DATA2201 - Relational Databases
Project Phase 2 – Supporting GUI

Presented by:
Dario Ospina, Jody Boston, Carl Ocsan, Michael Boddy

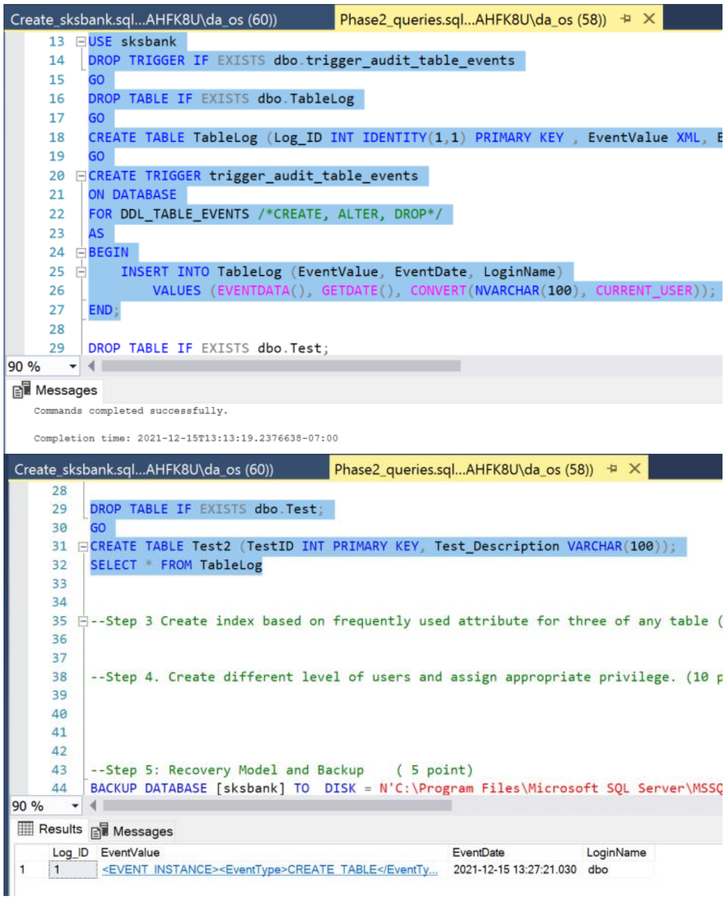
Step 1. Implementing different functionalities using stored procedure and user defined function. (15 points)

Refer to the SQL file

Step 2. Create different set of triggers (minimum 2 numbers) to monitor the different DML and DDL activates in the database (10 point)

Trigger example → Dario

SQL Server


Create_sksbank.sql...AHFK8U\da_os (60))
Phase2_queries.sql...AHFK8U\da_os (58))
90 %
Messages

```
USE sksbank
DROP TRIGGER IF EXISTS dbo.trigger_audit_table_events
GO
DROP TABLE IF EXISTS dbo.TableLog
GO
CREATE TABLE TableLog (Log_ID INT IDENTITY(1,1) PRIMARY KEY , EventValue XML, E
GO
CREATE TRIGGER trigger_audit_table_events
ON DATABASE
FOR DDL_TABLE_EVENTS /*CREATE, ALTER, DROP*/
AS
BEGIN
    INSERT INTO TableLog (EventValue, EventDate, LoginName)
        VALUES (EVENTDATA(), GETDATE(), CONVERT(NVARCHAR(100), CURRENT_USER));
END;

DROP TABLE IF EXISTS dbo.Test;
```

```
Commands completed successfully.

Completion time: 2021-12-15T13:13:19.2376638-07:00
```

Create_sksbank.sql...AHFK8U\da_os (60))
Phase2_queries.sql...AHFK8U\da_os (58))
90 %
Results
Messages

```

DROP TABLE IF EXISTS dbo.Test;
GO
CREATE TABLE Test2 (TestID INT PRIMARY KEY, Test_Description VARCHAR(100));
SELECT * FROM TableLog


--Step 3 Create index based on frequently used attribute for three of any table (


--Step 4. Create different level of users and assign appropriate privilege. (10 p




--Step 5: Recovery Model and Backup    ( 5 point)
BACKUP DATABASE [sksbank] TO  DISK = N'C:\Program Files\Microsoft SQL Server\MSSQ
```

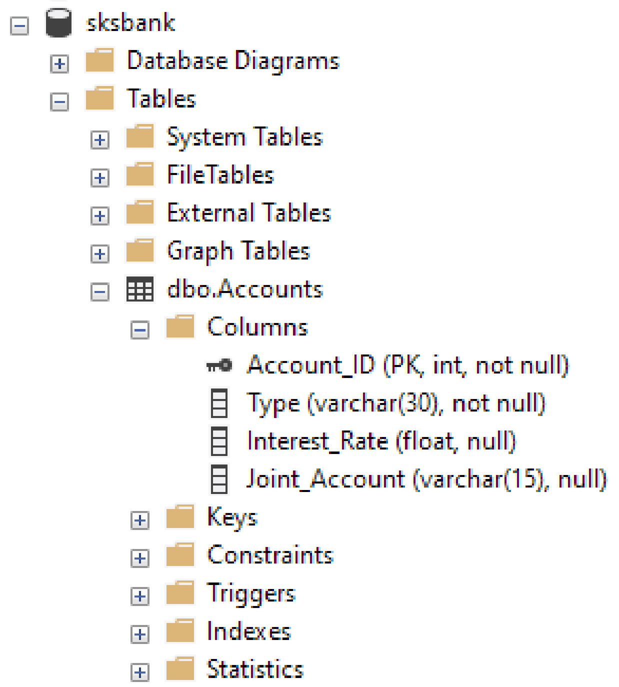
	Log_ID	EventValue	EventDate	LoginName
1	1	<EVENT_INSTANCE><EventType>CREATE_TABLE</EventTy...	2021-12-15 13:27:21.030	dbo

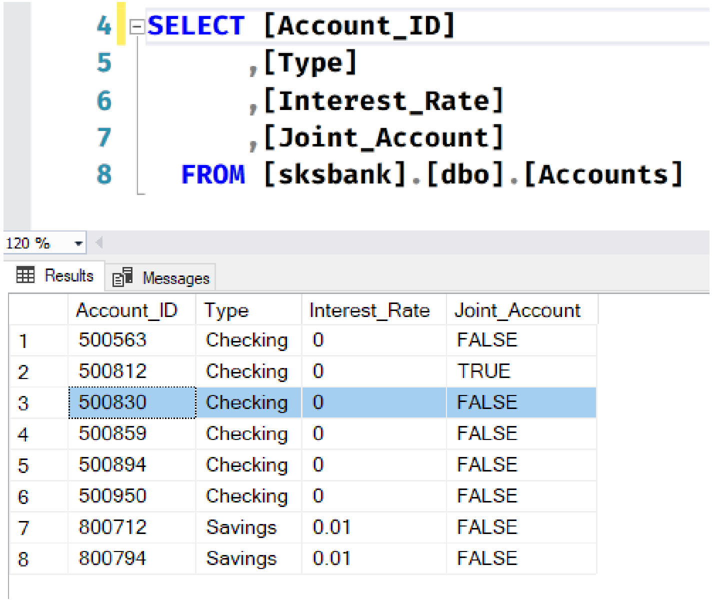
T-SQL

```
/*Creating Trigger and TableLog*/

USE sksbank
DROP TRIGGER IF EXISTS
dbo.trigger_audit_table_events
GO
DROP TABLE IF EXISTS dbo.TableLog
GO
CREATE TABLE TableLog (Log_ID INT
IDENTITY(1,1) PRIMARY KEY , EventValue XML,
EventDate DATETIME NOT NULL, LoginName
NVARCHAR(100))
GO
CREATE TRIGGER trigger_audit_table_events
ON DATABASE
FOR DDL_TABLE_EVENTS /*CREATE, ALTER, DROP*/
AS
BEGIN
       INSERT INTO TableLog (EventValue,
EventDate, LoginName)
              VALUES (EVENTDATA(), GETDATE(),
CONVERT(NVARCHAR(100), CURRENT_USER));
END;

/*Testing*/

DROP TABLE IF EXISTS dbo.Test;
GO
CREATE TABLE Test2 (TestID INT PRIMARY KEY,
Test_Description VARCHAR(100));
SELECT * FROM TableLog
```

