TASK 3

There are three methods to copy copy all tables from one database to another

1. **Using Scripts (T-SQL)**

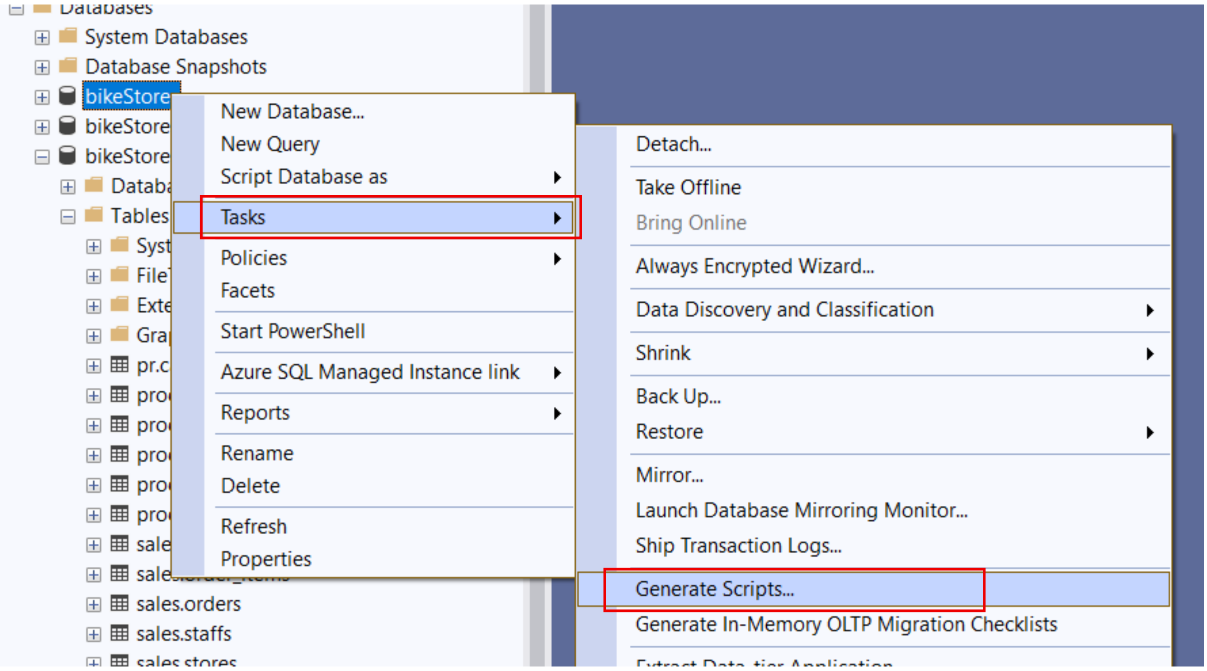
   a. Writing custom scripts to create tables and insert data.
   b. We can generate CREATE TABLE and INSERT statements manually or via SSMS scripting.
   c. Time-consuming and error-prone for large schemas.
   d. Lacks flexibility for data types, constraints, indexes unless carefully handled.

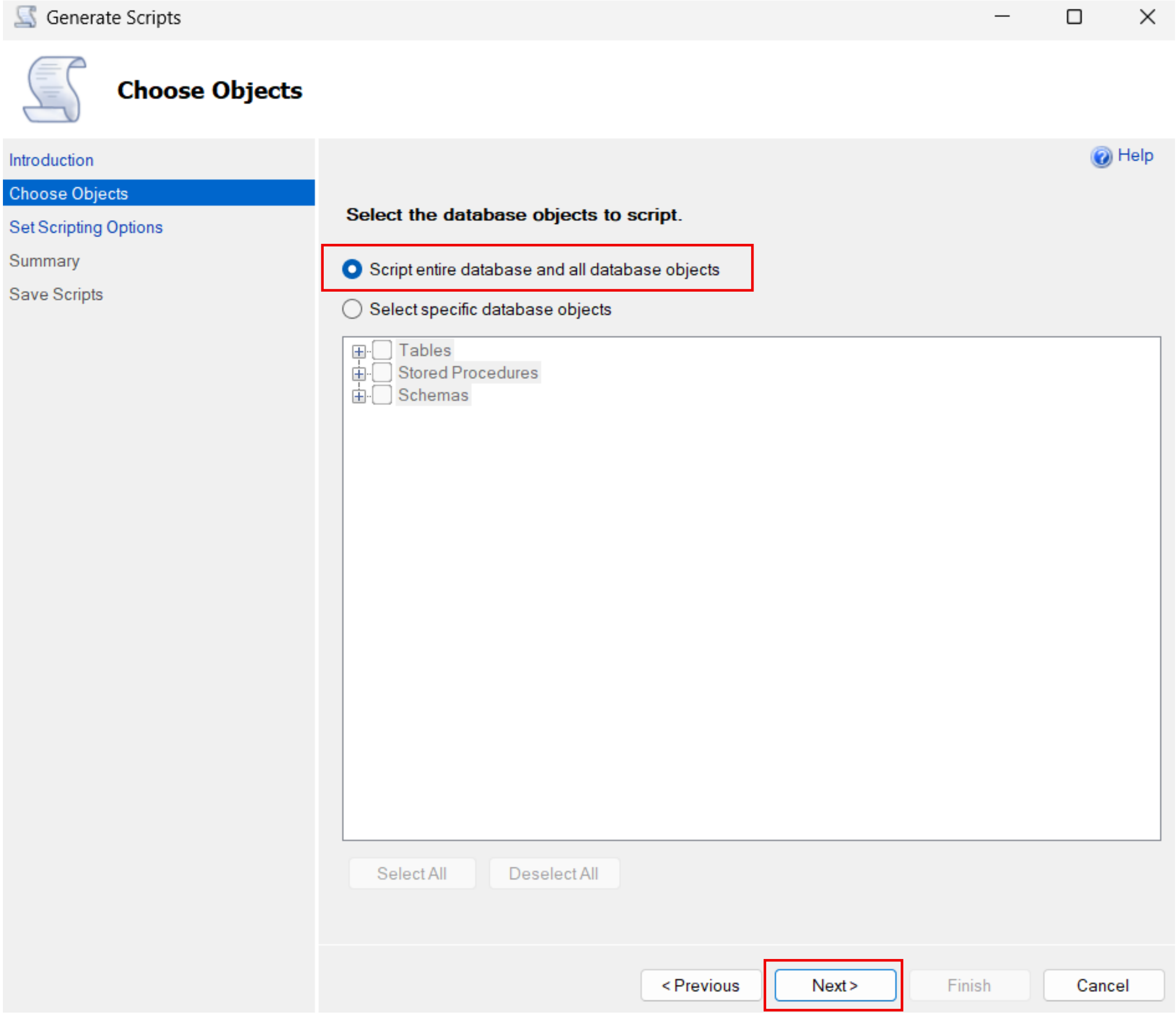
   Only recommended for small, controlled environments or when precise control over SQL is required.

2. **Using SQL Server Management Studio (SSMS) – Tasks → Generate Scripts / Export Data**

   a) Tasks → Generate Scripts

   - Allows scripting out all schema (tables, views, stored procedures, etc.) and optionally data.
   - Can be used to recreate the entire DB in another instance.
   - Flexible, but may not handle large datasets well if exporting data as inserts.



