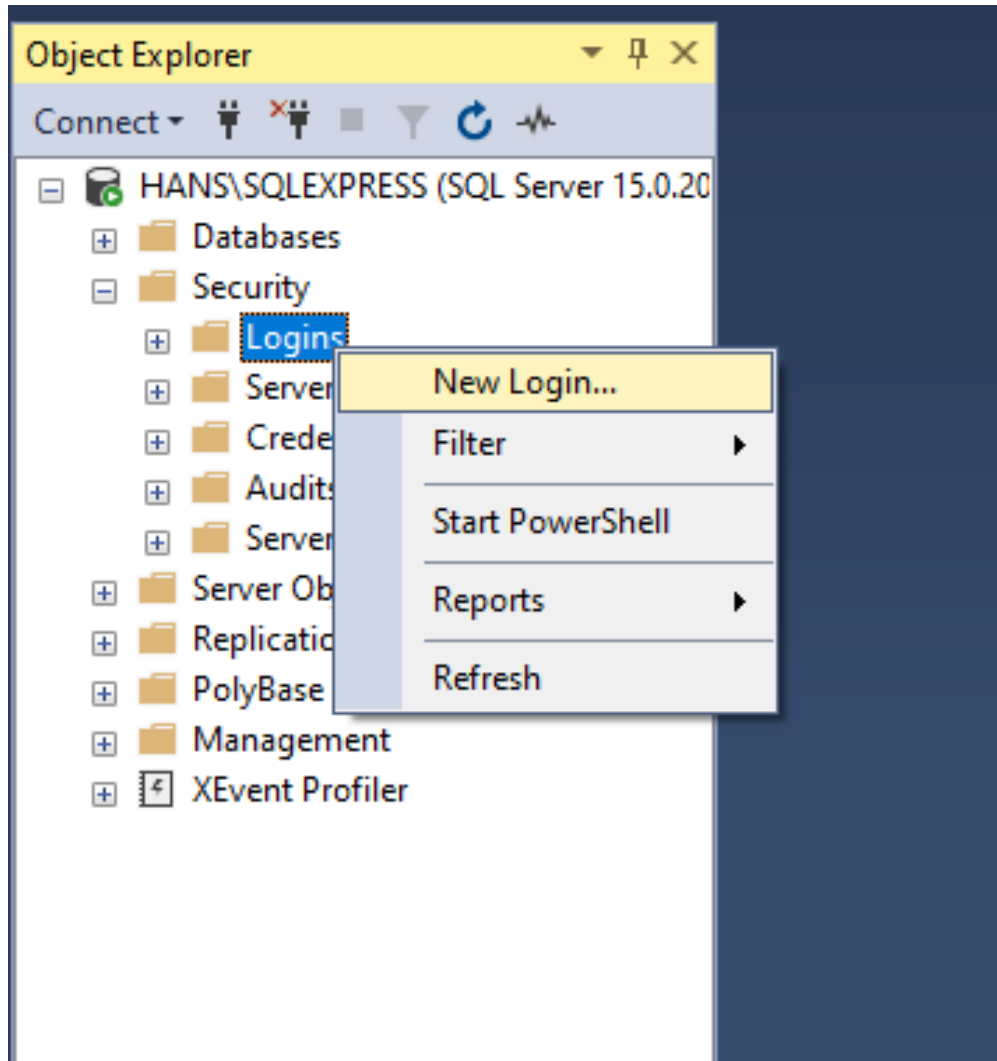
Ready

Step 3. On the Status page, in the Login section, click Enabled, and then click OK.



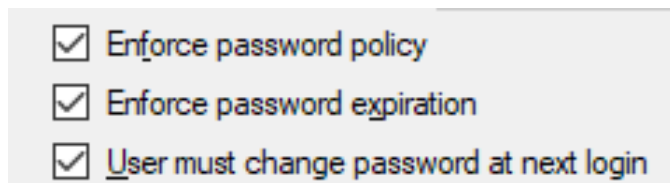
## To create a user login

Step 1. Object Explorer, expand Security, expand Logins, right-click and then choose new Login.



Step 2. On the General page, type the login name and the authentication. Type the password.

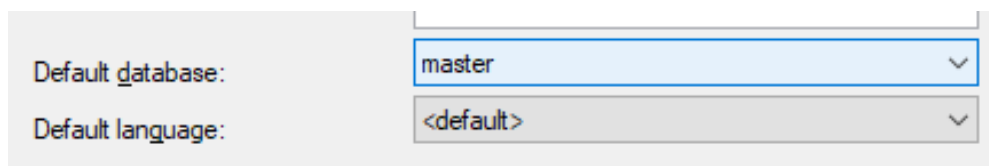
Check the three options under SQL server authentication



A screenshot of a configuration window showing three checked options under SQL server authentication:

- ☒ Enforce password policy
- ☒ Enforce password expiration
- ☒ User must change password at next login

Choose master as default database.



A screenshot of a configuration window showing two settings:

- Default database: master
- Default language: <default>



**Login - New**

**Select a page**

- General
- Server Roles
- User Mapping
- Securables
- Status

**Script** **Help**

Login name:  

☐ Windows authentication  
☒ SQL Server authentication

Password:

Confirm password:

☐ Specify old password

Old password:

☒ Enforce password policy  
☒ Enforce password expiration  
☒ User must change password at next login

☐ Mapped to certificate   
☐ Mapped to asymmetric key   
☐ Map to Credential

Mapped Credentials

Credential	Provider
------------	----------

Default database:

Default language:

**Connection**

Server: HANS\SQLEXPRESS

Connection: HANS\User

[View connection properties](#)

**Progress**

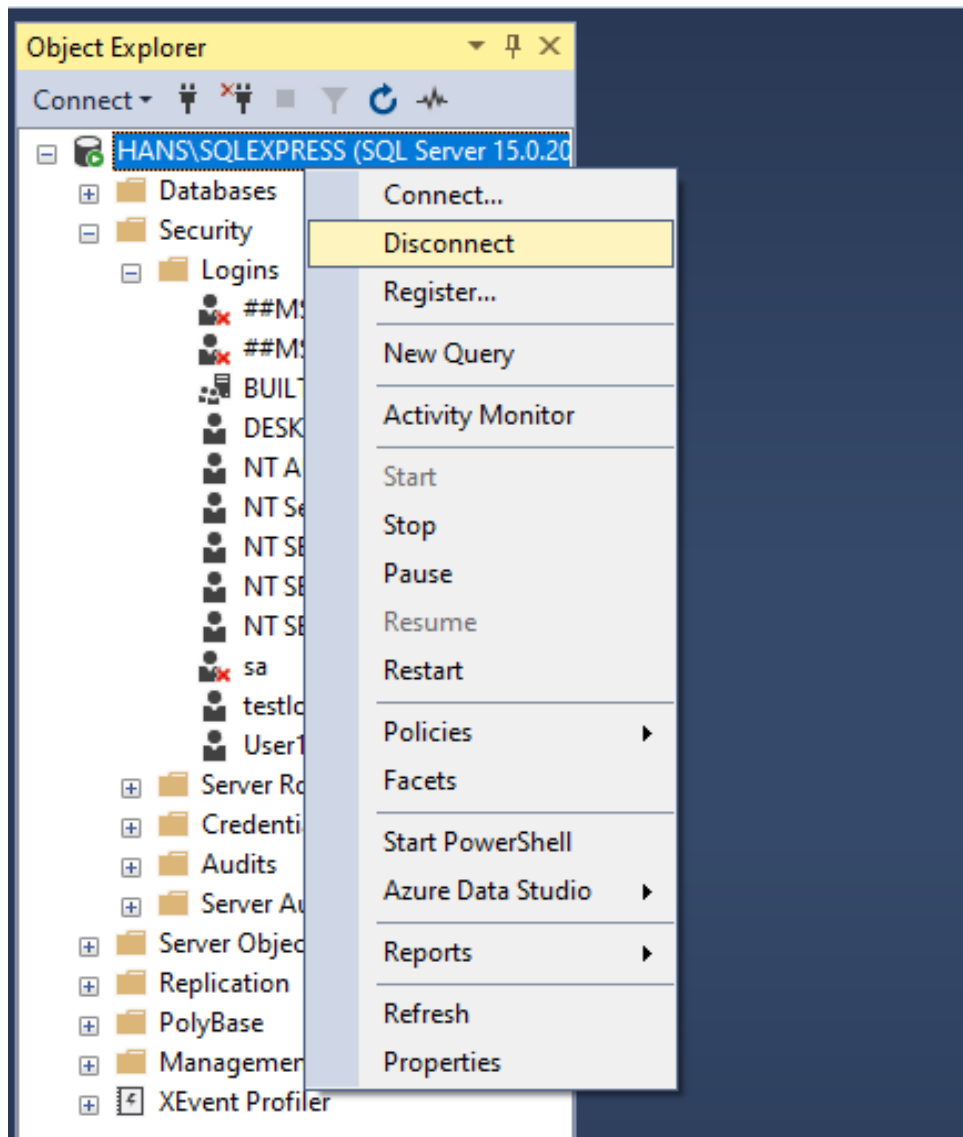
Ready

Have you successfully connected to the database engine? (Yes / No). If No, Explain the procedures to troubleshoot

### Observation:

We successfully connected to the database engine. Since the current login has access with security settings, thus, able to create new logins.

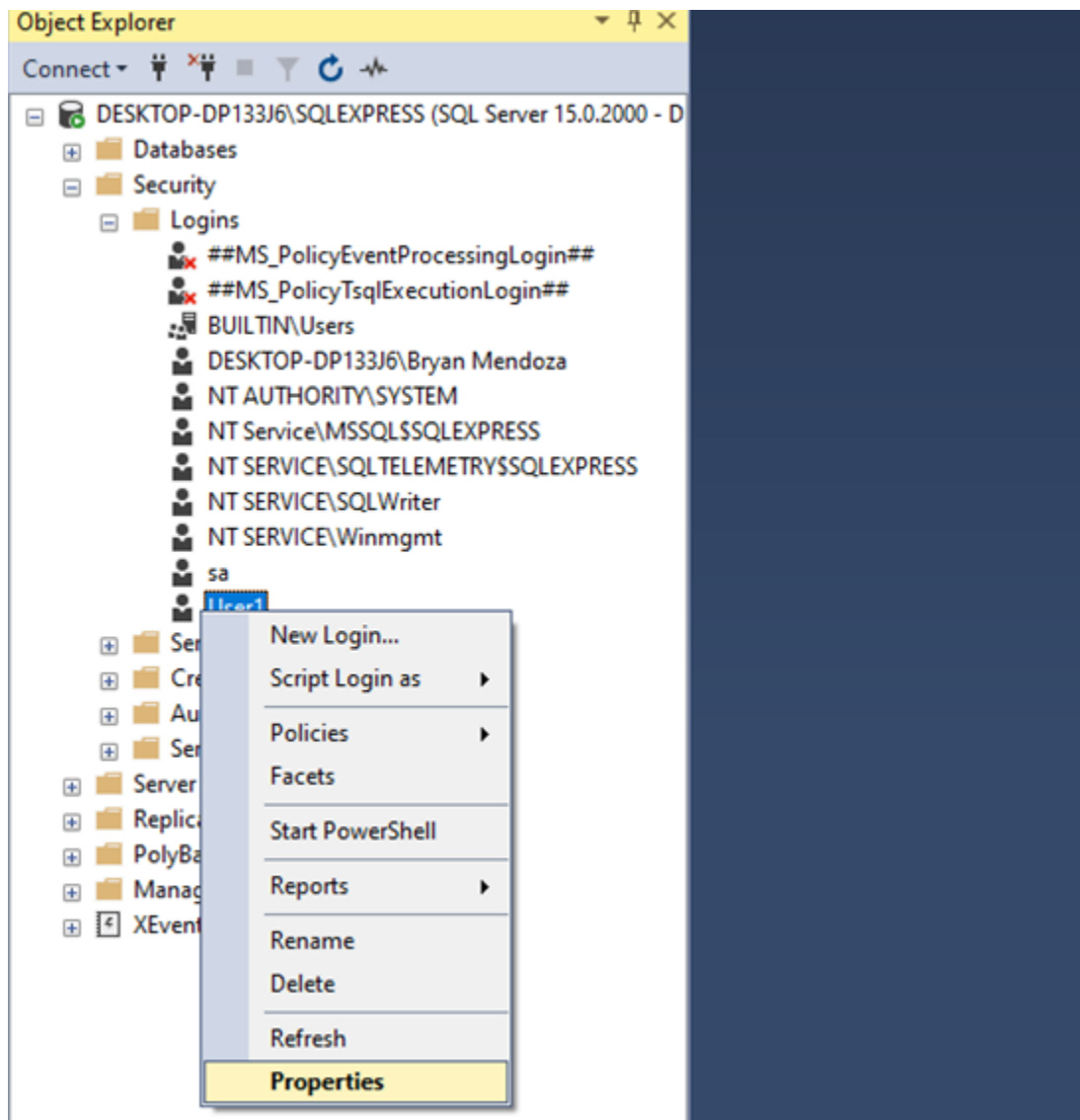
Step 3. Disconnect and reconnect the SQL server database engine to test the User1 user login.