Trigger example → Jody

SQL Server



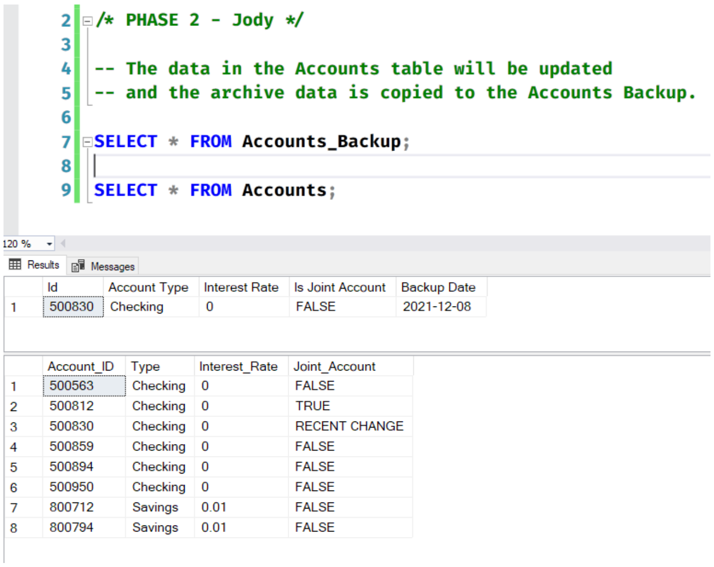


```
SELECT [Account_ID]
      ,[Type]
      ,[Interest_Rate]
      ,[Joint_Account]
  FROM [sksbank].[dbo].[Accounts]
```

	Account_ID	Type	Interest_Rate	Joint_Account
1	500563	Checking	0	FALSE
2	500812	Checking	0	TRUE
3	500830	Checking	0	FALSE
4	500859	Checking	0	FALSE
5	500894	Checking	0	FALSE
6	500950	Checking	0	FALSE
7	800712	Savings	0.01	FALSE
8	800794	Savings	0.01	FALSE

T-SQL

```
USE sksbank
GO
DROP TABLE Accounts_Backup -- drop backup
table.

USE sksbank
GO
-- creating new table to copy archive data
into.
CREATE TABLE Accounts_Backup
(
       [Id] INT NOT NULL,
       [Account Type] VARCHAR (30) NOT NULL,
       [Interest Rate] FLOAT NULL,
       [Is Joint Account] VARCHAR (15) NULL,
       [Backup Date] DATE NOT NULL,
       -- see we created a new col to store
date processed.
)

-- Create trigger to store updates.
USE sksbank
GO
CREATE TRIGGER dbo.TG_AccountsInsert
   ON Accounts AFTER UPDATE
AS
BEGIN
       INSERT INTO Accounts_Backup
       (Id, [Account Type], [Interest
Rate],[Is Joint Account],[Backup Date])
       SELECT Account_ID, [Type],
Interest_Rate, Joint_Account, GETDATE()
       FROM DELETED
END
GO

-- Now to test the trigger by updating the
Accounts table
UPDATE Accounts SET Joint_Account = 'RECENT
CHANGE' WHERE Account_ID = 500830;

-- The data in the Accounts table will be
updated
-- and the archive data is copied to the
Accounts Backup.

SELECT * FROM Accounts_Backup;
SELECT * FROM Accounts WHERE Account_ID =
500830;
```



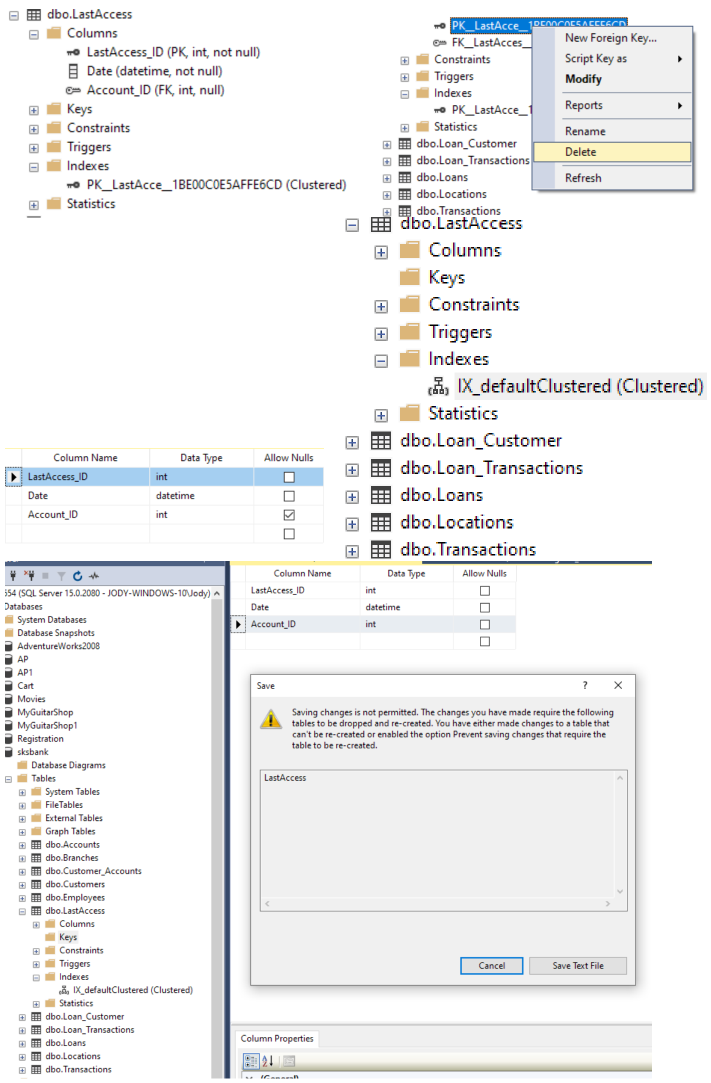
```
/* PHASE 2 - Jody */

-- The data in the Accounts table will be updated
-- and the archive data is copied to the Accounts Backup.

SELECT * FROM Accounts_Backup;

SELECT * FROM Accounts;
```

```
     Id      Account Type  Interest Rate  Is Joint Account  Backup Date
1    500830  Checking      0              FALSE             2021-12-08

     Account_ID  Type      Interest_Rate  Joint_Account
1    500563      Checking  0              FALSE
2    500812      Checking  0              TRUE
3    500830      Checking  0              RECENT CHANGE
4    500859      Checking  0              FALSE
5    500894      Checking  0              FALSE
6    500950      Checking  0              FALSE
7    800712      Savings   0.01           FALSE
8    800794      Savings   0.01           FALSE
```

Step 3. Create index based on frequently used attribute for three of any table (10 point)

- Jody → Replace the default cluster index with non-key attribute for one table. If you used GUI please provide details on which table you implemented it

SQL Server	T-SQL



Column Name	Data Type	Allow Nulls
LastAccess_ID	int	□
Date	datetime	□
Account_ID	int	☑
		□

Column Name	Data Type	Allow Nulls
LastAccess_ID	int	□
Date	datetime	□
Account_ID	int	□
		□

Save

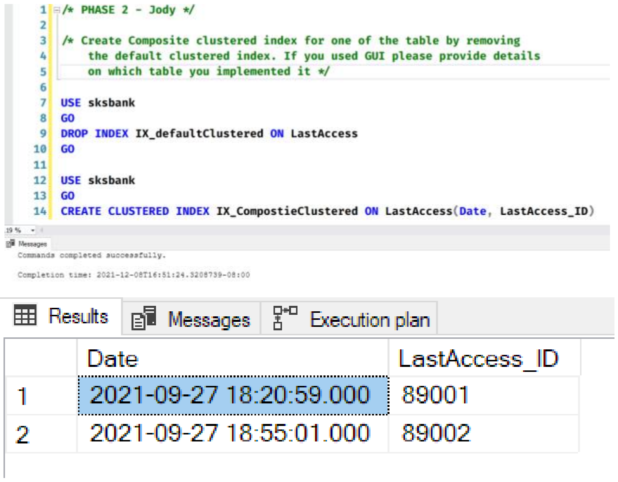
Saving changes is not permitted. The changes you have made require the following tables to be dropped and re-created. You have either made changes to a table that can't be re-created or enabled the option Prevent saving changes that require the table to be re-created.