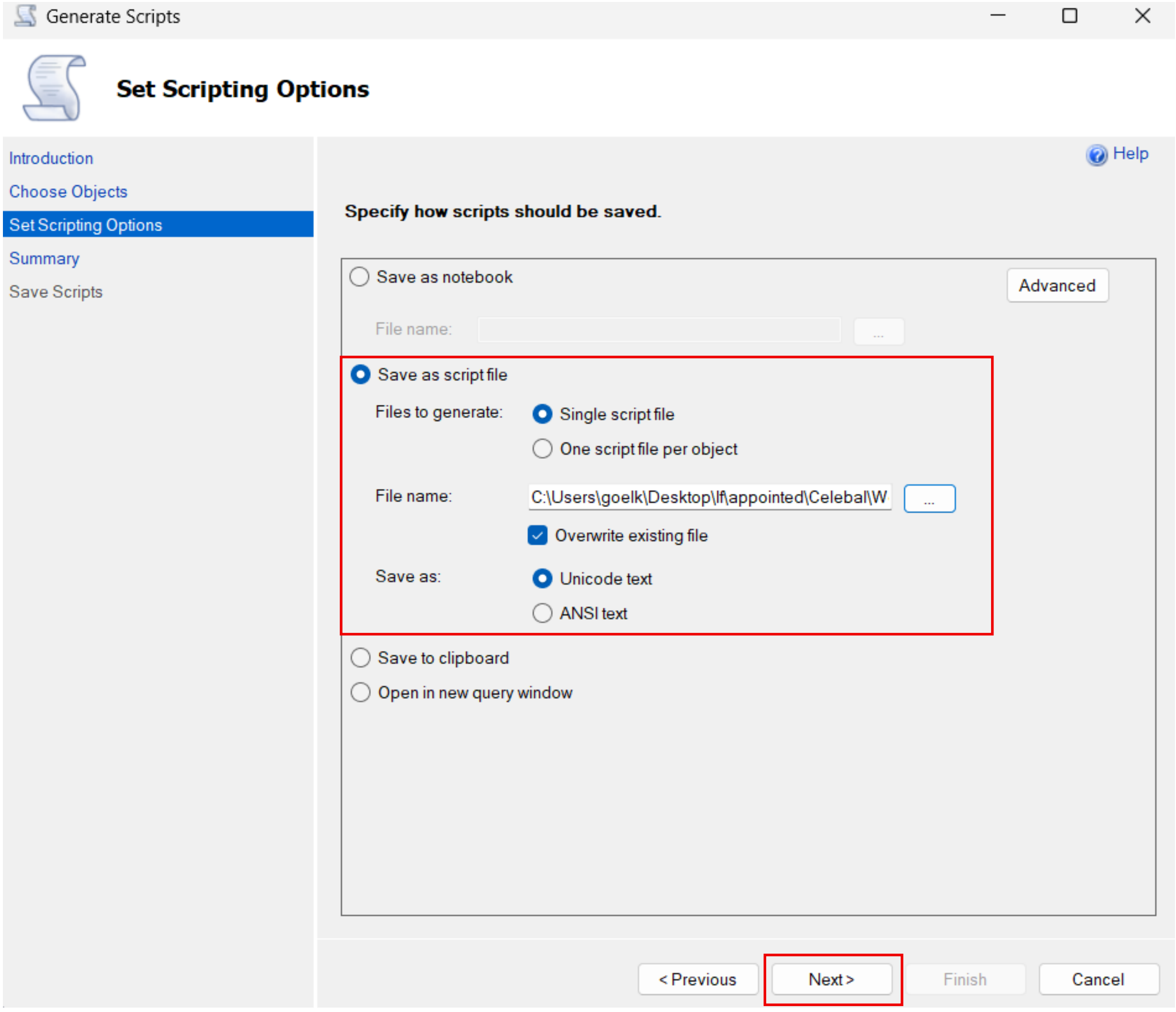


Select the database objects to script.

● Script entire database and all database objects

○ Select specific database objects

- □ Tables
- □ Stored Procedures
- □ Schemas



Set Scripting Options

Specify how scripts should be saved.

○ Save as notebook

File name:

● Save as script file

Files to generate: ● Single script file

○ One script file per object

File name: C:\Users\goelk\Desktop\lf\appointed\Celebal\W

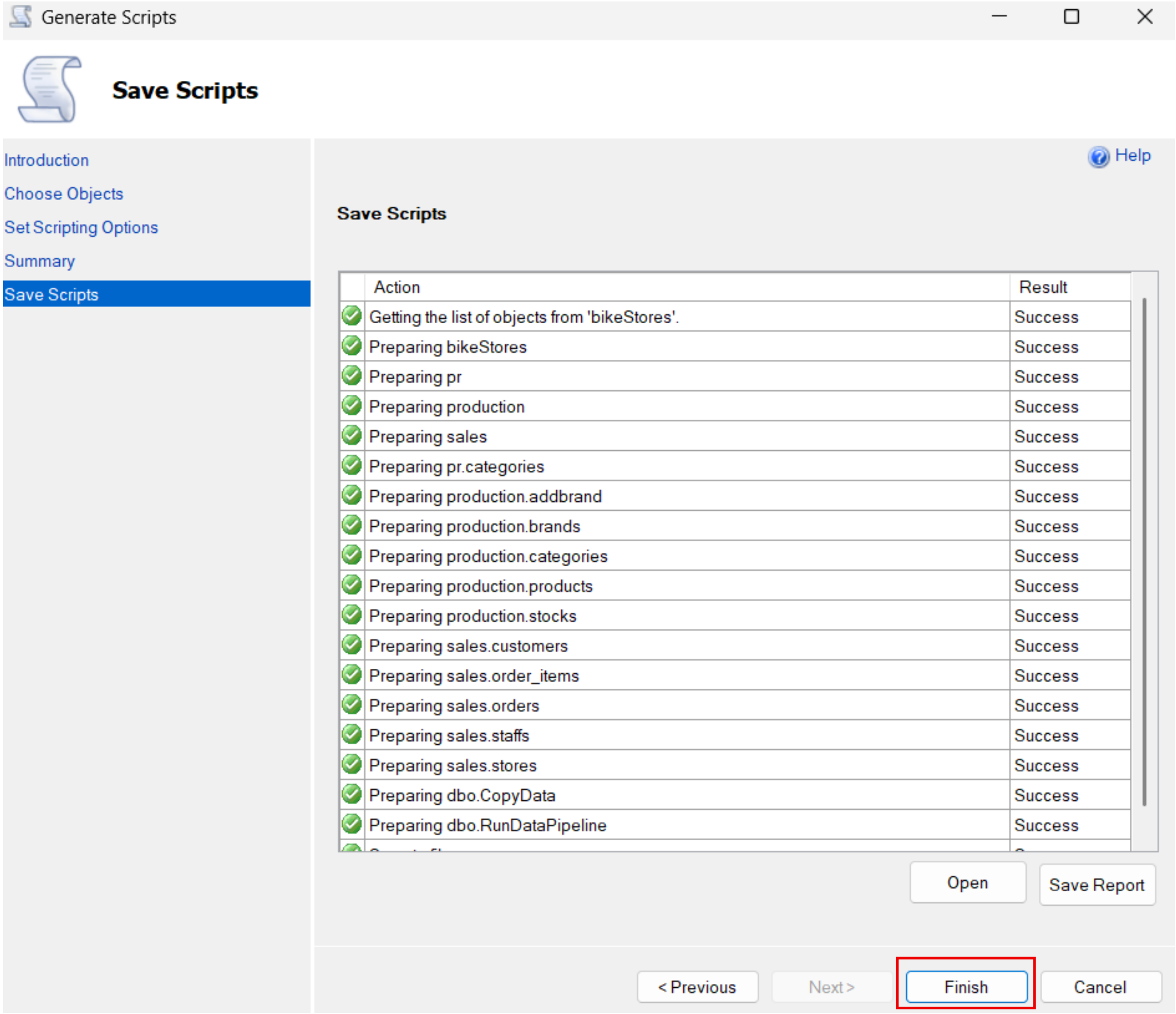
☑ Overwrite existing file

Save as: ● Unicode text

○ ANSI text

○ Save to clipboard

○ Open in new query window



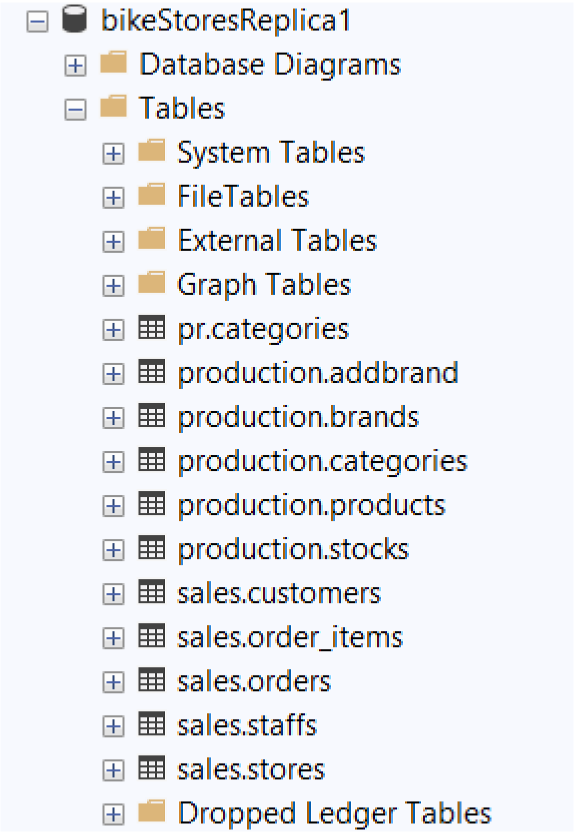
Save Scripts

	Action	Result
✓	Getting the list of objects from 'bikeStores'.	Success
✓	Preparing bikeStores	Success
✓	Preparing pr	Success
✓	Preparing production	Success
✓	Preparing sales	Success
✓	Preparing pr.categories	Success
✓	Preparing production.addbrand	Success
✓	Preparing production.brands	Success
✓	Preparing production.categories	Success
✓	Preparing production.products	Success
✓	Preparing production.stocks	Success
✓	Preparing sales.customers	Success
✓	Preparing sales.order_items	Success
✓	Preparing sales.orders	Success
✓	Preparing sales.staffs	Success
✓	Preparing sales.stores	Success
✓	Preparing dbo.CopyData	Success
✓	Preparing dbo.RunDataPipeline	Success

Run the script generated in the specified path. (Script generated is present in Task3 folder, script name: **transfer.sql**)

In this method, we have to modify the script and update the database name. The scripts are generated for the source database, change the source database name to the destination database name everywhere in the script.

OUTPUT:

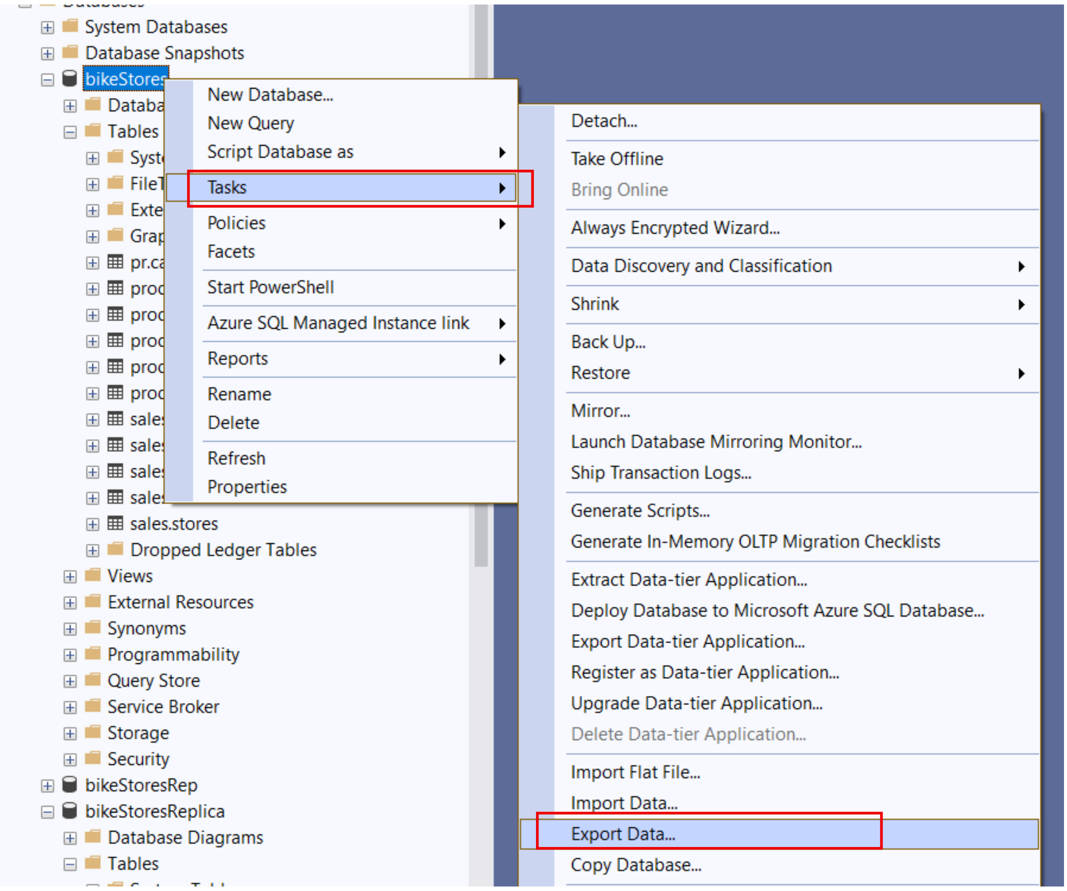


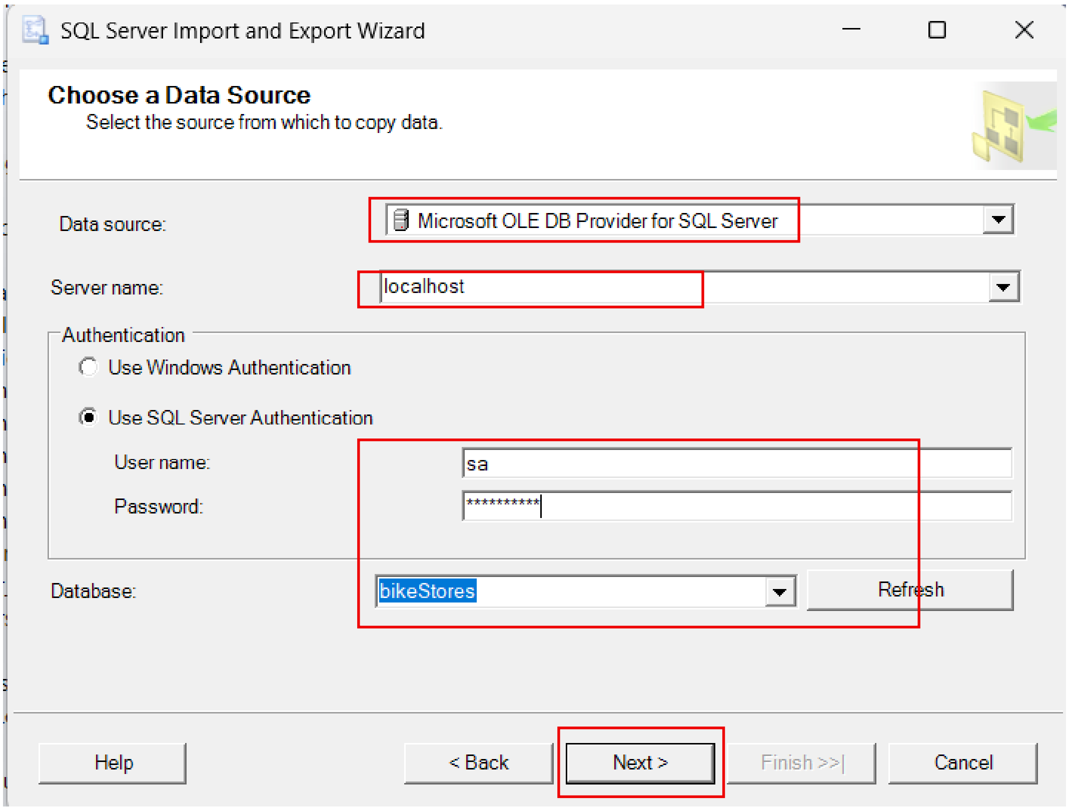
```
⊟ bikeStoresReplica1
    ⊞ Database Diagrams
    ⊟ Tables
        ⊞ System Tables
        ⊞ FileTables
        ⊞ External Tables
        ⊞ Graph Tables
        ⊞ pr.categories
        ⊞ production.addbrand
        ⊞ production.brands
        ⊞ production.categories
        ⊞ production.products
        ⊞ production.stocks
        ⊞ sales.customers
        ⊞ sales.order_items
        ⊞ sales.orders
        ⊞ sales.staffs
        ⊞ sales.stores
        ⊞ Dropped Ledger Tables
```

b) Tasks → Export Data (Wizard)

- Uses the SQL Server Import and Export Wizard.
- Good for quickly copying tables (schema + data) to another DB or server.
- We can select all tables, define transformations, etc.
- Efficient for data, but limited for schema objects like stored procedures, functions, etc.

Copy all tables from **bikeStores** database to **bikeStoresReplica** database.





Choose a Data Source

Select the source from which to copy data.

Data source: Microsoft OLE DB Provider for SQL Server

Server name: localhost

Authentication

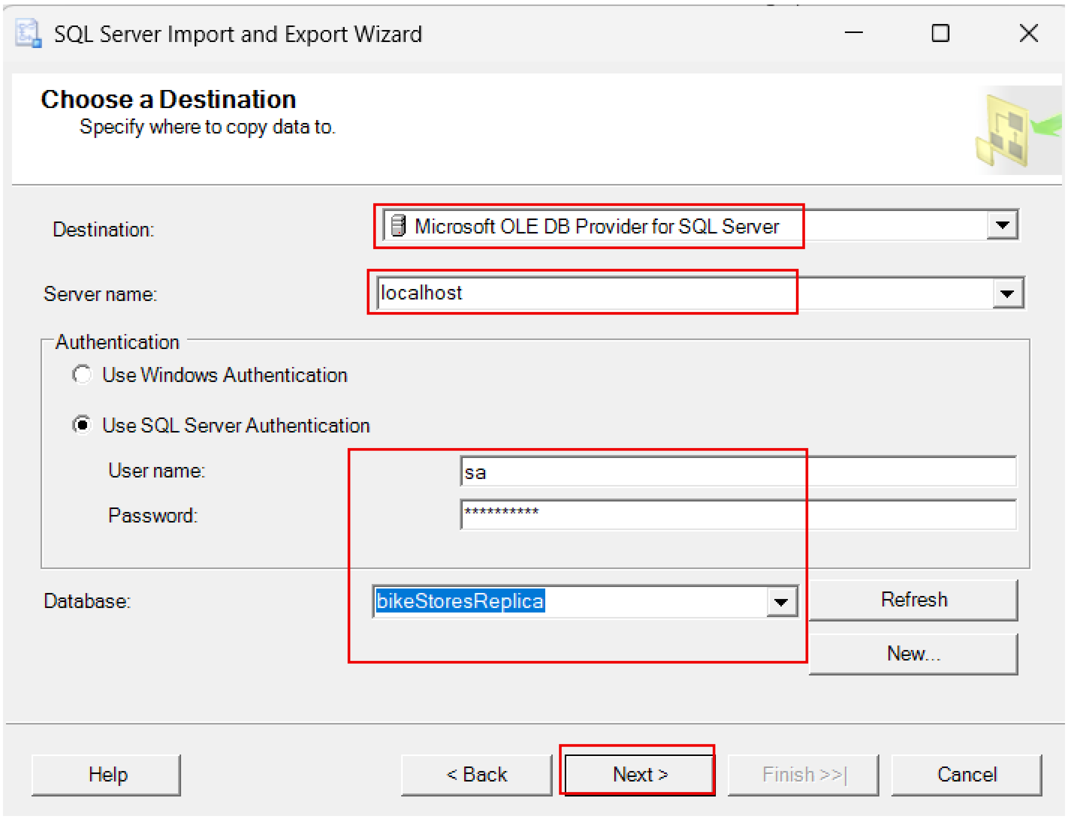
○ Use Windows Authentication

● Use SQL Server Authentication

User name: sa

Password: **********

Database: bikeStores



SQL Server Import and Export Wizard

Choose a Destination

Specify where to copy data to.