**Observation:** Upon Disconnecting, we tried to login to the server using the created user “User1”. The login was successful.

## To change the server role of a user

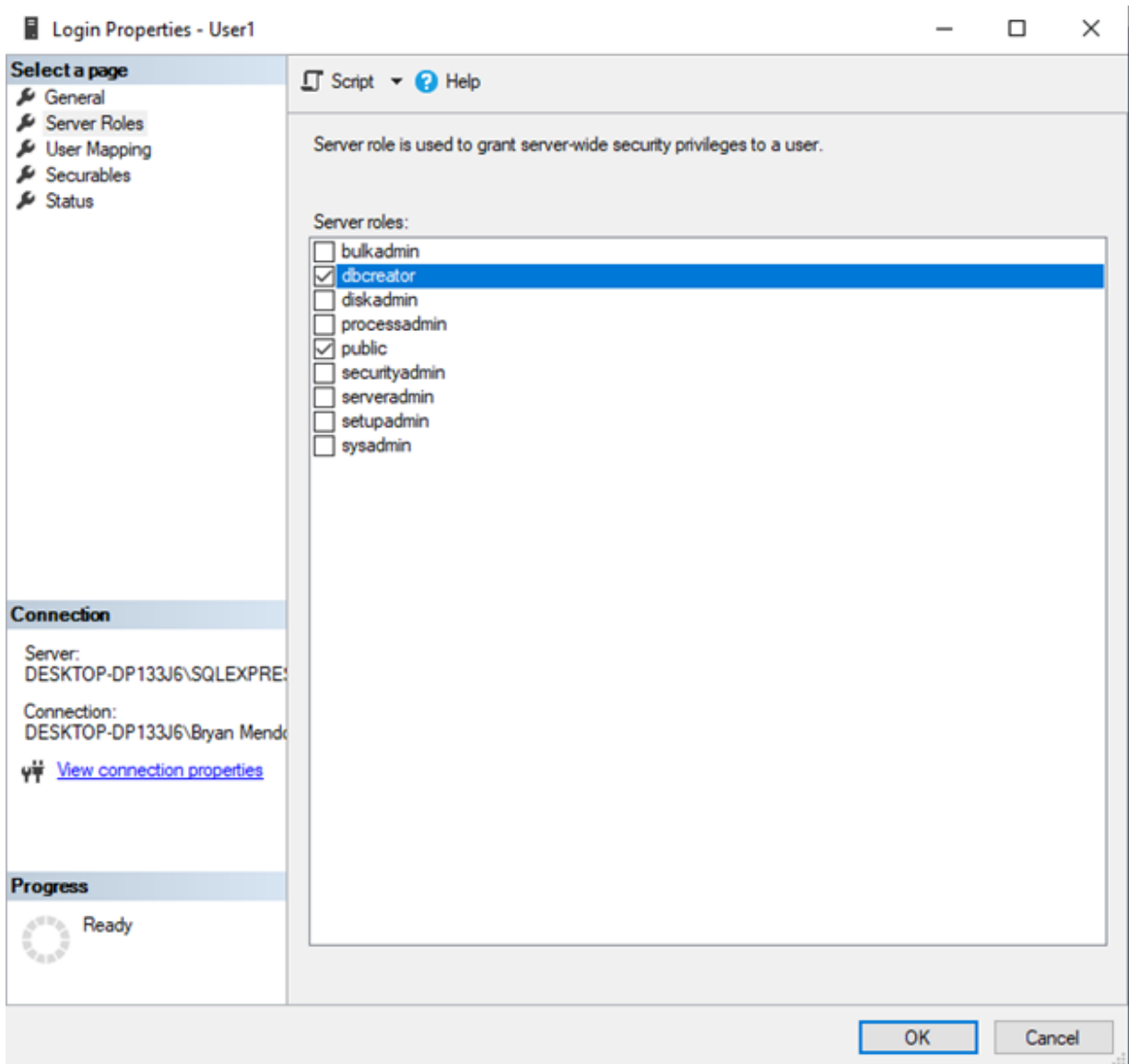
**Step 1.** In Object Explorer, expand Security, expand Logins, right-click a desired user, and then click Properties.



**Observation:** We choose the login or the user we created wherein we will be changing its server roles.

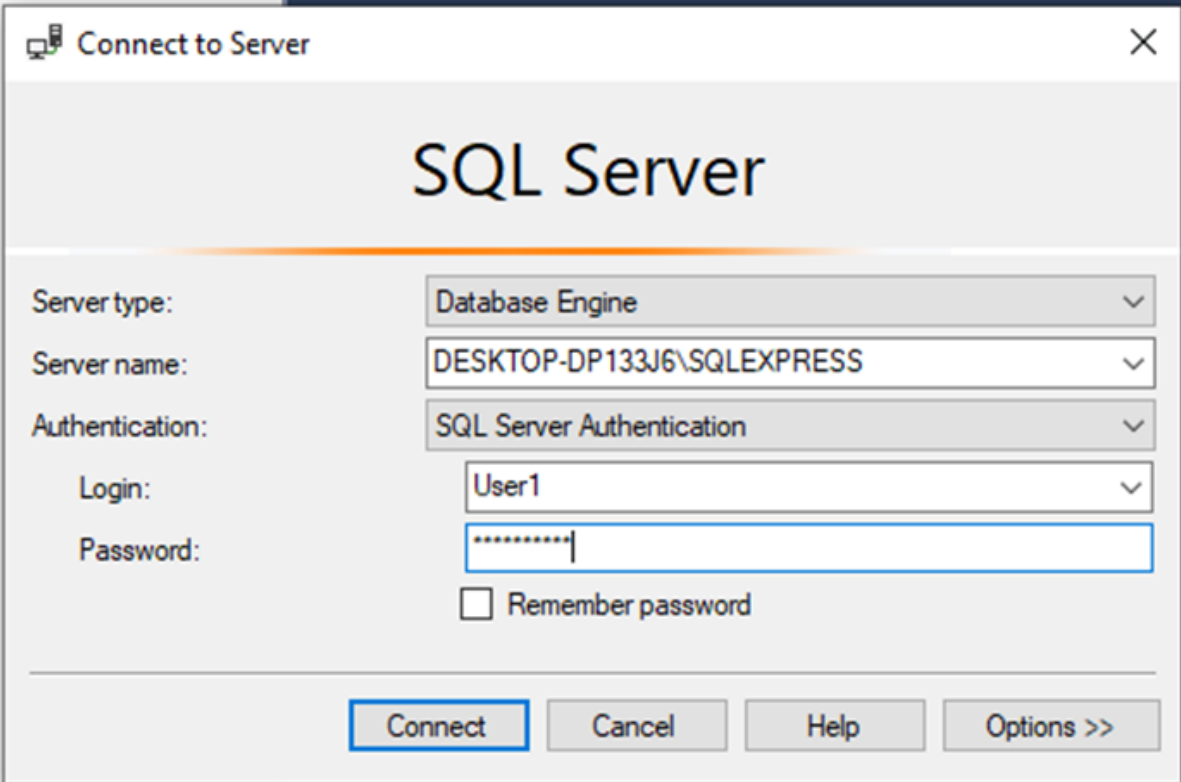
**Step 2.** On the Server Roles page, check the appropriate server role according to the permission allowed to the user and then click OK.

- Use dbcreator for this activity



**Observation:** On the server roles of the selected user or login, we tick the box dbcreator. This server role will enable the user or login to manage databases.

**Step 3.** Disconnect and reconnect to the SQL server. Choose the user account you previously changed as login. Type the password and then click Connect.



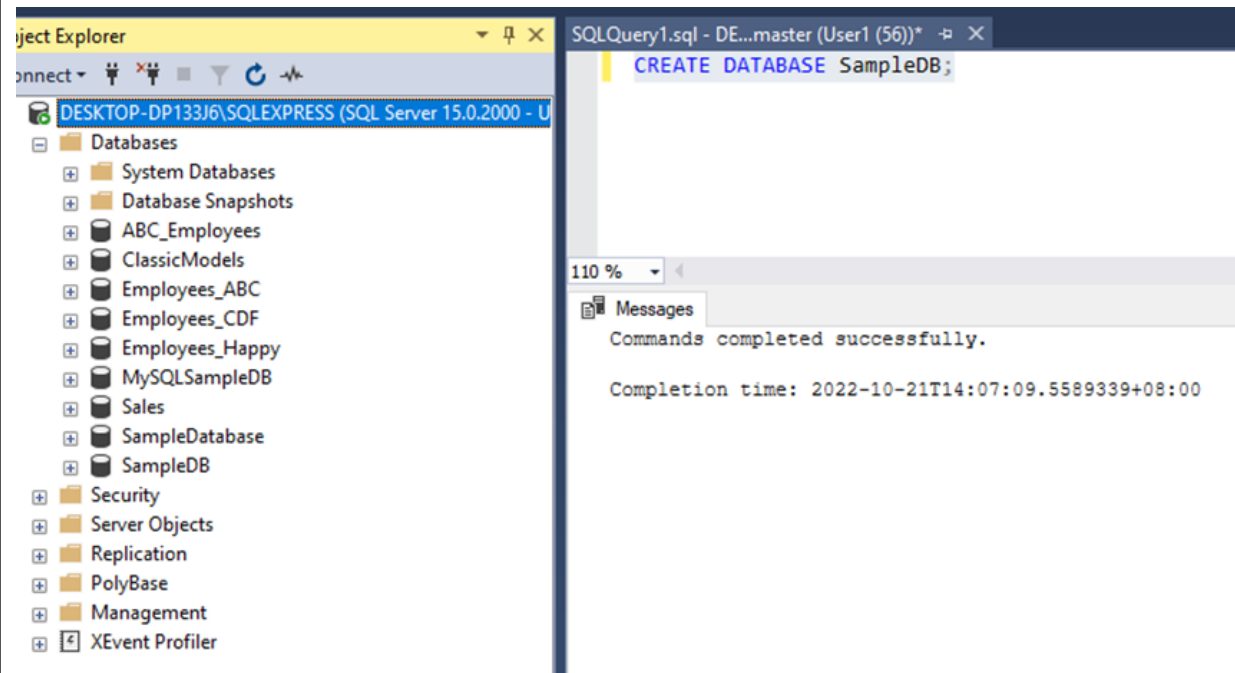
The screenshot shows the 'Connect to Server' dialog box for SQL Server. The title bar reads 'Connect to Server' with a close button (X) on the right. The main heading is 'SQL Server'. Below this, there are several fields and a checkbox:

- Server type:** A dropdown menu showing 'Database Engine'.
- Server name:** A text box containing 'DESKTOP-DP133J6\SQLEXPRESS'.
- Authentication:** A dropdown menu showing 'SQL Server Authentication'.
- Login:** A dropdown menu showing 'User1'.
- Password:** A text box containing a masked password '\*\*\*\*\*'.
- ☐ Remember password

At the bottom, there are four buttons: 'Connect' (highlighted with a blue border), 'Cancel', 'Help', and 'Options >>'.