LastAccess

```
USE sksbank
GO
CREATE CLUSTERED INDEX IX_defaultClustered ON
LastAccess(Account_ID)

-- NOTES: Due to the way our DB is
constructed, we only have 2 tables that
'maybe' could use this on
-- However, after attempting this, it is not
possible without dropping the table and
recreating it.
```

- **Jody →** Create Composite clustered index for one of the table by removing the default clustered index. . If you used GUI please provide details on which table you implemented it

SQL Server	T-SQL

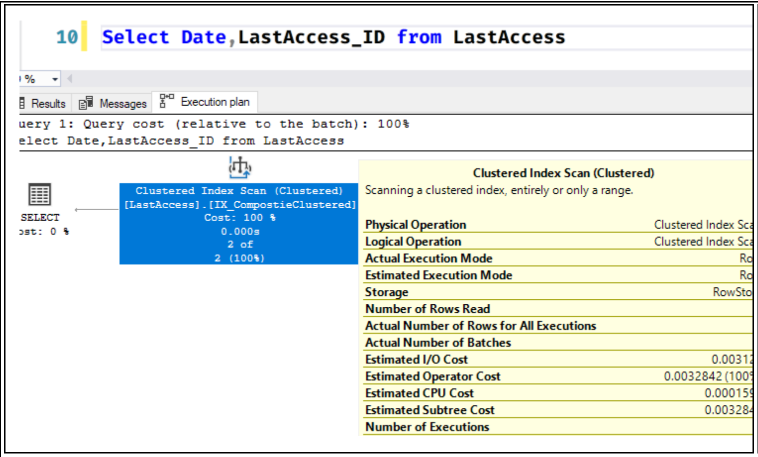


```
/* PHASE 2 - Jody */

/* Create Composite clustered index for one of the table by removing
   the default clustered index. If you used GUI please provide details
   on which table you implemented it */

USE sksbank
GO
DROP INDEX IX_defaultClustered ON LastAccess
GO

USE sksbank
GO
CREATE CLUSTERED INDEX IX_CompostieClustered ON LastAccess(Date, LastAccess_ID)
```

```
Commands completed successfully.

Completion time: 2021-12-08T16:51:24.3208739-08:00
```

	Date	LastAccess_ID
1	2021-09-27 18:20:59.000	89001
2	2021-09-27 18:55:01.000	89002

```
USE sksbank
GO
DROP INDEX IX_defaultClustered ON LastAccess
GO

USE sksbank
GO
CREATE CLUSTERED INDEX IX_CompositeClustered
ON LastAccess(Date, LastAccess_ID)

    -- SELECT Date,LastAccess_ID FROM
    LastAccess
```



```
Select Date,LastAccess_ID from LastAccess
```

```
uery 1: Query cost (relative to the batch): 100%
elect Date,LastAccess_ID from LastAccess
```

SELECT
ost: 0 %
Clustered Index Scan (Clustered)
[LastAccess].[IX_CompostieClustered]
Cost: 100 %
0.000s
2 of
2 (100%)

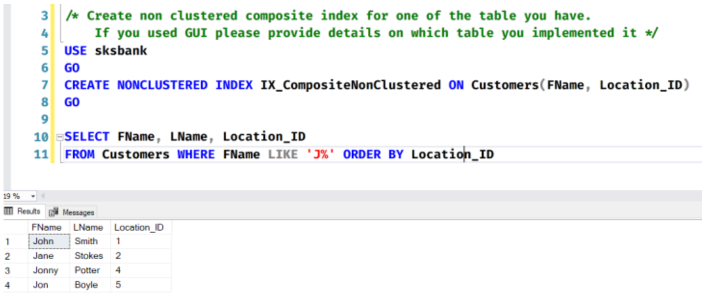
Clustered Index Scan (Clustered)

Scanning a clustered index, entirely or only a range.

Physical Operation	Clustered Index Sca
Logical Operation	Clustered Index Sca
Actual Execution Mode	Ro
Estimated Execution Mode	Ro
Storage	RowSto
Number of Rows Read	
Actual Number of Rows for All Executions	
Actual Number of Batches	
Estimated I/O Cost	0.00312
Estimated Operator Cost	0.0032842 (100
Estimated CPU Cost	0.00015
Estimated Subtree Cost	0.00328
Number of Executions	

- **Jody →** Create non clustered composite index for one of the table you have. If you used GUI please provide details on which table you implemented it

SQL Server



```
/* Create non clustered composite index for one of the table you have.
   If you used GUI please provide details on which table you implemented it */
USE sksbank
GO
CREATE NONCLUSTERED INDEX IX_CompositeNonClustered ON Customers(FName, Location_ID)
GO

SELECT FName, LName, Location_ID
FROM Customers WHERE FName LIKE 'J%' ORDER BY Location_ID
```

	FName	LName	Location_ID
1	John	Smith	1
2	Jane	Stokes	2
3	Jonny	Potter	4
4	Jon	Boyle	5

T-SQL

```
USE sksbank
GO
CREATE NONCLUSTERED INDEX
IX_CompositeNonClustered ON Customers(FName,
Location_ID)
GO

SELECT FName, LName, Location_ID
FROM Customers WHERE FName LIKE 'J%'
```

Step 4. Create different level of users and assign appropriate privilege. (10 point)

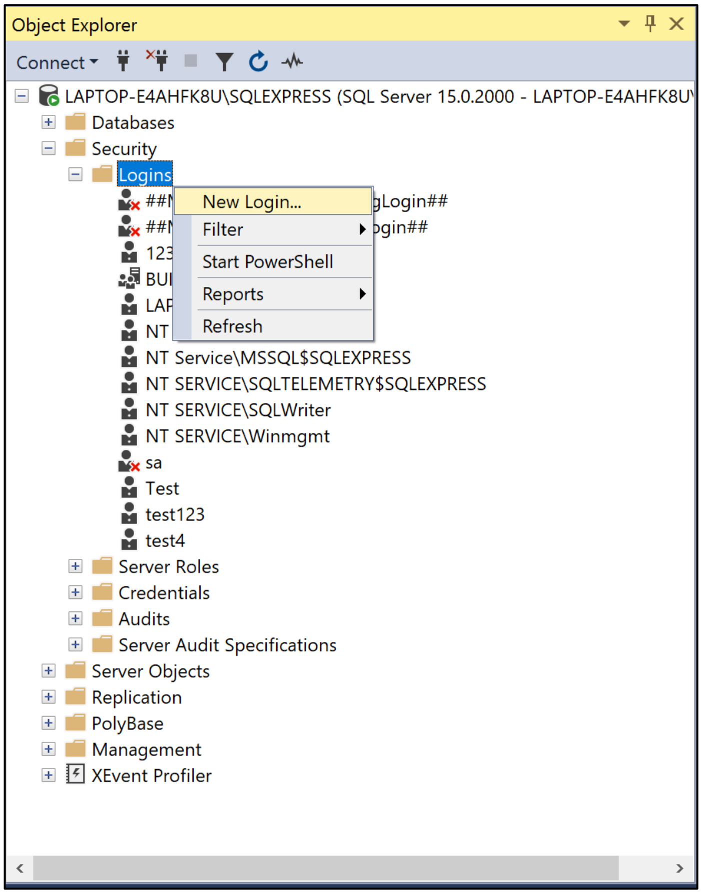
- **Dario →** For instance, customer can read transactions, but they can't update or delete. Accountant can read, edit or delete records etc. Create a user as customer_yourID and password customer. When you login with this account you should be able to read and write only on selected tables that are related to customer such as customer, account, loan and payment tables. Provide testing query script after you enforced the privileges.