Destination: Microsoft OLE DB Provider for SQL Server

Server name: localhost

Authentication

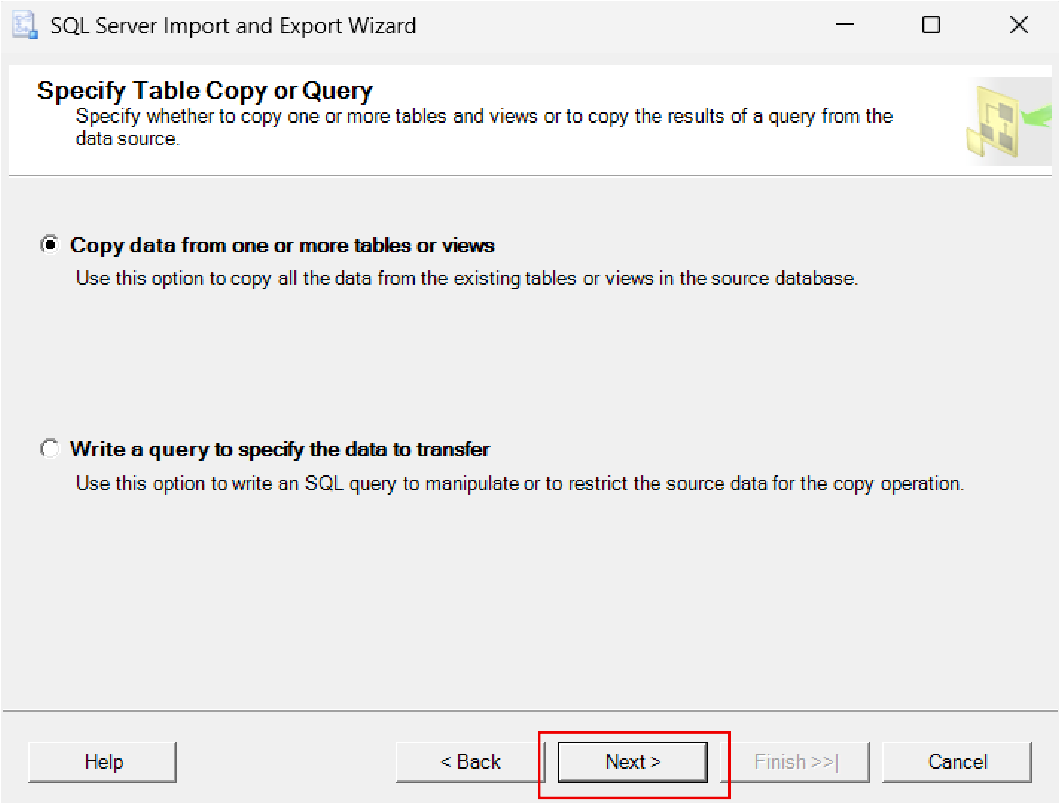
○ Use Windows Authentication

● Use SQL Server Authentication

User name: sa

Password: **********

Database: bikeStoresReplica



SQL Server Import and Export Wizard

Specify Table Copy or Query

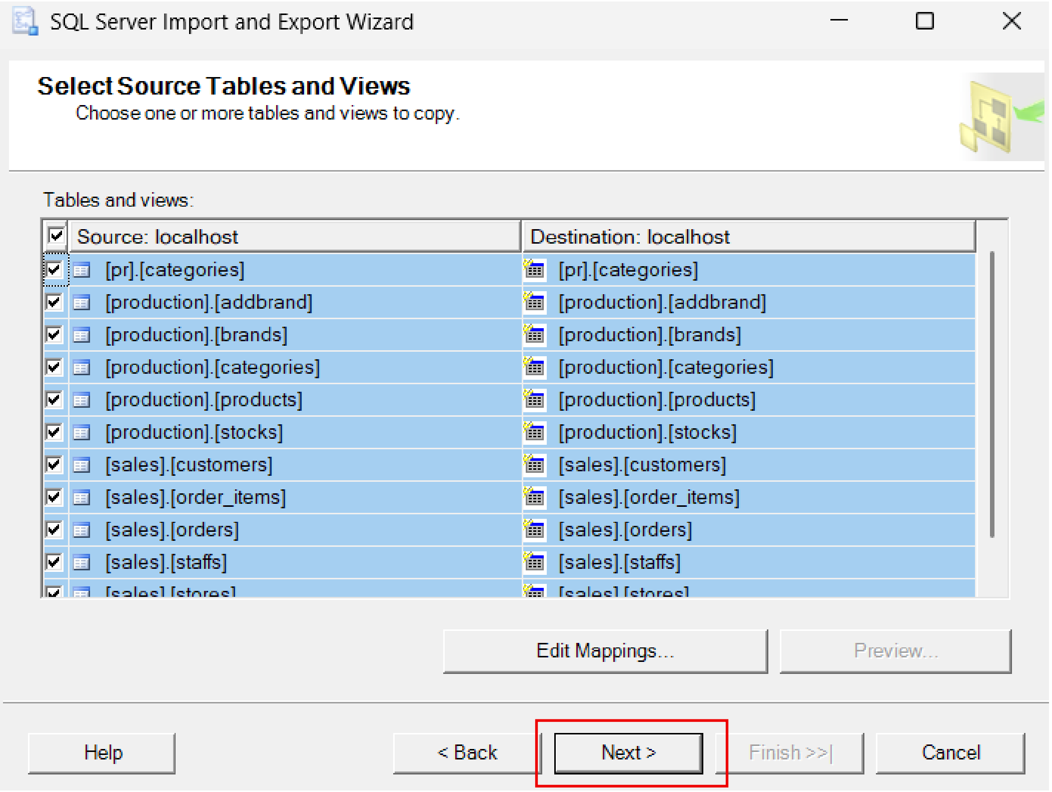
Specify whether to copy one or more tables and views or to copy the results of a query from the data source.

● **Copy data from one or more tables or views**

Use this option to copy all the data from the existing tables or views in the source database.

○ **Write a query to specify the data to transfer**

Use this option to write an SQL query to manipulate or to restrict the source data for the copy operation.