**Observation:** Disconnecting and Connecting using the newly created User "User1" with the server role dbcreator.

#### Step 4. Create a new database SampleDB

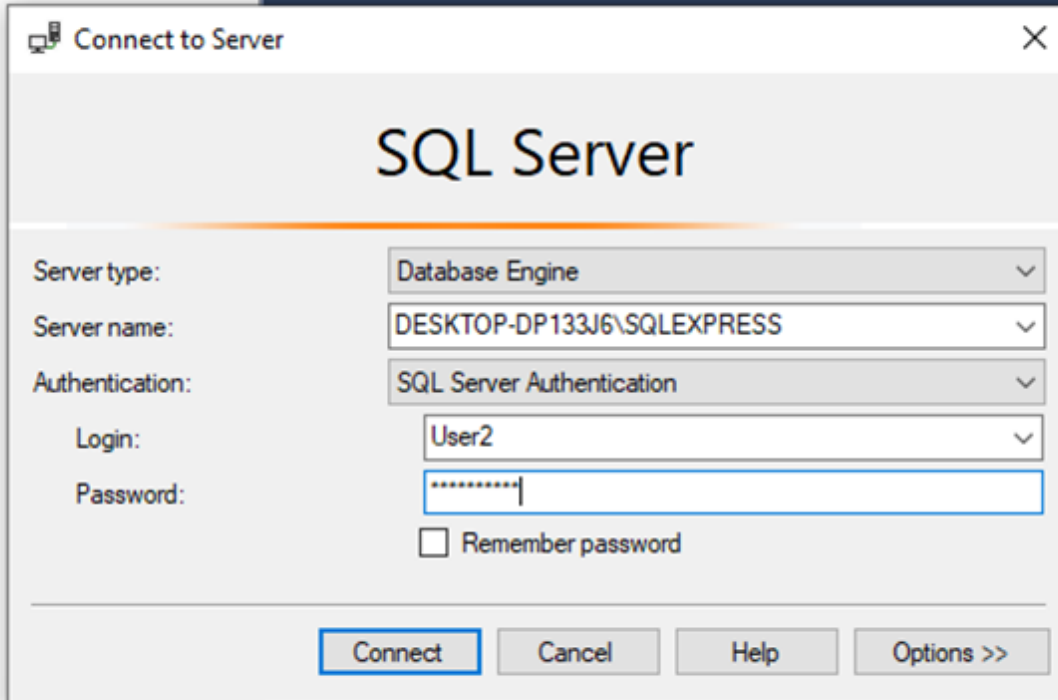


Have you successfully created a database? (Yes / No). **Yes**

If yes, explain why.

**Observation:** We are able to create the database using this login because we assigned a server role dbcreator to this user or login.

**Step 5.** Create a new user login.

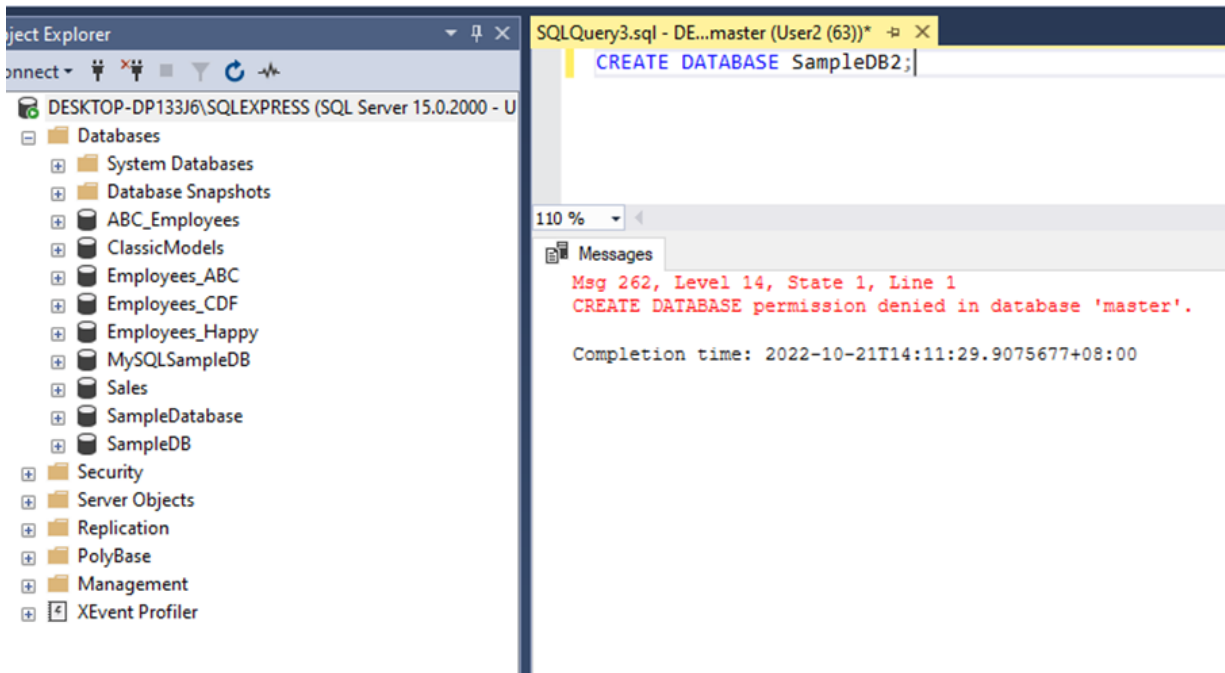


The screenshot shows the 'Connect to Server' dialog box. The title bar reads 'Connect to Server' with a close button. The main heading is 'SQL Server'. The dialog contains the following fields and options:

- Server type: Database Engine (dropdown)
- Server name: DESKTOP-DP133J6\SQLEXPRESS (dropdown)
- Authentication: SQL Server Authentication (dropdown)
- Login: User2 (dropdown)
- Password: [masked with asterisks]
- ☐ Remember password

At the bottom, there are four buttons: 'Connect' (highlighted with a blue border), 'Cancel', 'Help', and 'Options >>'.

**Observation:** Using the main user of the SQL Server, we created another user “User2” which has the same properties as “User1”, except that this user “User2” does not have a server role dbcreator.



Have you successfully created a user login? (Yes / No). **No**

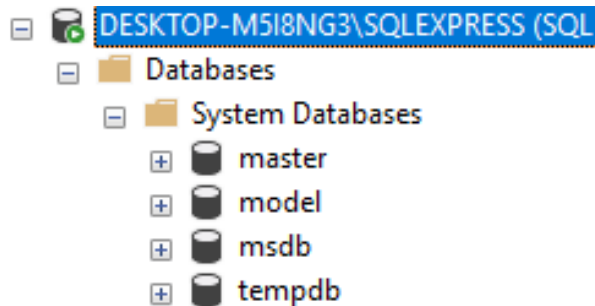
If No, explain why.

**Observation:** We are unable to create a database using this login since we only created this login or user but not assigned with a role. Therefore, this login lacks roles in order to create a database.

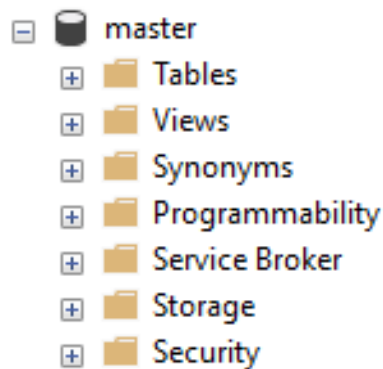


## To create a database user

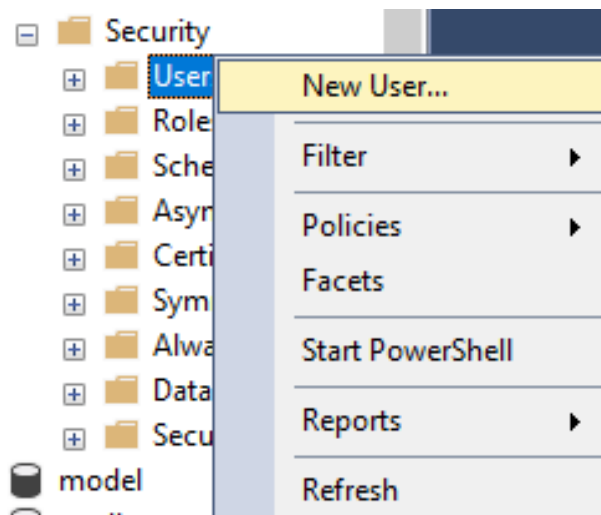
Step 1. In Object Explorer, expand the Databases folder.



Step 2. Expand the database in which to create the new database user.