Creation of user using SQL Server	Creation of user using T-SQL
User creation	**User creation**

Creation of user using T-SQL

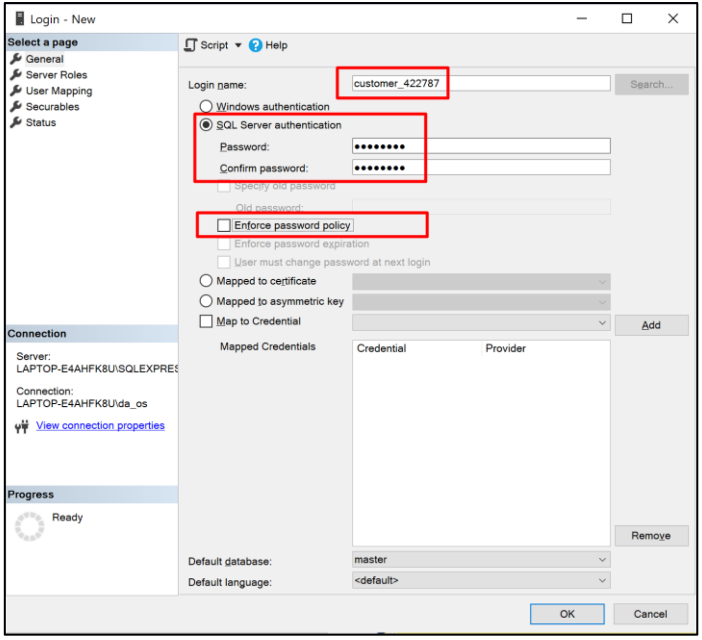
User creation

```
--Creating login customer_422787
USE [sksbank]
GO
DROP USER [customer_422787]
GO
USE [master]
GO
DROP LOGIN [customer_422787]
GO
CREATE LOGIN [customer_422787] WITH
PASSWORD=N'customer',
```



Object Explorer

LAPTOP-E4AHFK8U\SQLEXPRESS (SQL Server 15.0.2000 - LAPTOP-E4AHFK8U
- Databases
- Security
  - Logins
    - ##M...gLogin##
    - ##M...ogin##
    - 123
    - BUI
    - LAP
    - NT
    - NT Service\MSSQL$SQLEXPRESS
    - NT SERVICE\SQLTELEMETRY$SQLEXPRESS
    - NT SERVICE\SQLWriter
    - NT SERVICE\Winmgmt
    - sa
    - Test
    - test123
    - test4
  - Server Roles
  - Credentials
  - Audits
  - Server Audit Specifications
- Server Objects
- Replication
- PolyBase
- Management
- XEvent Profiler



Login - New

Login name: customer_422787

○ 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: master

Default language: <default>

Connection

Server:
LAPTOP-E4AHFK8U\SQLEXPRES

Connection:
LAPTOP-E4AHFK8U\da_os

View connection properties

Progress

Ready

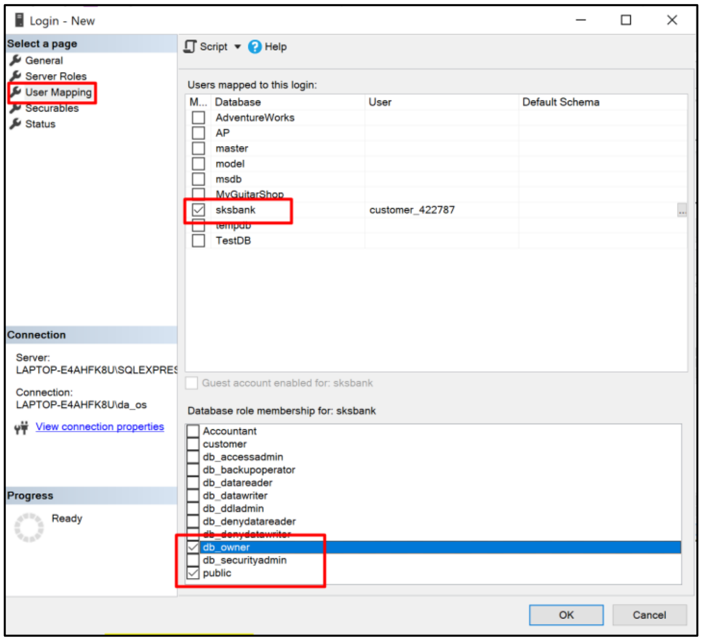
```
DEFAULT_DATABASE=[master],
CHECK_EXPIRATION=OFF, CHECK_POLICY=OFF
--Creating user customer_422787 for database
sksbank
USE [sksbank]
GO
CREATE USER [customer_422787] FOR LOGIN
[customer_422787]
ALTER ROLE [db_owner] ADD MEMBER
[customer_422787]
GO
```

Granting and denying privileges

```
--Denying permissions for user customer_422787
USE [sksbank]
GO
DENY SELECT, INSERT, DELETE, UPDATE ON
[dbo].[Branches] TO [customer_422787]
DENY SELECT, INSERT, DELETE, UPDATE ON
[dbo].[Customers] TO [customer_422787]
DENY SELECT, INSERT, DELETE, UPDATE ON
[dbo].[Employees] TO [customer_422787]
DENY SELECT, INSERT, DELETE, UPDATE ON
[dbo].[Locations] TO [customer_422787]
GO
```

Testing

```
--Testing
INSERT INTO Locations VALUES
(6,'Edmonton','AB','Canada')
DELETE FROM Locations WHERE Location_ID = 6
SELECT * FROM Locations
```



Server:
LAPTOP-E4AHFK8U\SQLEXPRES

Connection:
LAPTOP-E4AHFK8U\da_os

Ready

Users mapped to this login:

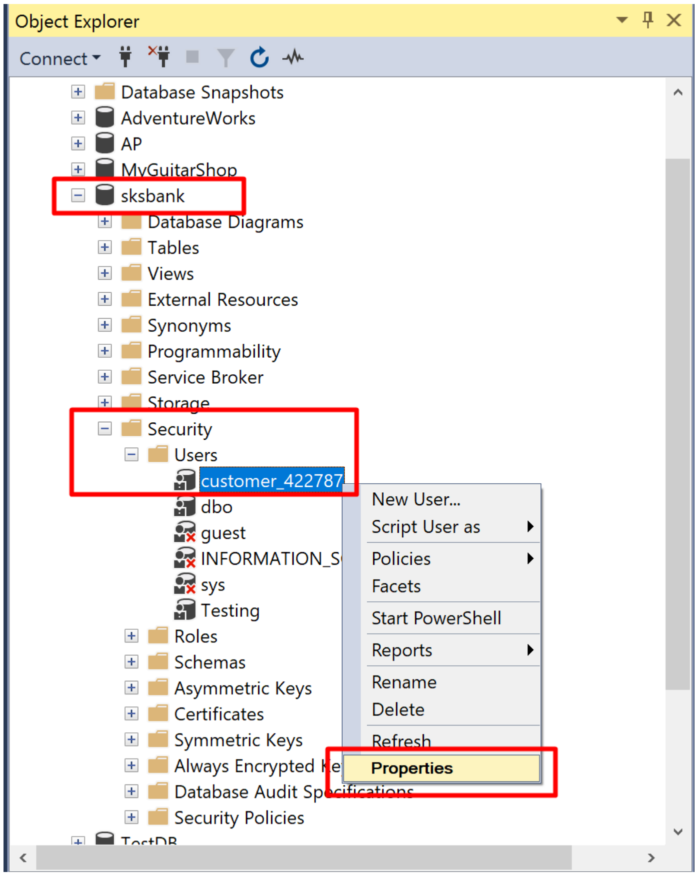
M...	Database	User	Default Schema
□	AdventureWorks		
□	AP		
□	master		
□	model		
□	msdb		
□	MyGuitarShop		
☑	sksbank	customer_422787	
□	tempdb		
□	TestDB		