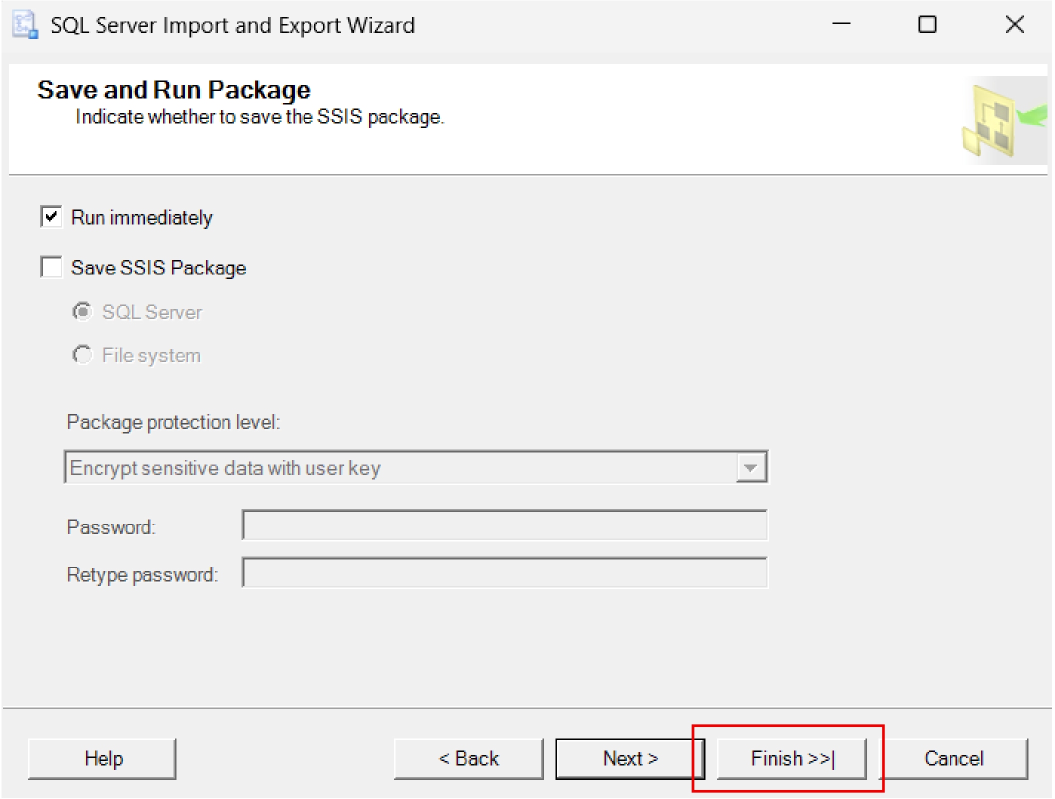
SQL Server Import and Export Wizard

Select Source Tables and Views

Choose one or more tables and views to copy.

Tables and views:

☑ Source: localhost	Destination: localhost
☑ [pr].[categories]	[pr].[categories]
☑ [production].[addbrand]	[production].[addbrand]
☑ [production].[brands]	[production].[brands]
☑ [production].[categories]	[production].[categories]
☑ [production].[products]	[production].[products]
☑ [production].[stocks]	[production].[stocks]
☑ [sales].[customers]	[sales].[customers]
☑ [sales].[order_items]	[sales].[order_items]
☑ [sales].[orders]	[sales].[orders]
☑ [sales].[staffs]	[sales].[staffs]
☑ [sales].[stores]	[sales].[stores]



SQL Server Import and Export Wizard

Save and Run Package

Indicate whether to save the SSIS package.

☑ Run immediately

□ Save SSIS Package

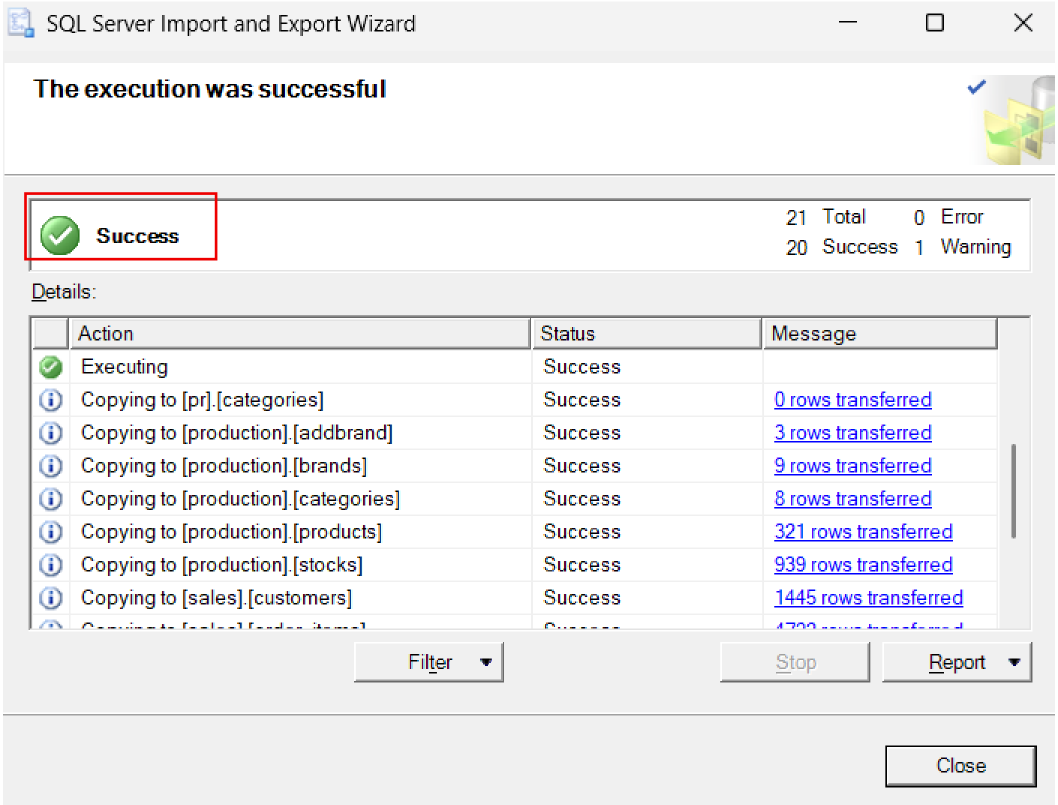
● SQL Server

○ File system

Package protection level: Encrypt sensitive data with user key

Password:

Retype password:



SQL Server Import and Export Wizard

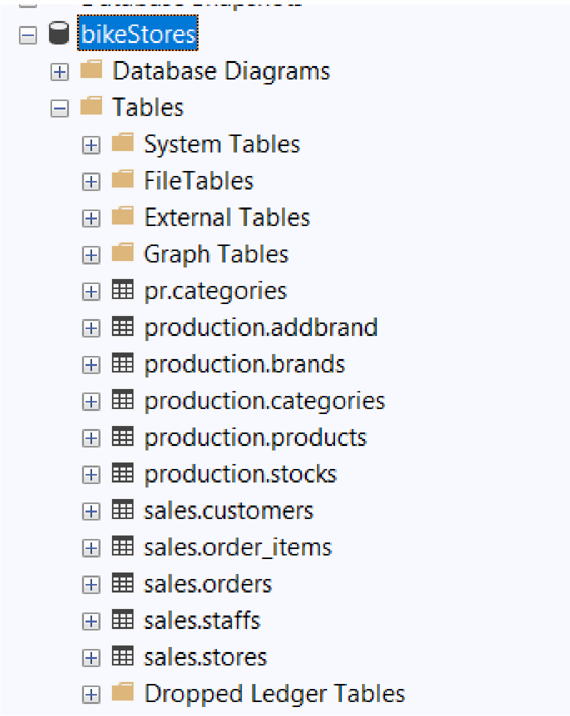
The execution was successful

✓ **Success**

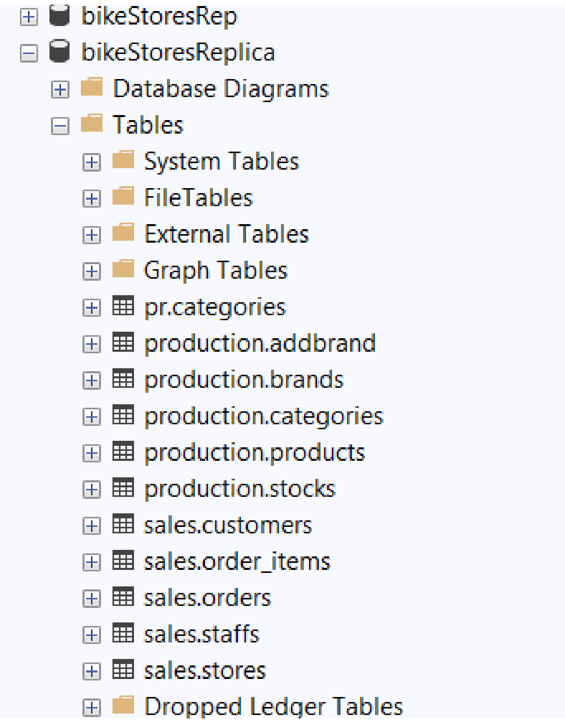
21	Total	0	Error
20	Success	1	Warning

Details:

	Action	Status	Message
✓	Executing	Success	
ⓘ	Copying to [pr].[categories]	Success	0 rows transferred
ⓘ	Copying to [production].[addbrand]	Success	3 rows transferred
ⓘ	Copying to [production].[brands]	Success	9 rows transferred
ⓘ	Copying to [production].[categories]	Success	8 rows transferred
ⓘ	Copying to [production].[products]	Success	321 rows transferred
ⓘ	Copying to [production].[stocks]	Success	939 rows transferred
ⓘ	Copying to [sales].[customers]	Success	1445 rows transferred



- ⊟ bikeStores
  - ⊞ Database Diagrams
  - ⊟ Tables
    - ⊞ System Tables
    - ⊞ FileTables
    - ⊞ External Tables
    - ⊞ Graph Tables
    - ⊞ pr.categories
    - ⊞ production.addbrand
    - ⊞ production.brands
    - ⊞ production.categories
    - ⊞ production.products
    - ⊞ production.stocks
    - ⊞ sales.customers
    - ⊞ sales.order_items
    - ⊞ sales.orders
    - ⊞ sales.staffs
    - ⊞ sales.stores
    - ⊞ Dropped Ledger Tables



- ⊞ bikeStoresRep
- ⊟ bikeStoresReplica
  - ⊞ Database Diagrams
  - ⊟ Tables
    - ⊞ System Tables
    - ⊞ FileTables
    - ⊞ External Tables
    - ⊞ Graph Tables
    - ⊞ pr.categories
    - ⊞ production.addbrand
    - ⊞ production.brands
    - ⊞ production.categories
    - ⊞ production.products
    - ⊞ production.stocks
    - ⊞ sales.customers
    - ⊞ sales.order_items
    - ⊞ sales.orders
    - ⊞ sales.staffs
    - ⊞ sales.stores
    - ⊞ Dropped Ledger Tables

3. Using Visual Studio SQL Server Integration Services (SSIS)

- SSIS is a professional-grade ETL (Extract, Transform, Load) tool.
- Supports complex data flows, error handling, transformations, automation.
- Can copy:
  - All tables and data
  - Specific subsets with transformations
  - From different data sources (e.g., Excel, Oracle) into SQL Server

Best for enterprise-level data movement, especially for large volumes or scheduled processes.

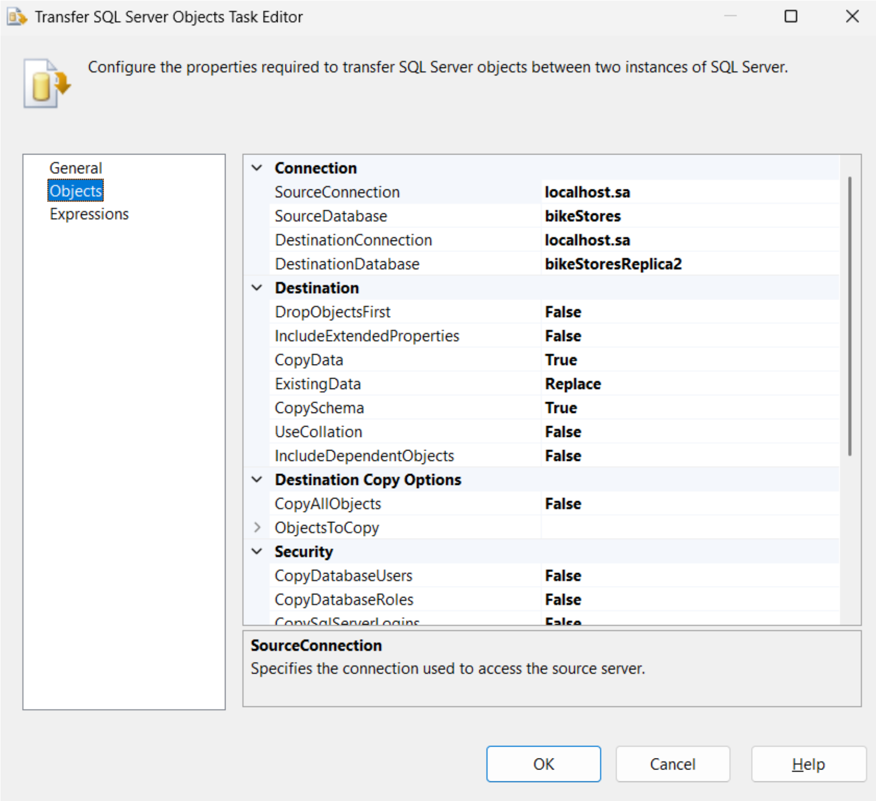
STEPS:

Open Visual Studio and Create a New SSIS Project

- Open Visual Studio.
- Click **"Create a new project"**.
- Choose **"Integration Services Project"** (the SQL SERVER INTEGRATION SERVICES PROJECTS extension must be installed in Visual Studio).
- Name the project and click **Create**.

Configure the Source (Source Database)

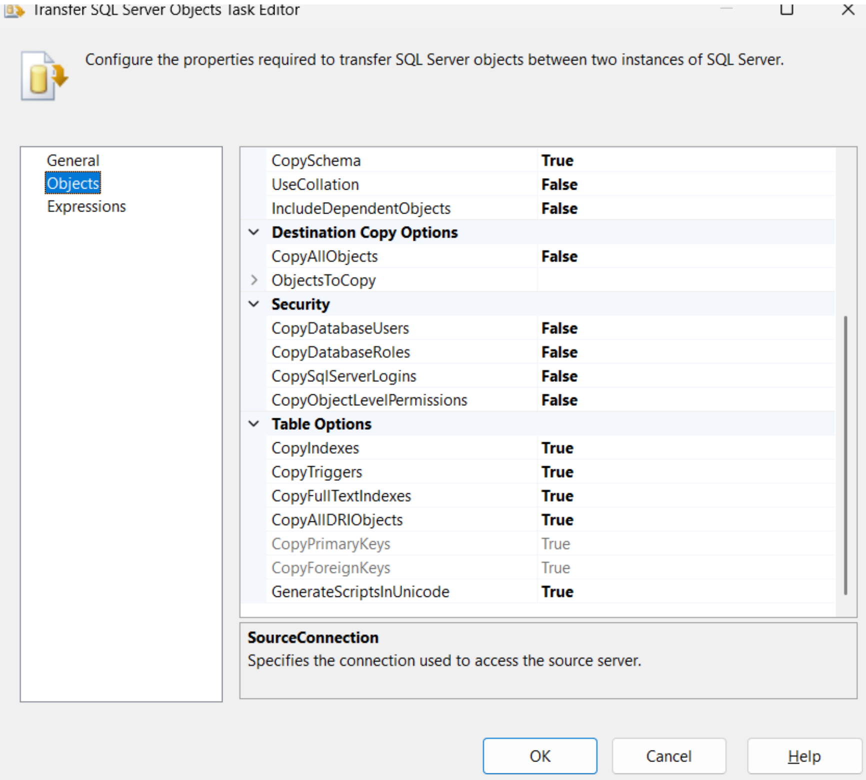
- In the Data Flow tab, drag **"Transfer SQL Server Objects Task"** from the toolbox.
- Double-click it and in the Objects section:
  - Create a **connection** to the **source database**.
  - Create a **connection** to the **source database**.
  - Enable options in **ObjectsToCopy**, **Table Options**