Step 3. Right-click the Security folder, point to New and Select User

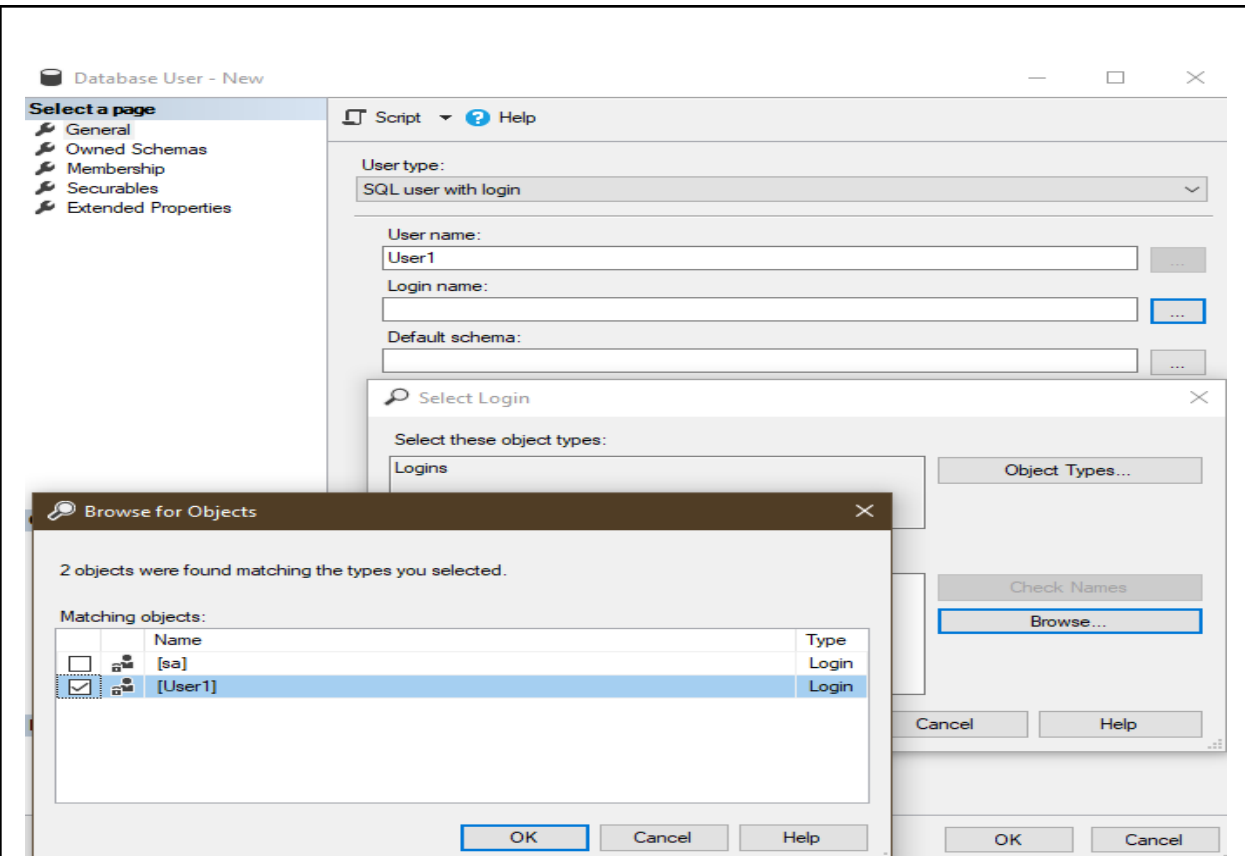


Step 4. On the General page of the Database User- New window, choose one of the user type options from the User type list.

For this activity, use the following options:

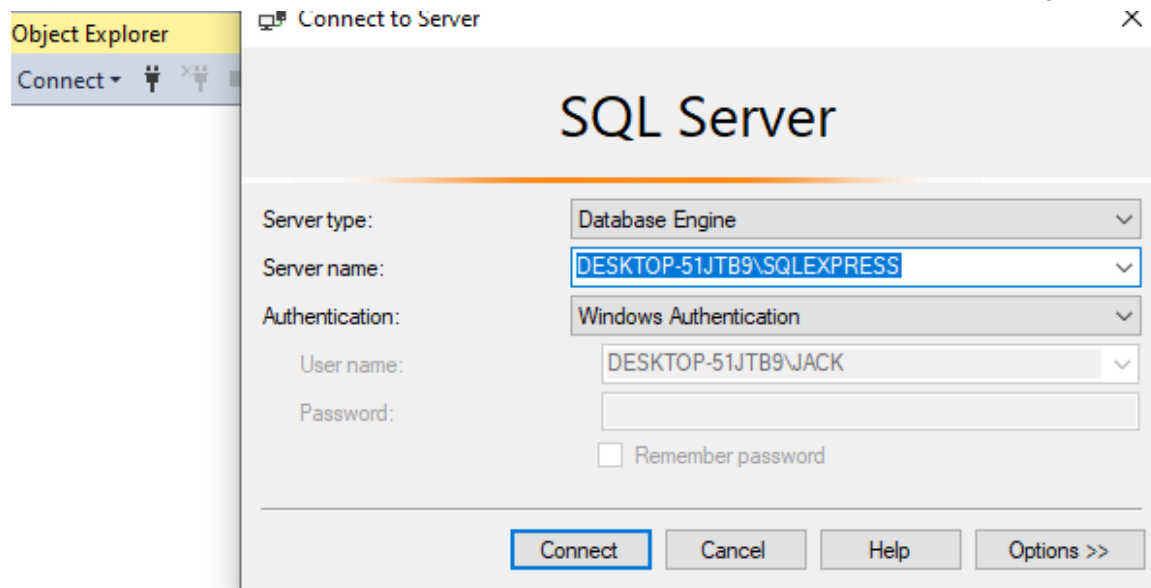
- User type: SQL Server with login
- User name: User1
- Login name: User1

The screenshot shows the 'Database User - New' dialog box. On the left, the 'Select a page' list includes 'General', 'Owned Schemas', 'Membership', 'Securables', and 'Extended Properties'. The 'General' page is active. Below this, the 'Connection' section displays 'Server: DESKTOP-M5I8NG3\SQLEXPRESS' and 'Connection: User1', with a link to 'View connection properties'. At the bottom left, the 'Progress' section shows a 'Ready' status. The main area on the right is titled 'Script' and 'Help'. It contains a 'User type:' dropdown menu set to 'SQL user with login'. Below this are three input fields: 'User name:' (containing 'User1'), 'Login name:' (containing 'User1'), and 'Default schema:' (empty). Each input field has a browse button ('...') to its right. At the bottom right of the dialog are 'OK' and 'Cancel' buttons.



## To create a user using T-SQL

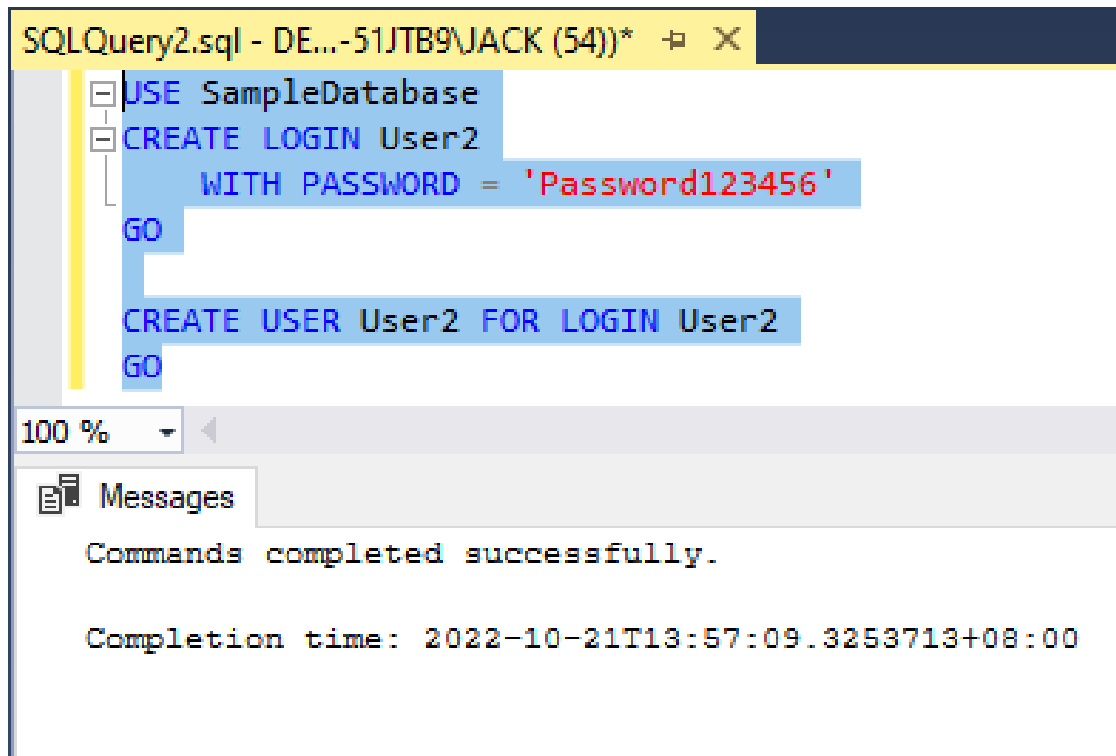
Step 1. In the Object Explorer, connect to an instance of Database Engine.



Step 2. On the Standard bar, click New Query.



Step 3. Copy and paste the following example into the query window and click execute.



The screenshot shows a SQL query window titled "SQLQuery2.sql - DE...-51JTB9\JACK (54))\*". The query text is as follows:

```
USE SampleDatabase
CREATE LOGIN User2
    WITH PASSWORD = 'Password123456'
GO
CREATE USER User2 FOR LOGIN User2
GO
```

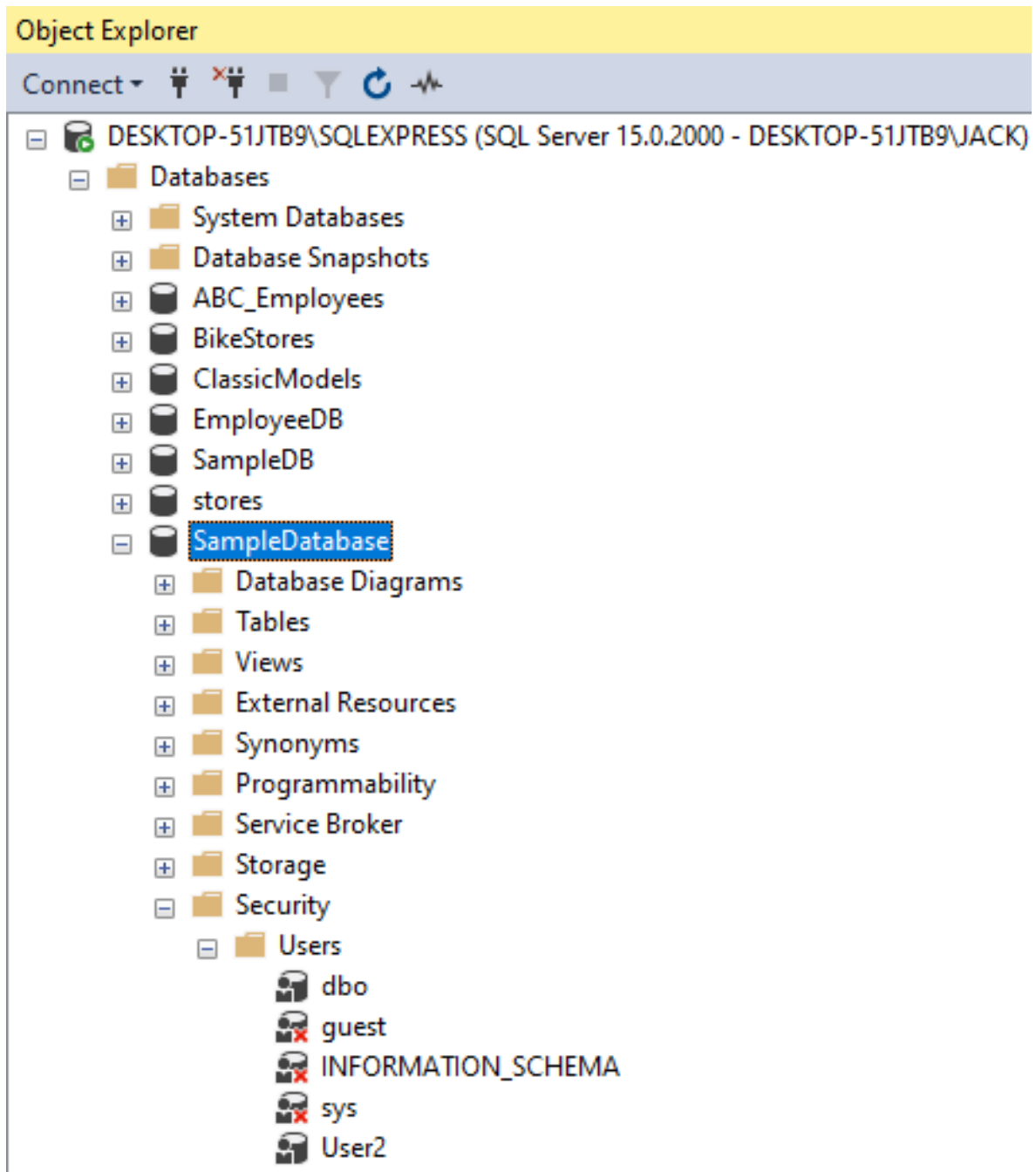
Below the query window, the "Messages" pane shows the following output:

```
Commands completed successfully.

Completion time: 2022-10-21T13:57:09.3253713+08:00
```