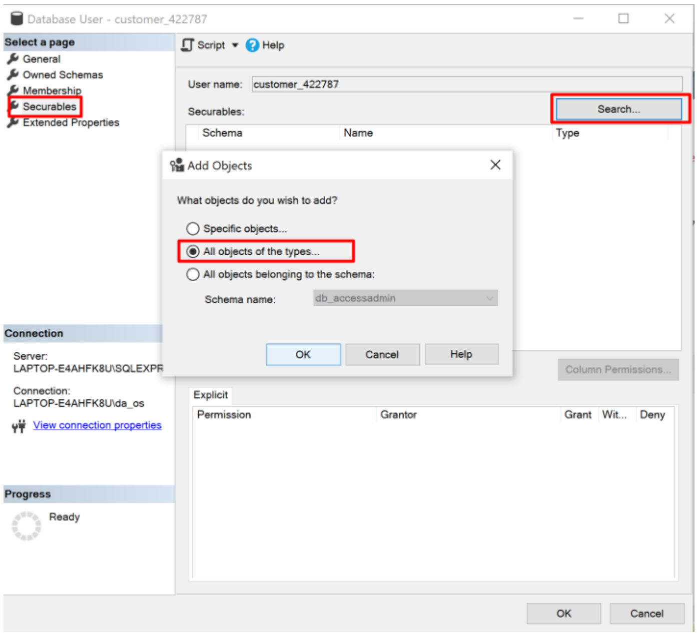
□ Guest account enabled for: sksbank

Database role membership for: sksbank

□ Accountant
□ customer
□ db_accessadmin
□ db_backupoperator
□ db_datareader
□ db_datawriter
□ db_ddladmin
□ db_denydatareader
□ db_denydatawriter
☑ db_owner
□ db_securityadmin
☑ public

Granting and denying privileges





User name: customer_422787

Securables:

Schema	Name	Type

What objects do you wish to add?

○ Specific objects...

● All objects of the types...

○ All objects belonging to the schema:

Schema name: db_accessadmin

Connection

Server:
LAPTOP-E4AHFK8U\SQLEXPR

Connection:
LAPTOP-E4AHFK8U\da_os

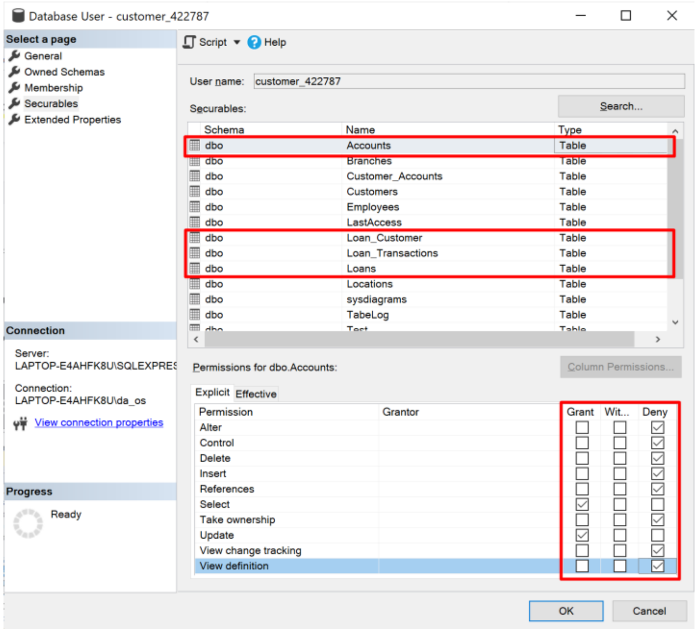
View connection properties

Progress

Ready

Explicit

Permission	Grantor	Grant	Wit...	Deny



User name: customer_422787

Securables:

Schema	Name	Type
dbo	Accounts	Table
dbo	Branches	Table
dbo	Customer_Accounts	Table
dbo	Customers	Table
dbo	Employees	Table
dbo	LastAccess	Table
dbo	Loan_Customer	Table
dbo	Loan_Transactions	Table
dbo	Loans	Table
dbo	Locations	Table
dbo	sysdiagrams	Table
dbo	TabeLog	Table
dbo	Test	Table

Connection

Server:
LAPTOP-E4AHFK8U\SQLEXPRES

Connection:
LAPTOP-E4AHFK8U\da_os

View connection properties

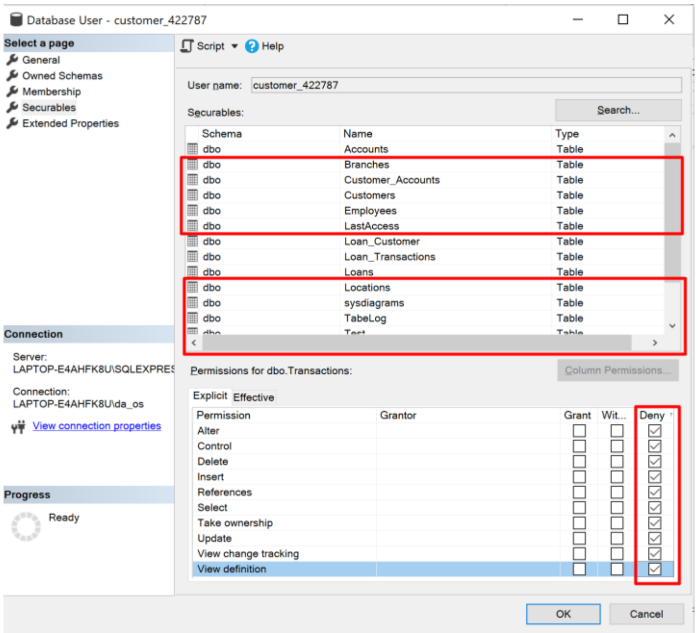
Progress

Ready

Permissions for dbo.Accounts:

Explicit Effective

Permission	Grantor	Grant	Wit...	Deny
Alter		□	□	☑
Control		□	□	☑
Delete		□	□	☑
Insert		□	□	☑
References		□	□	☑
Select		☑	□	□
Take ownership		□	□	☑
Update		☑	□	□
View change tracking		□	□	☑
View definition		□	□	☑



Database User - customer_422787

Select a page
- General
- Owned Schemas
- Membership
- Securables
- Extended Properties

Script ▾ Help

User name: customer_422787

Securables:

Schema	Name	Type
dbo	Accounts	Table
dbo	Branches	Table
dbo	Customer_Accounts	Table
dbo	Customers	Table
dbo	Employees	Table
dbo	LastAccess	Table
dbo	Loan_Customer	Table
dbo	Loan_Transactions	Table
dbo	Loans	Table
dbo	Locations	Table
dbo	sysdiagrams	Table
dbo	TabeLog	Table
dbo	Test	Table