Configure the properties required to transfer SQL Server objects between two instances of SQL Server.

Connection	
SourceConnection	**localhost.sa**
SourceDatabase	**bikeStores**
DestinationConnection	**localhost.sa**
DestinationDatabase	**bikeStoresReplica2**
Destination	
DropObjectsFirst	**False**
IncludeExtendedProperties	**False**
CopyData	**True**
ExistingData	**Replace**
CopySchema	**True**
UseCollation	**False**
IncludeDependentObjects	**False**
Destination Copy Options	
CopyAllObjects	**False**
ObjectsToCopy	
Security	
CopyDatabaseUsers	**False**
CopyDatabaseRoles	**False**
CopySqlServerLogins	**False**

SourceConnection
Specifies the connection used to access the source server.



Configure the properties required to transfer SQL Server objects between two instances of SQL Server.

CopySchema	**True**
UseCollation	**False**
IncludeDependentObjects	**False**
Destination Copy Options	
CopyAllObjects	**False**
ObjectsToCopy	
Security	
CopyDatabaseUsers	**False**
CopyDatabaseRoles	**False**
CopySqlServerLogins	**False**
CopyObjectLevelPermissions	**False**
Table Options	
CopyIndexes	**True**
CopyTriggers	**True**
CopyFullTextIndexes	**True**
CopyAllDRIObjects	**True**
CopyPrimaryKeys	True
CopyForeignKeys	True
GenerateScriptsInUnicode	**True**

SourceConnection
Specifies the connection used to access the source server.

- **SourceConnection**: Create/select a connection to the source database.
- **DestinationConnection**: Create/select a connection to the destination database.
- Set:
  - CopyData = True

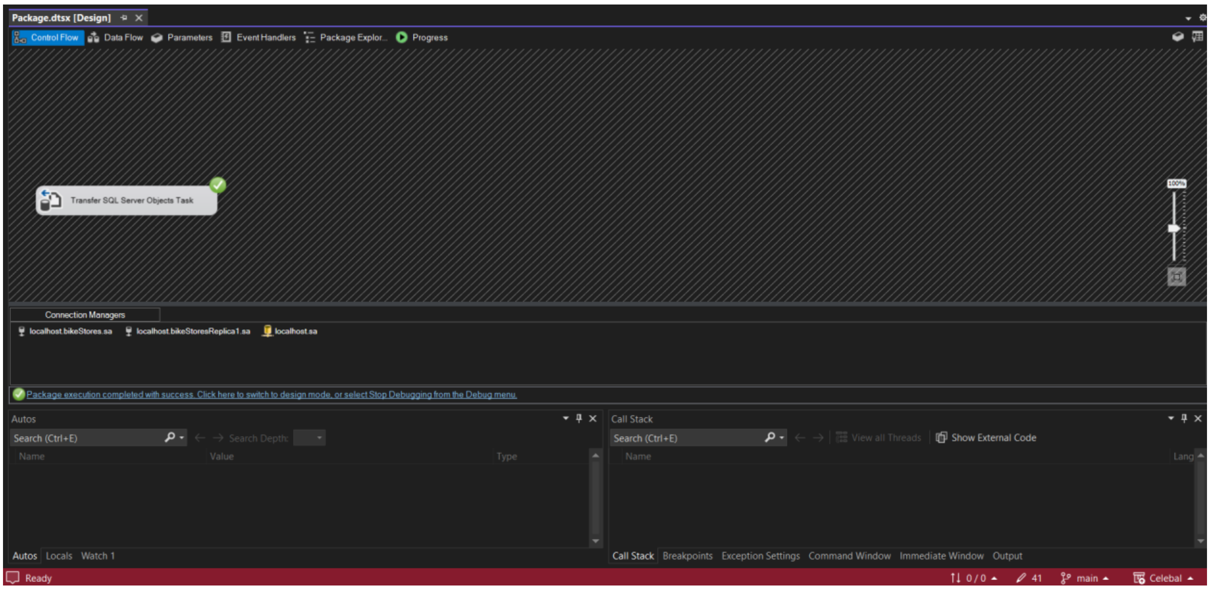
- CopySchema = True
- CopyAllObjects = True (or manually select specific tables in ObjectsToCopy)
- DropObjectsFirst = False (unless we want to replace existing tables)

2. Click OK.

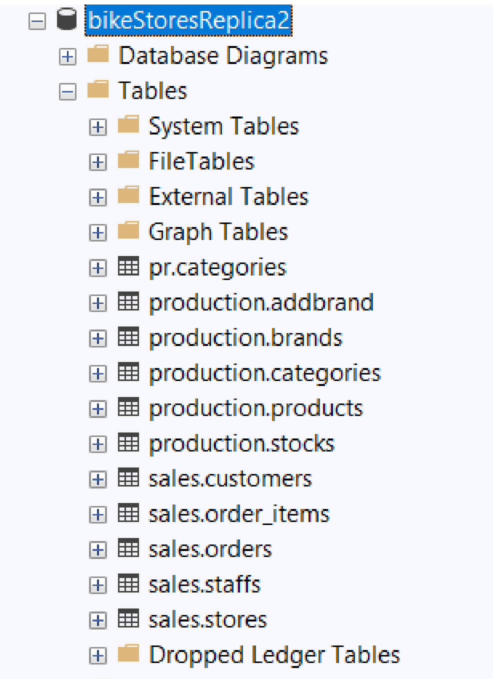
Save and Deploy (or Run Locally)

- Click **Start** (green arrow) or press **F5** to run the package.
- Visual Studio will execute the SSIS package and display progress/output.

SUCCESSFUL DATA COPY



Package execution completed with success. Click here to switch to design mode, or select Stop Debugging from the Debug menu.



- bikeStoresReplica2
  - Database Diagrams
  - Tables
    - System Tables
    - FileTables
    - External Tables
    - Graph Tables
    - pr.categories
    - production.addbrand
    - production.brands
    - production.categories
    - production.products
    - production.stocks
    - sales.customers
    - sales.order_items
    - sales.orders
    - sales.staffs
    - sales.stores
    - Dropped Ledger Tables