**Observations:** Using the syntax given in the procedure, we were able to create a user login via a query window.

Step 4. Verify the database user and login. In Object Explorer, expand SampleDatabase. Choose Security and expand Users. Check if User2 exists.



## Supplementary Activity

1. Create a desired SQL user with login. Make sure that the user login can create another database user and login.
  - Choose a strong password

**Login - New**

Select a page

- General
- Server Roles
- User Mapping
- Securables
- Status

Script Help

Login name: Supplementary8 Search...

☐ Windows authentication

☒ SQL Server authentication

Password: .....

Confirm password: .....

☐ Specify old password

Old password:

☒ Enforce password policy

☒ Enforce password expiration

☒ User must change password at next login

☐ Mapped to certificate

☐ Mapped to asymmetric key

☐ Map to Credential

Mapped Credentials

Credential	Provider
------------	----------

Add

Remove

Default database: master

Default language: <default>

OK Cancel

**Connection**

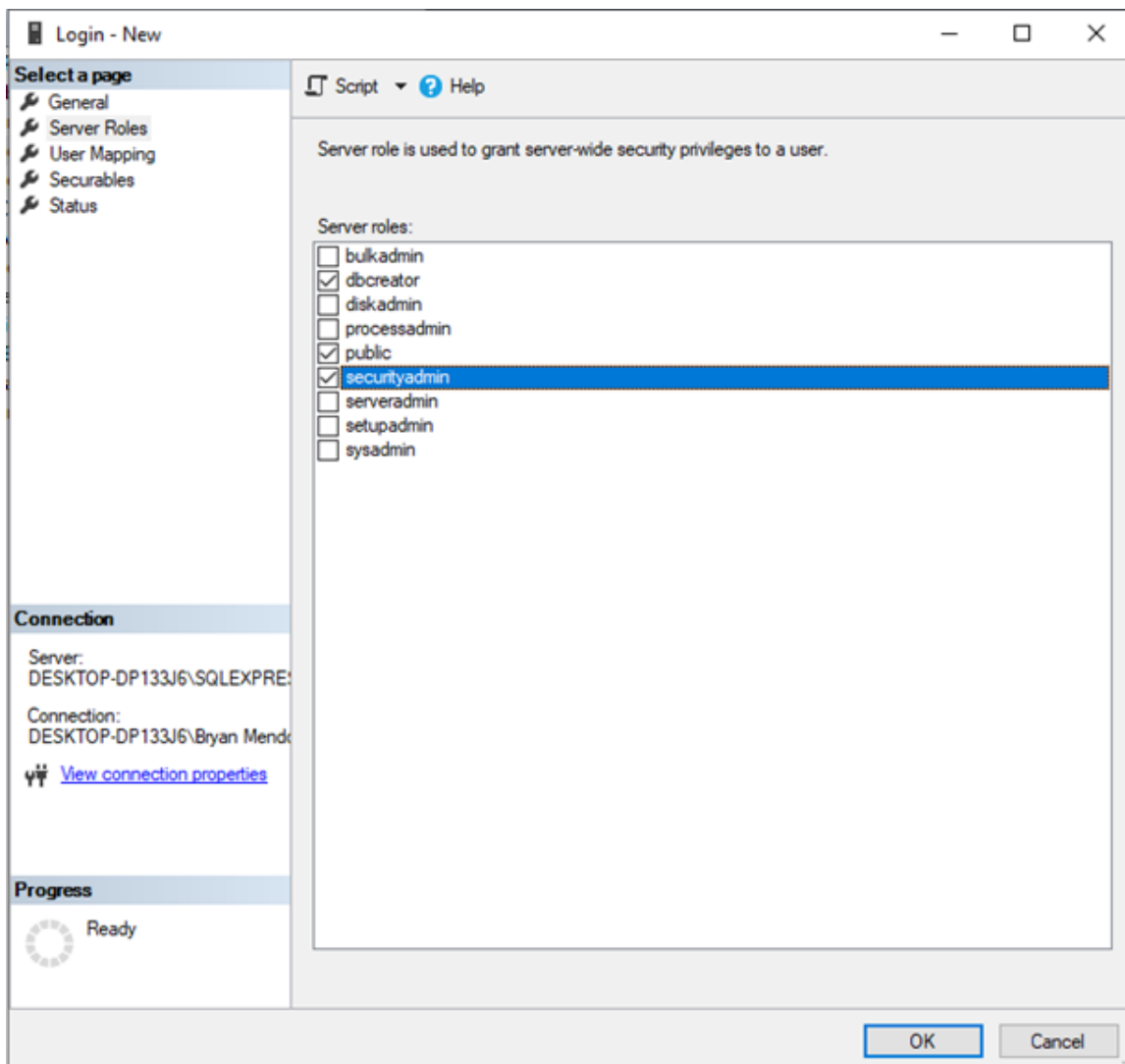
Server: DESKTOP-DP133J6\SQLEXPRESS

Connection: DESKTOP-DP133J6\Bryan Mendez

[View connection properties](#)

**Progress**

Ready



**Details:**

**Login Name used:** Supplementary8

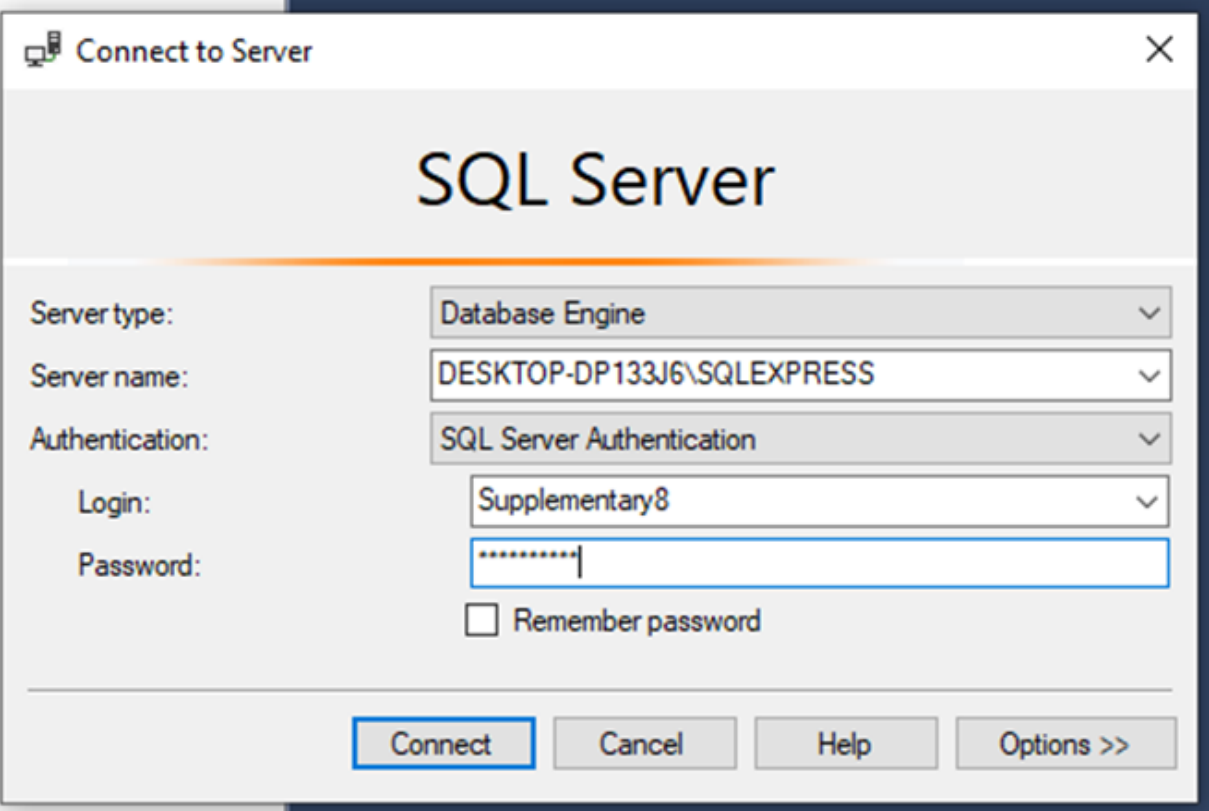
**Password used:** u5erSupp13

**Server Roles:** dbcreator  
Securityadmin

**Observation:** We created a new user login named “Supplementary8” having a password “u5erSupp13”. The login has been given a role of “dbcreator” and “securityadmin” in order for the login or the user to have access with creating, and managing databases as well as making changes with the logins.