Connection

Server:
LAPTOP-E4AHFK8U\SQLEXPRES

Connection:
LAPTOP-E4AHFK8U\da_os

View connection properties

Progress

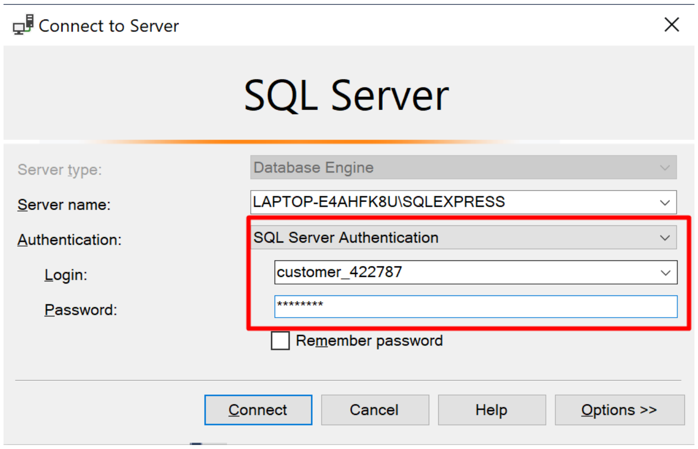
Ready

Permissions for dbo.Transactions:

Explicit Effective

Permission	Grantor	Grant	Wit...	Deny
Alter		□	□	☑
Control		□	□	☑
Delete		□	□	☑
Insert		□	□	☑
References		□	□	☑
Select		□	□	☑
Take ownership		□	□	☑
Update		□	□	☑
View change tracking		□	□	☑
View definition		□	□	☑

Login as customer_422787



Connect to Server

SQL Server

Server type: Database Engine

Server name: LAPTOP-E4AHFK8U\SQLEXPRESS

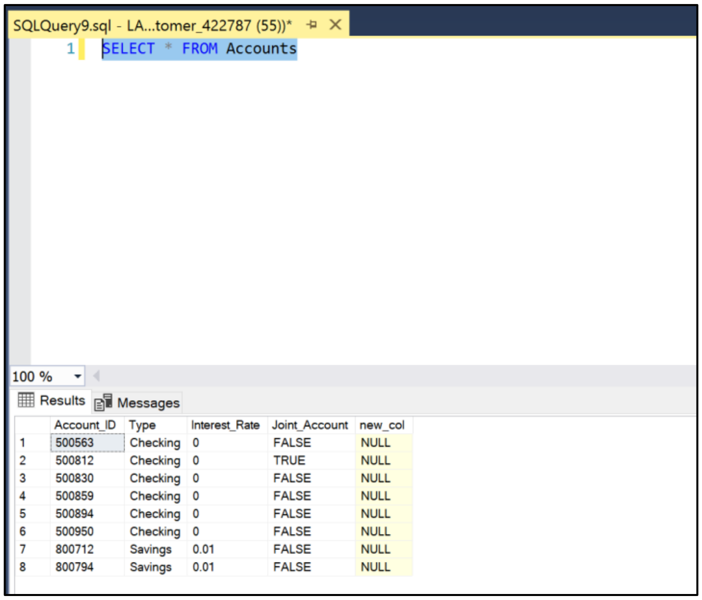
Authentication: SQL Server Authentication

Login: customer_422787

Password: ********

□ Remember password

Testing



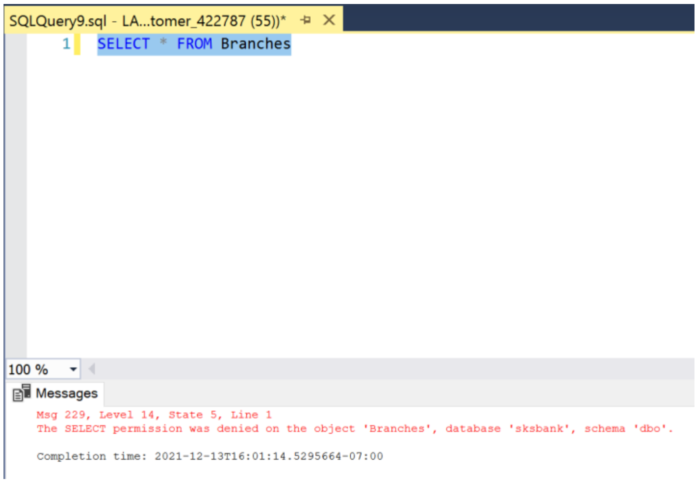
SQLQuery9.sql - LA...tomer_422787 (55))*

```sql
SELECT * FROM Accounts
```

100 %

Results | Messages

	Account_ID	Type	Interest_Rate	Joint_Account	new_col
1	500563	Checking	0	FALSE	NULL
2	500812	Checking	0	TRUE	NULL
3	500830	Checking	0	FALSE	NULL
4	500859	Checking	0	FALSE	NULL
5	500894	Checking	0	FALSE	NULL
6	500950	Checking	0	FALSE	NULL
7	800712	Savings	0.01	FALSE	NULL
8	800794	Savings	0.01	FALSE	NULL



SQLQuery9.sql - LA...tomer_422787 (55))*

```sql
SELECT * FROM Branches
```

100 %

Messages

```
Msg 229, Level 14, State 5, Line 1
The SELECT permission was denied on the object 'Branches', database 'sksbank', schema 'dbo'.

Completion time: 2021-12-13T16:01:14.5295664-07:00
```

- Dario → Create a user as accountant_yourID and password accountant. When you login with this account you should be able read all tables but cannot update account, payment and loan tables. Provide testing query script after you enforced the privileges.

Creation of user using T-SQL (Accountant_422787)

User creation

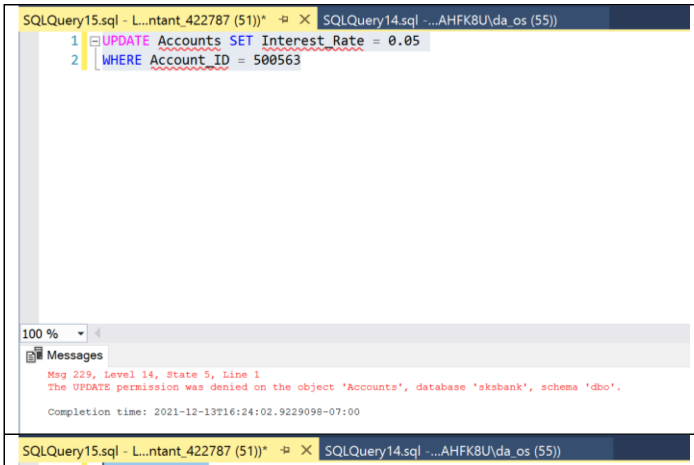
```sql
--Creating login accountant_422787
USE [sksbank]
GO
DROP USER IF EXISTS [accountant_422787]
GO
USE [master]
GO
DROP LOGIN [accountant_422787]
```

```
GO
USE [master]
CREATE LOGIN [accountant_422787] WITH PASSWORD=N'accountant', DEFAULT_DATABASE=[master],
CHECK_EXPIRATION=OFF, CHECK_POLICY=OFF
--Creating user accountant_422787 for database sksbank
USE [sksbank]
GO
CREATE USER [accountant_422787] FOR LOGIN [accountant_422787]
ALTER ROLE [db_owner] ADD MEMBER [accountant_422787]
GO
```

Granting and denying privileges

```
--Denying permissions for user accountant_422787
USE [sksbank]
GO
DENY UPDATE ON [dbo].[Accounts] TO [accountant_422787]
DENY UPDATE ON [dbo].[Customer_Accounts] TO [accountant_422787]
DENY UPDATE ON [dbo].[Loans] TO [accountant_422787]
DENY UPDATE ON [dbo].[Loan_Customer] TO [accountant_422787]
DENY UPDATE ON [dbo].[Loan_Transactions] TO [accountant_422787]
GO
```