2. Connect to the database engine using the created user login.

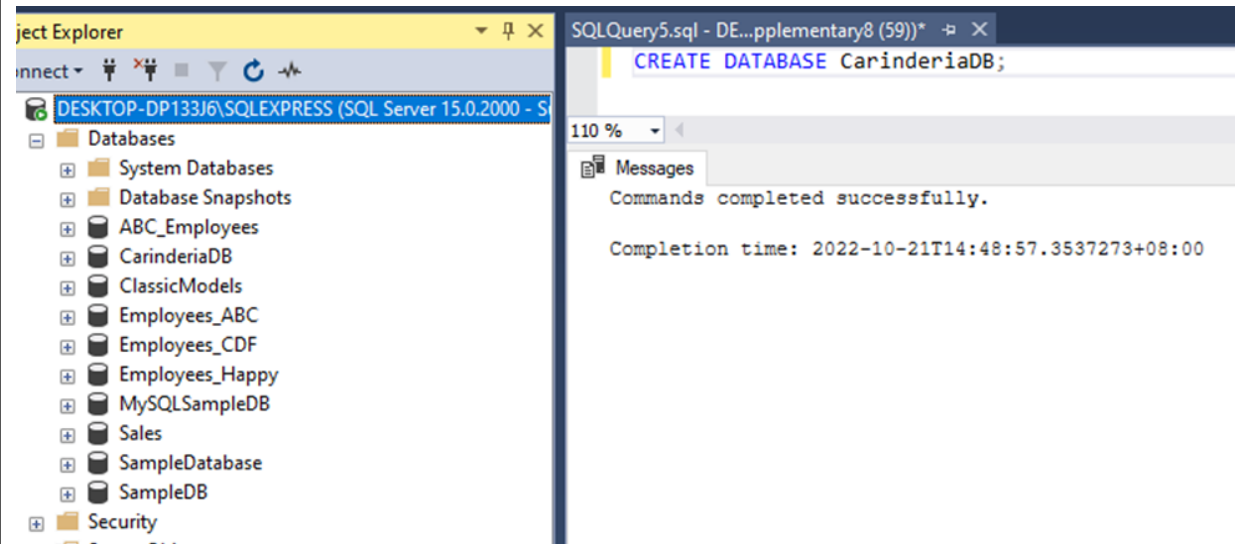


The screenshot shows the 'Connect to Server' dialog box for SQL Server. The title bar reads 'Connect to Server' with a close button. The main heading is 'SQL Server'. The dialog contains the following fields and controls:

- Server type:** A dropdown menu set to 'Database Engine'.
- Server name:** A text box containing 'DESKTOP-DP133J6\SQLEXPRESS'.
- Authentication:** A dropdown menu set to 'SQL Server Authentication'.
- Login:** A text box containing 'Supplementary8'.
- Password:** A text box with masked characters (dots).
- Remember password:** An unchecked checkbox.
- Buttons:** 'Connect' (highlighted with a blue border), 'Cancel', 'Help', and 'Options >>'.

**Observation:** Disconnected with the previous login, and reconnected using the created user “Supplementary8”.

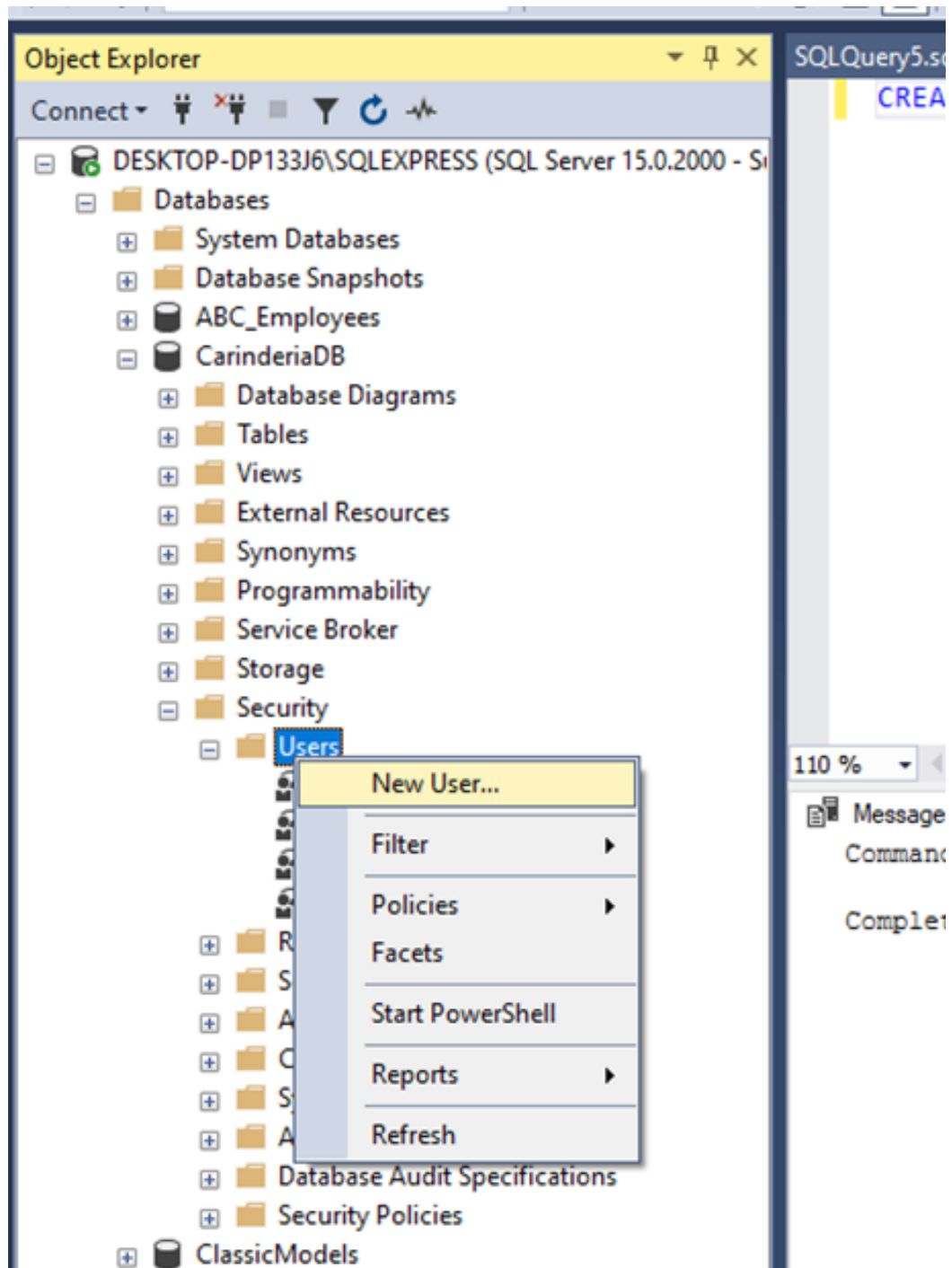
3. Create a database CarindieriaDB.



**Observation:** Using a new query under the new login, we created a new database “CarindieriaDB”. This creation of the database proves that the new login “Supplementary8” has access to modifying, managing, creating and deleting Databases.

After refreshing the Server (without disconnecting), the created database appeared on the object explorer tab of the Microsoft SQL Server.

4. Create a user peter that can only create, modify and delete databases.



Database User - New

Select a page

- General
- Owned Schemas
- Membership
- Securables
- Extended Properties

Connection

Server:  
DESKTOP-DP133J6\SQLEXPRESS

Connection:  
Supplementary8

 [View connection properties](#)

Progress

 Ready

Script   Help

User type:

SQL user with login

User name:

Peter

Login name:

Peter


Default schema:

OK

Cancel

[-]  DESKTOP-DP133J6\SQLEXPRESS (SQL Server 15.0.2000 - Si


[-]  Databases

[+]  System Databases

[+]  Database Snapshots


[+]  ABC\_Employees

[-]  CarindieriaDB

[+]  Database Diagrams

[+]  Tables

[+]  Views

[+]  External Resources

[+]  Synonyms

[+]  Programmability

[+]  Service Broker

[+]  Storage


[-]  Security

[-]  Users

 dbo

 guest

 INFORMATION\_SCHEMA

 sys

 Peter

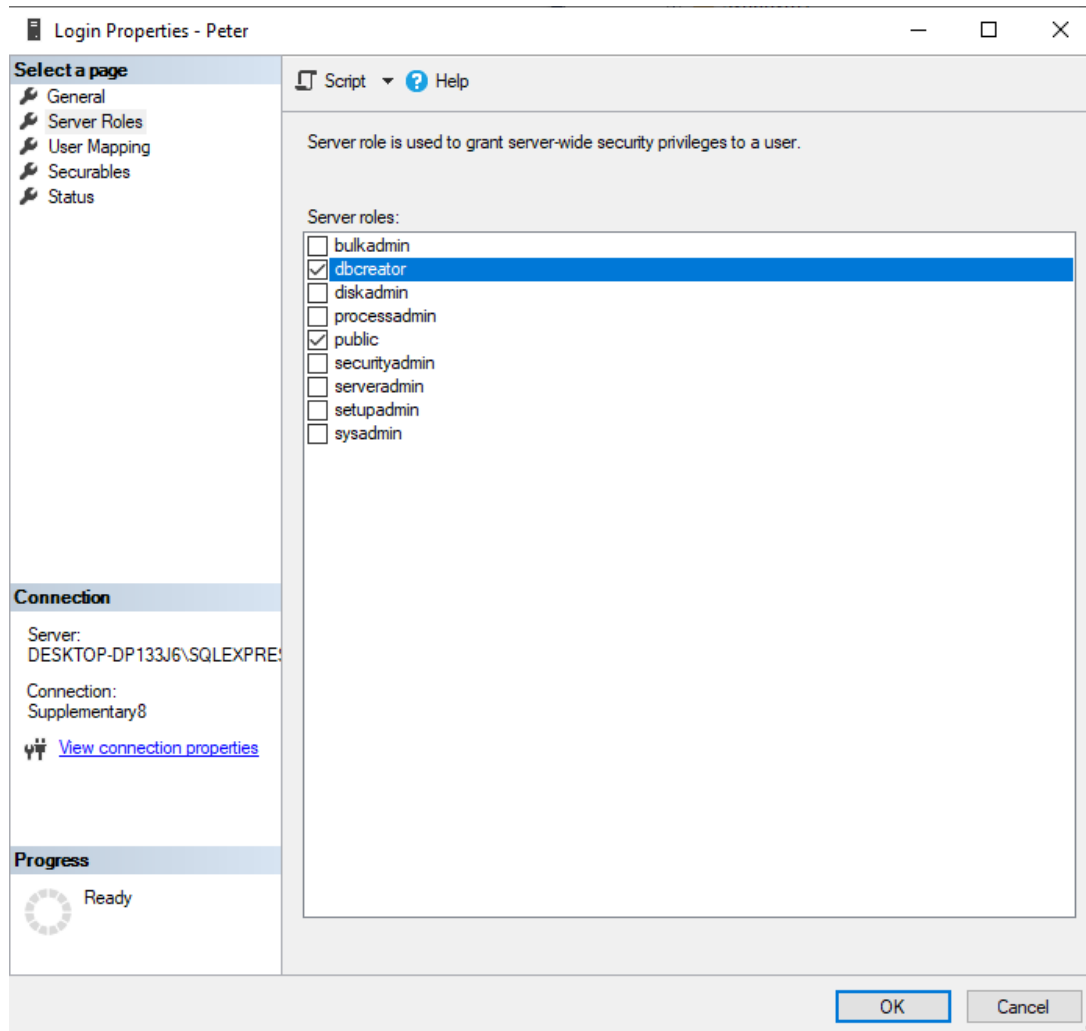
[+]  Roles

[+]  Schemas

[+]  Asymmetric Keys

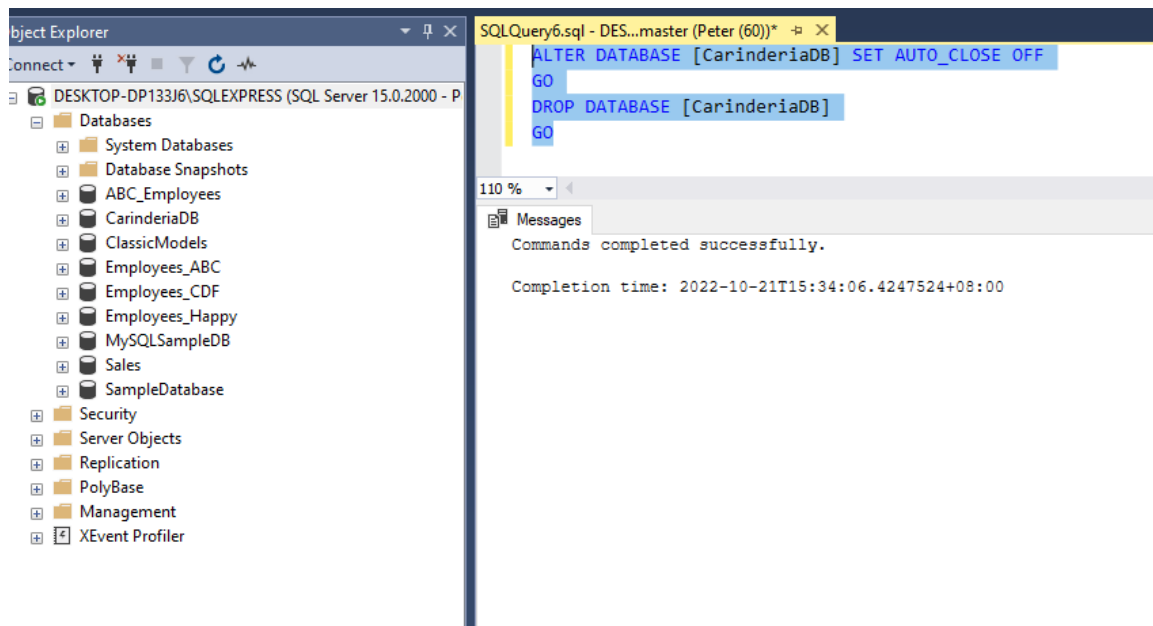
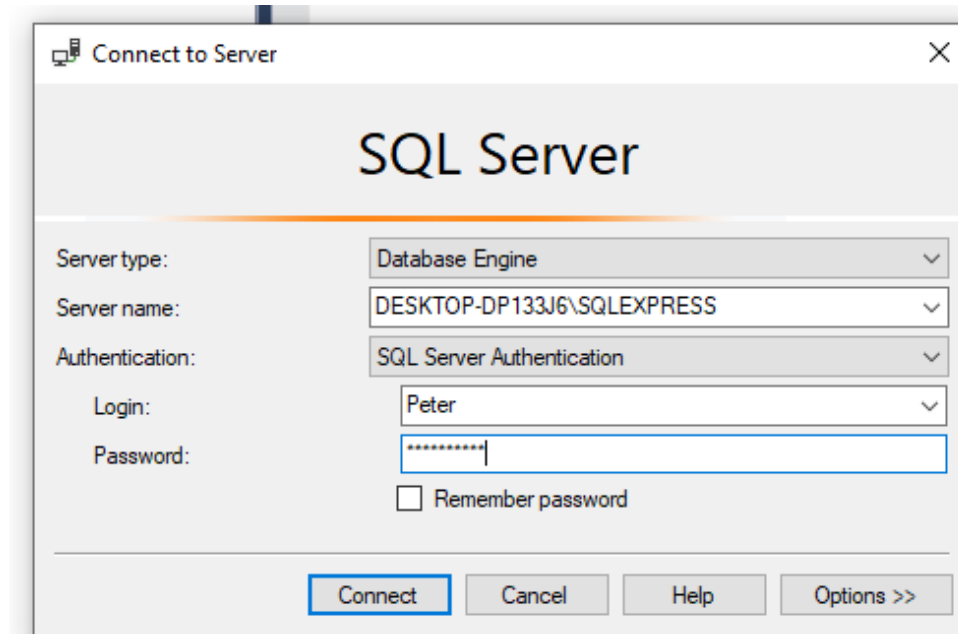
[+]  Certificates

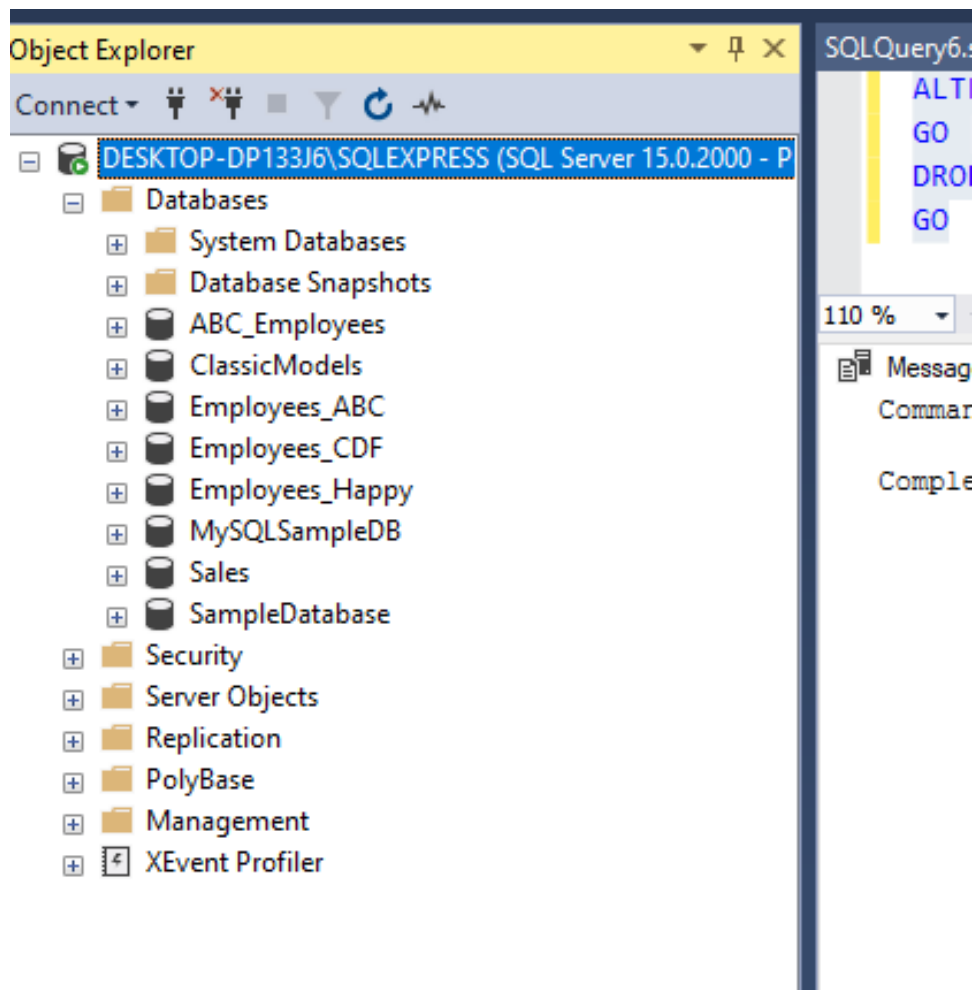
[+]  Symmetric Keys



**Observation:** Under the created database “CarindieriaDB”, we added a new user “Peter” which has server roles dbcreator which enables the user create, modify and delete databases.

5. Connect using user peter and delete CarindieriaDB.



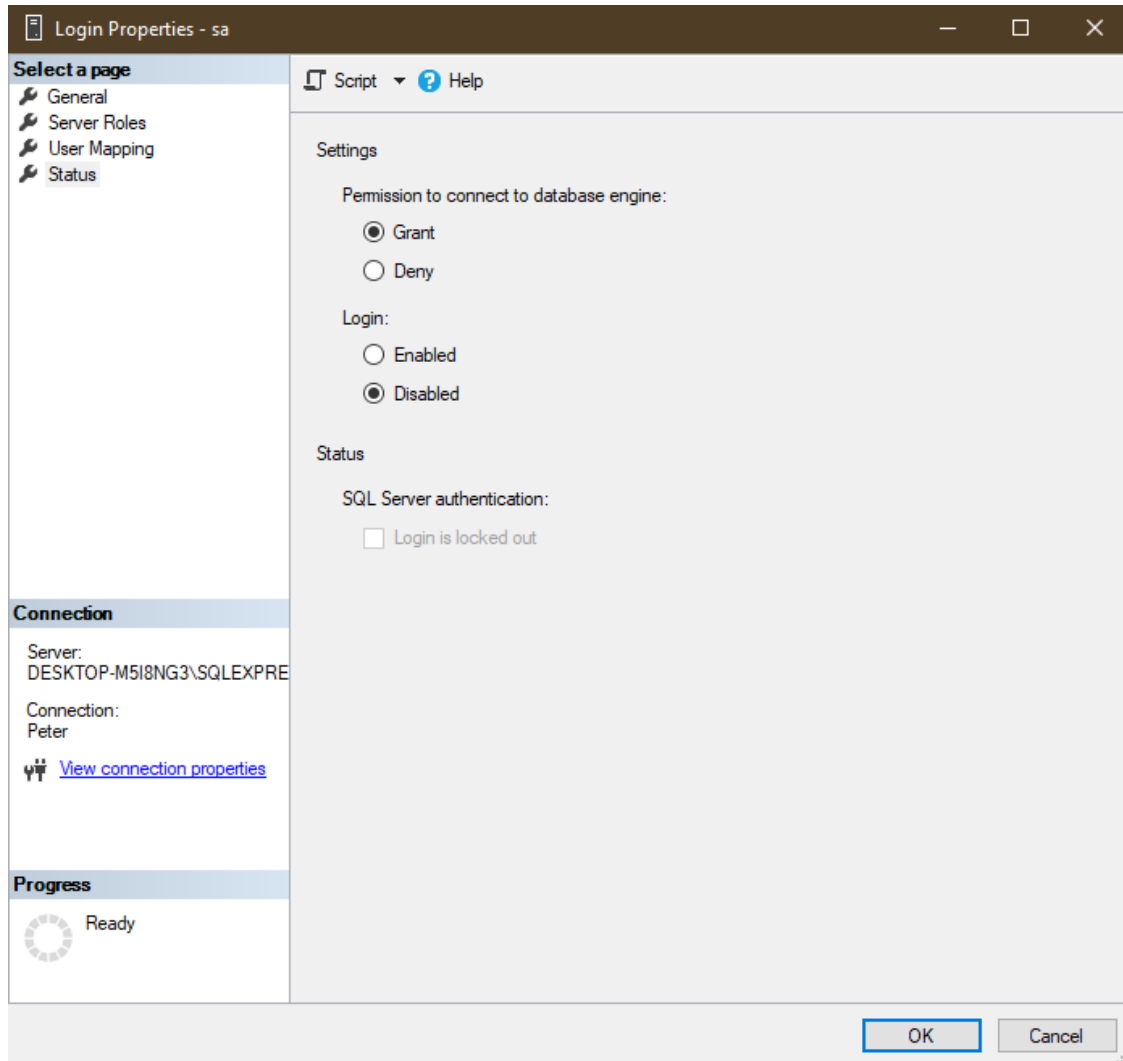


**Observation:** Using the user “Peter” we are able to drop the database since the user “Peter” has a server role dbcreator which enables the user to create, modify, and delete databases. Since this user is only on CarindariaDB, he is only able to create, modify, and delete the same database.

We used a query in order to perform deletion of the database. The images provided above shows the before and after images which shows what happened before and after the query.