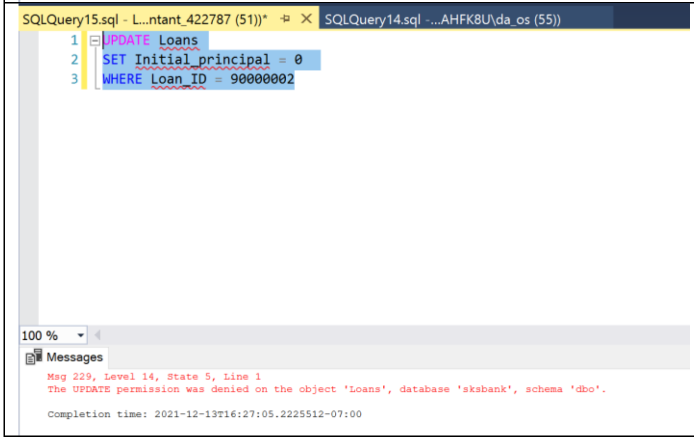
Testing


SQLQuery15.sql - L...ntant_422787 (51))*
SQLQuery14.sql -...AHFK8U\da_os (55))
100 %
Messages

```
UPDATE Accounts SET Interest_Rate = 0.05
WHERE Account_ID = 500563
```

```
Msg 229, Level 14, State 5, Line 1
The UPDATE permission was denied on the object 'Accounts', database 'sksbank', schema 'dbo'.

Completion time: 2021-12-13T16:24:02.9229098-07:00
```

```
UPDATE Accounts SET Interest_Rate = 0.05
WHERE Account_ID = 500563
```


SQLQuery15.sql - L...ntant_422787 (51))*
SQLQuery14.sql -...AHFK8U\da_os (55))
100 %
Messages

```
UPDATE Loans
SET Initial_principal = 0
WHERE Loan_ID = 90000002
```

```
Msg 229, Level 14, State 5, Line 1
The UPDATE permission was denied on the object 'Loans', database 'sksbank', schema 'dbo'.

Completion time: 2021-12-13T16:27:05.2225512-07:00
```

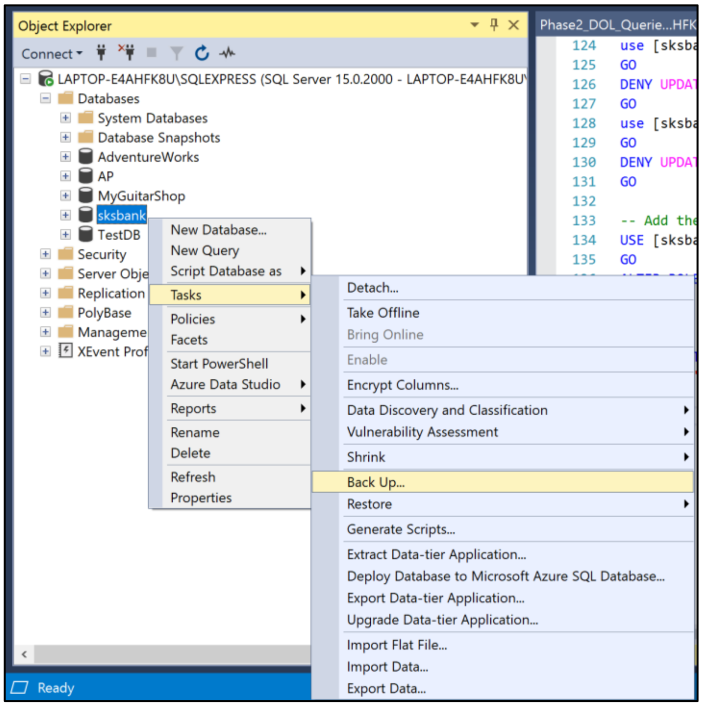
```
UPDATE Loans
SET Initial_principal = 0
WHERE Loan_ID = 90000002
```

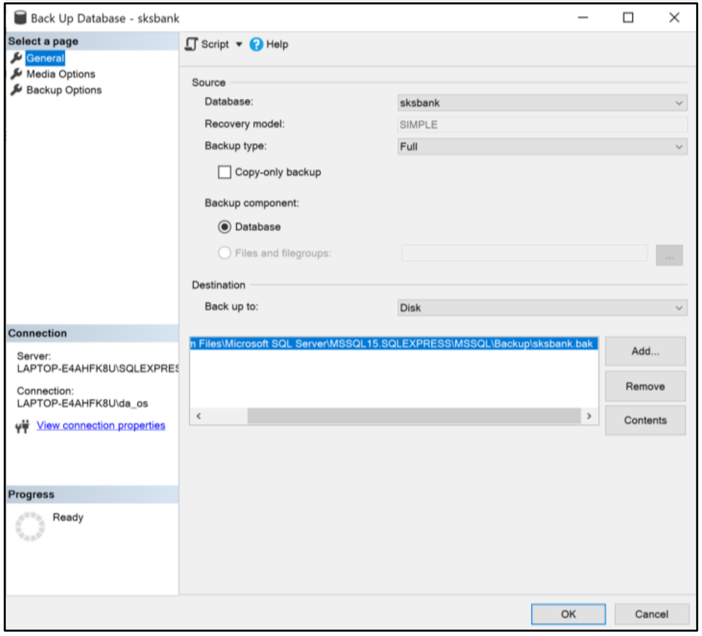
Step 5: Recovery Model and Backup

- **Dario →** Set the recovery model for your database as full recovery model. Take a full backup of your database.

SQL Server

Full Backup





Source

Database: sksbank

Recovery model: SIMPLE

Backup type: Full

□ Copy-only backup

Backup component:

● Database

○ Files and filegroups:

Destination

Back up to: Disk

n Files\Microsoft SQL Server\MSSQL15.SQLEXPRESS\MSSQL\Backup\sksbank.bak

Server:
LAPTOP-E4AHFK8U\SQLEXPRES

Connection:
LAPTOP-E4AHFK8U\da_os

Ready

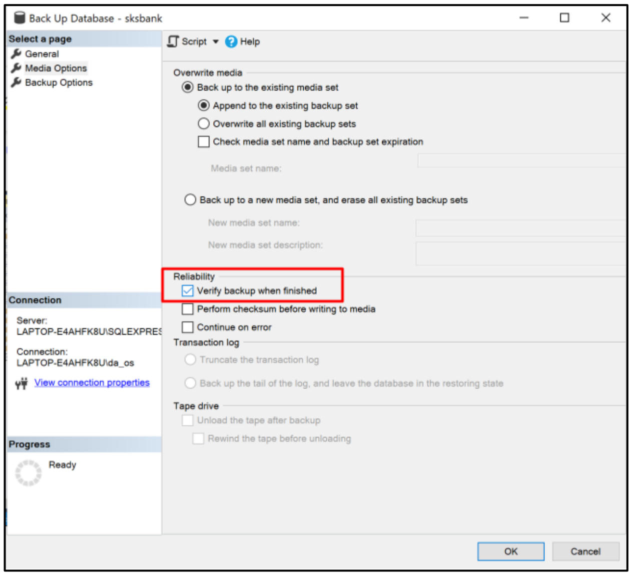
T-SQL

Full Backup

```
USE [master]
BACKUP DATABASE [sksbank]
TO DISK ='C:\Program Files\Microsoft SQL
Server\MSSQL15.SQLEXPRESS\MSSQL\Backup\sksbank
15122021.bak'
```

Restoration

```
USE [master]
RESTORE DATABASE [sksbank] FROM  DISK =
N'C:\Program Files\Microsoft SQL
Server\MSSQL15.SQLEXPRESS\MSSQL\Backup\sksbank
15122021.bak'
GO
```



Back Up Database - sksbank

Overwrite media

● Back up to the existing media set

● Append to the existing backup set

○ Overwrite all existing backup sets

□ Check media set name and backup set expiration

Media set name:

○ Back up to a new media set, and erase all existing backup sets

New media set name:

New media set description:

Reliability

☑ Verify backup when finished

□ Perform checksum before writing to media

□ Continue on error

Transaction log

○ Truncate the transaction log

○ Back up the tail of the log, and leave the database in the restoring state

Tape drive

□ Unload the tape after backup

□ Rewind the tape before unloading

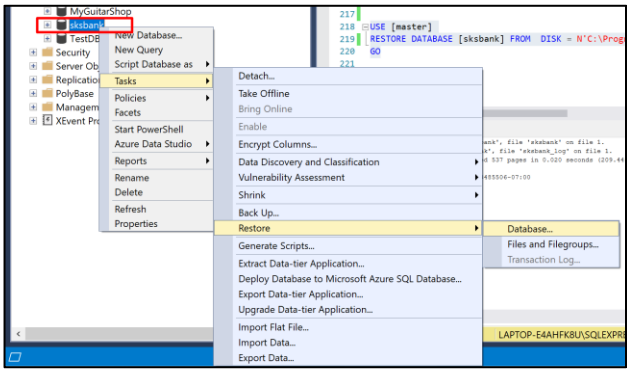
Server:
LAPTOP-E4AHFK8U\SQLEXPRES

Connection:
LAPTOP-E4AHFK8U\da_os

View connection properties

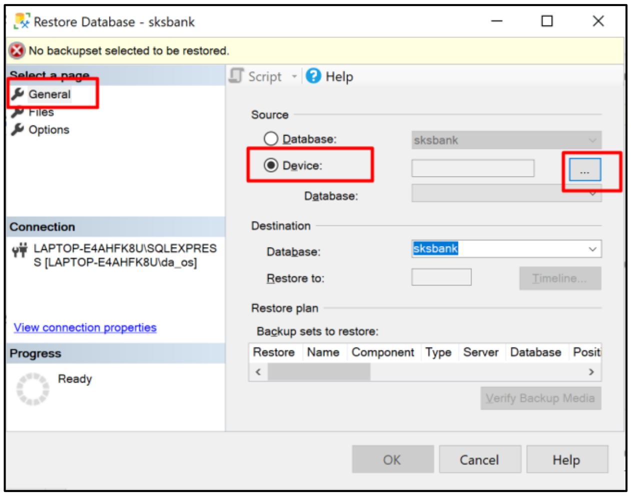
Ready

Restoration



```
USE [master]
RESTORE DATABASE [sksbank] FROM  DISK = N'C:\Prog
GO
```

```
ank', file 'sksbank' on file 1.
k', file 'sksbank_log' on file 1.
d 537 pages in 0.020 seconds (209.44
485506-07:00
```



Restore Database - sksbank

No backupset selected to be restored.

Source

○ Database: sksbank

● Device:

Database:

Destination

Database: sksbank

Restore to:

Restore plan

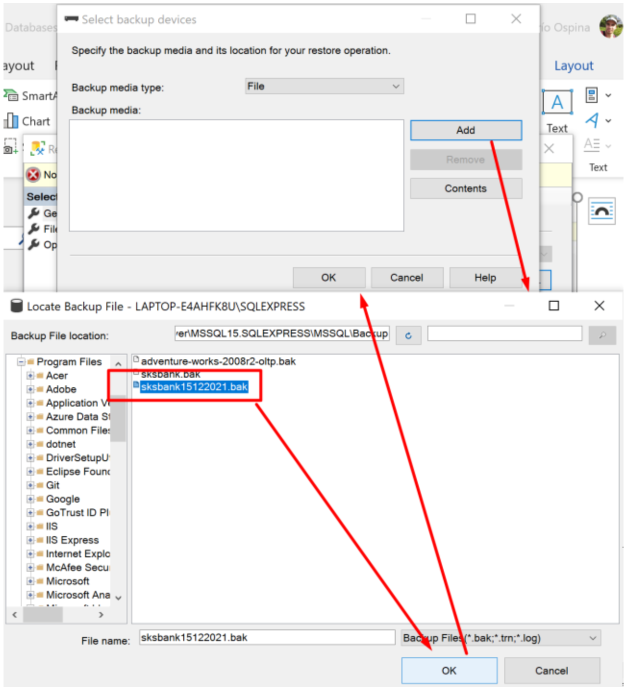
Backup sets to restore:

Restore	Name	Component	Type	Server	Database	Posit

LAPTOP-E4AHFK8U\SQLEXPRES S [LAPTOP-E4AHFK8U\da_os]

View connection properties

Ready



Select backup devices

Specify the backup media and its location for your restore operation.

Backup media type: File

Backup media:

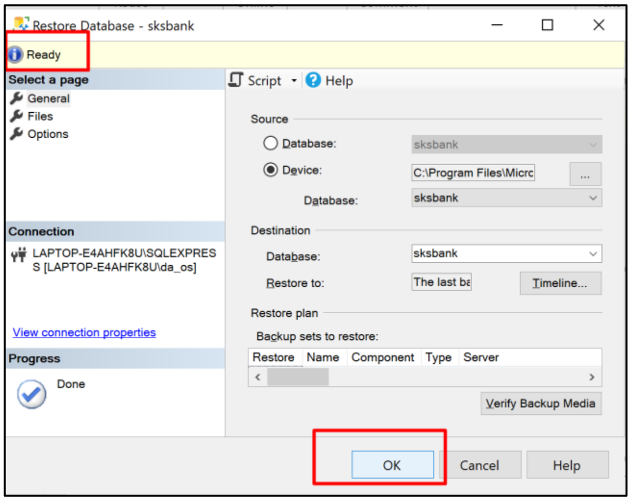
Locate Backup File - LAPTOP-E4AHFK8U\SQLEXPRESS

Backup File location: rer\MSSQL15.SQLEXPRESS\MSSQL\Backup

adventure-works-2008r2-oltp.bak
sksbank.bak
sksbank15122021.bak

File name: sksbank15122021.bak

Backup Files(*.bak;*.trn;*.log)



Restore Database - sksbank

Ready

Source

○ Database: sksbank

● Device: C:\Program Files\Micrc

Database: sksbank

Destination

Database: sksbank

Restore to: The last ba

Restore plan

Backup sets to restore:

Restore	Name	Component	Type	Server

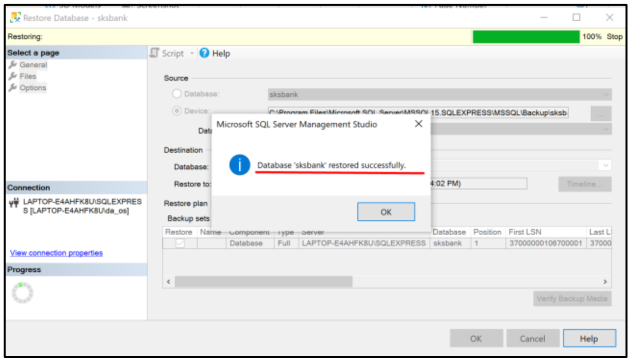
Connection

LAPTOP-E4AHFK8U\SQLEXPRESS [LAPTOP-E4AHFK8U\da_os]

View connection properties

Progress

Done



Restore Database - sksbank

Restoring: 100% Stop

Source

Database: sksbank

Device: C:\Program Files\Microsoft SQL Server\MSSQL15.SQLEXPRESS\MSSQL\Backup\sksb

Microsoft SQL Server Management Studio

Database 'sksbank' restored successfully.

Restore	Name	Component	Type	Server	Database	Position	First LSN	Last L
☑		Database	Full	LAPTOP-E4AHFK8U\SQLEXPRESS	sksbank	1	37000000106700001	37000

Connection

LAPTOP-E4AHFK8U\SQLEXPRESS [LAPTOP-E4AHFK8U\da_os]

View connection properties