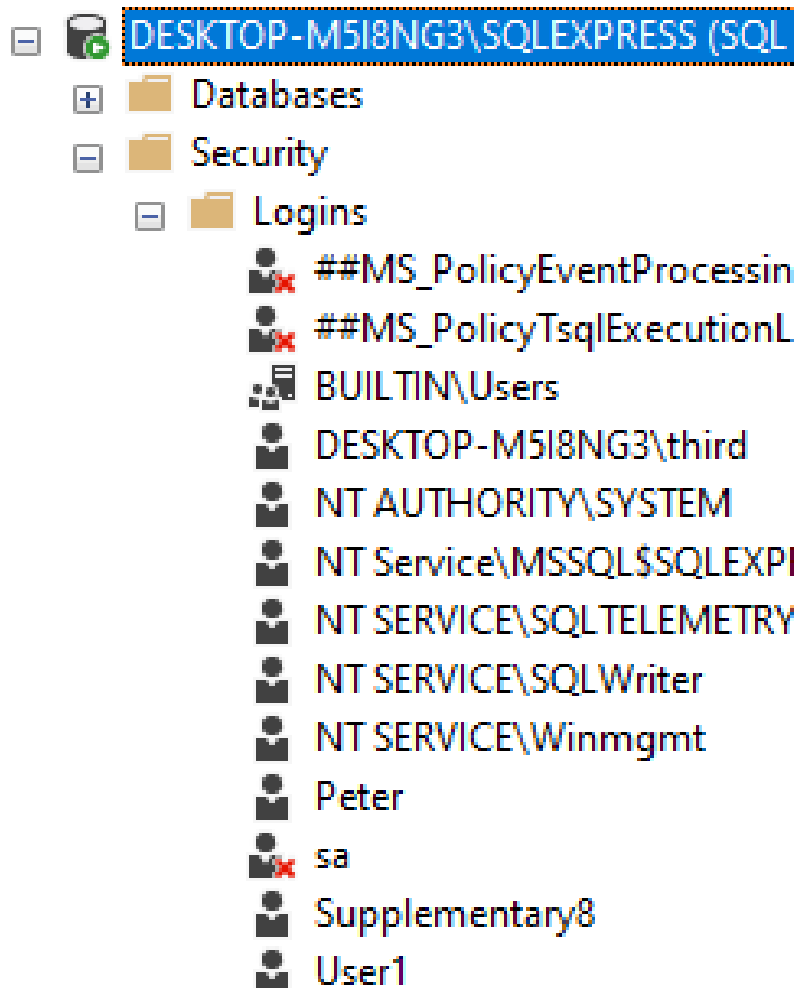


6. Disable the user sa login.



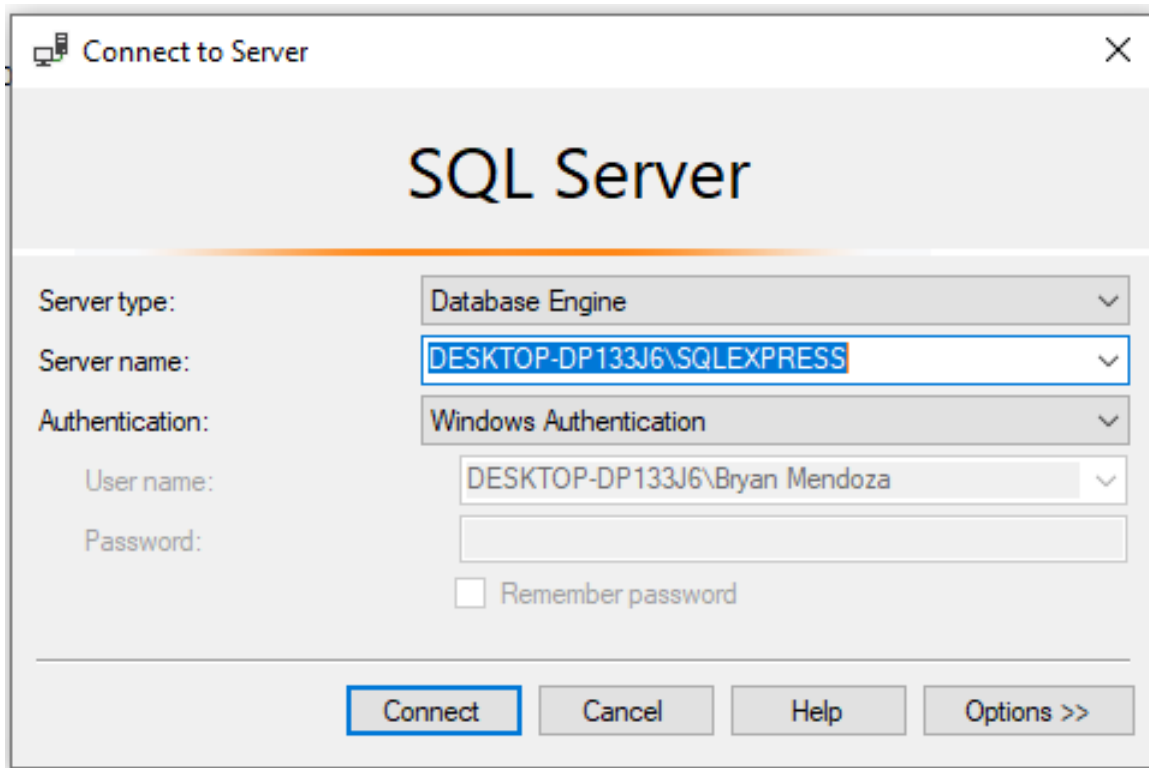
**Observation:** In order to disable the sa login, we should use a login that has access with the security settings.

7. Change the authentication to Windows only.



**Observation:** Using the Windows Authentication, we are able to see all of the logins available in the server since this authentication has the main role.

8. Connect to the database engine using the Windows authentication.

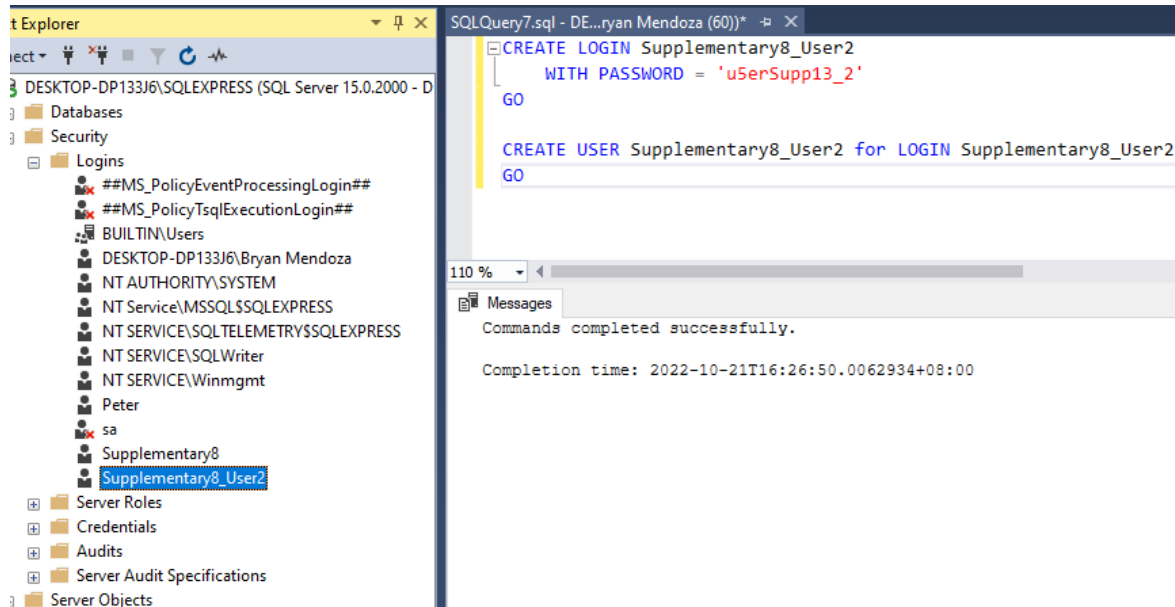


The screenshot shows the 'Connect to Server' dialog box for SQL Server. The title bar reads 'Connect to Server'. The main heading is 'SQL Server'. The dialog contains the following fields and options:

- Server type: Database Engine (dropdown menu)
- Server name: DESKTOP-DP133J6\SQLEXPRESS (dropdown menu)
- Authentication: Windows Authentication (dropdown menu)
- User name: DESKTOP-DP133J6\Bryan Mendoza (text box)
- Password: (empty text box)
- ☐ Remember password (checkbox)
- Buttons: Connect, Cancel, Help, Options >>

**Observation:** Logging to the SQL Server using Windows Authentication.

9. Create a desired SQL user with login using T-SQL.



**Observation:** Using a new query, we are able to add a new login and user to our server. This created user does not exist on a specific database, this user exists in the server since we have not included a specific database for it to be included in our query.

## Conclusion

In this laboratory activity, the students were introduced with the security settings of databases using Microsoft SQL Studio. The security settings differ on the level of its capabilities. There are security settings which is server-level security wherein the user can access the data in the server as well as create log-ins, implement server settings and other users. In contrast with database-level security, it only enables the user to access the data within the database only. We observed database-level security with the supplementary activity wherein the user can only access the database in which he is assigned and not the other databases outside of its scope.

Moreover, there are also roles which dictate the capabilities of the user or the login. There are fixed-roles which are already predefined by the Microsoft SQL Server wherein it grants the user an access to specific functions. There are also user-defined roles wherein we, the user of the server defines what capabilities the created user should have. The user-defined role has been observed throughout this laboratory, wherein we added a dbcreator server role to our created user, and this role enabled the user to create, modify, and delete databases.

Administering Databases Securing Databases restricting access to certain users, controlling what each user can do and running anti-virus software. We conclude that databases are the foundation of all applications. A database may be attacked because it includes sensitive and significant information. This document discusses several database hacks.

## Proof of Collaboration

The screenshot shows a Google Docs interface within a virtual meeting room. The top bar indicates 'Breakout room time remaining: 85:18' and 'CPE 011-CPE21S3 - Database Management System Conference (Room 11)'. The document title is 'Group11\_HandsonActivity 8.1'. The left sidebar lists users: NICOLAS, SEAN JULIAN (now), EFA, CHRISTIAN ED, GUEVARRA, HANS ANGELO, VINLUAN, ARMANDO, and MENDOZA, JOHN RENZO. The main document area contains a table with the following data:

Activity No. 8.1 - Administering Databases Securing Databases	
<b>Name:</b> Efa, Christian Guevarra, Hans Angelo Mendoza, John Renzo Nicolas, Sean Julian Vinluan, Armando	<b>Date:</b> 21/12/22
<b>Section:</b> CPE21S3	<b>Instructor:</b> Dr. Jonathan Vidal Taylor

Below the table, the text reads: 'Database Output' and 'To change security authentication mode'. A numbered list follows: '1. In SQL Server Management Studio Object Explorer, right-click the server, and then click Properties.' An inset image shows the 'Object Explorer' window in SQL Server Management Studio with the 'Security' folder expanded. At the bottom, a status bar indicates 'ayde138759.rox2.blindsidenetworks.com is sharing a window.' with 'Stop sharing' and 'Hide' buttons.

## Honor Pledge

“I accept responsibility for my role in ensuring the integrity of the work submitted by the group in which I participated.”