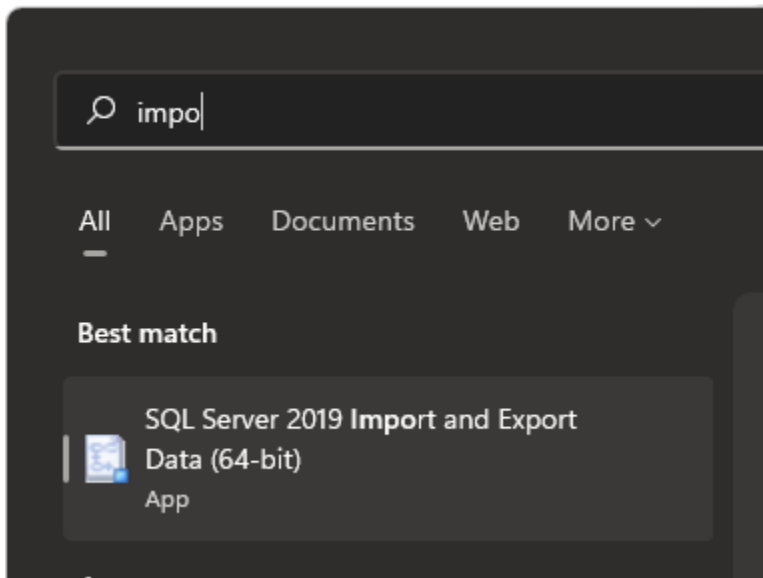
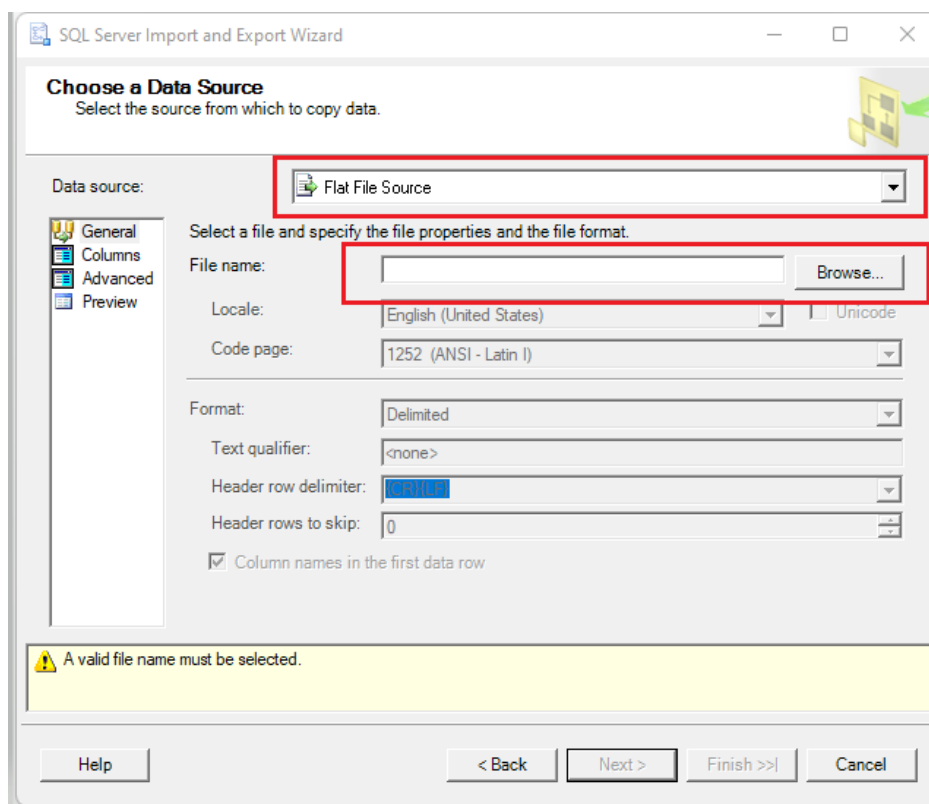


Below are the instructions on how to import raw data into a schema in SQL Server and then migrate it to MySQL Workbench.

- Launch **SQL Server 2019 Import and Export Data (64-bit)**.

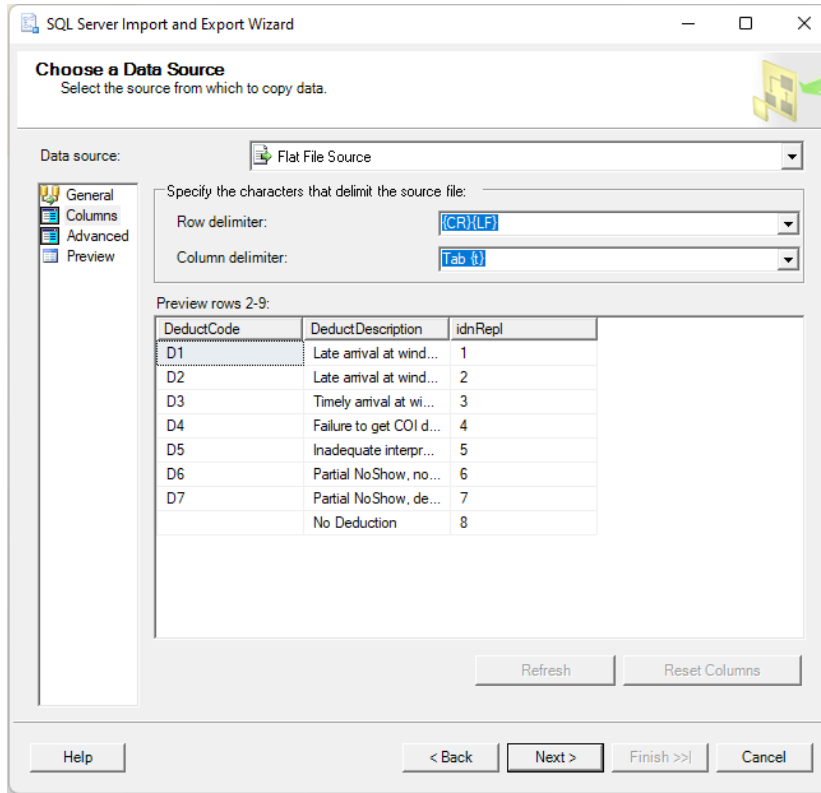


- Select **Flat File Source** for **Data source** from dropdown.
- Then click **browser** to locate and select your csv file.



- Leave other option as default then click **Next**.

- Preview the data then click **Next**.



**SQL Server Import and Export Wizard**

**Choose a Data Source**  
Select the source from which to copy data.

Data source: Flat File Source

Specify the characters that delimit the source file:

Row delimiter: (CR)(LF)

Column delimiter: Tab (t)

Preview rows 2-9:

DeductCode	DeductDescription	idnRepl
D1	Late arrival at wind...	1
D2	Late arrival at wind...	2
D3	Timely arrival at wi...	3
D4	Failure to get COI d...	4
D5	Inadequate interpr...	5
D6	Partial NoShow, no...	6
D7	Partial NoShow, de...	7
	No Deduction	8

Buttons: Help, < Back, Next >, Finish >>, Cancel

- Select **SQL Server Native Client 11.0** from the dropdown for **Destination**.
- Select your SQL server connection from the dropdown then use the appropriate connection to connect to your SQL server.
- Select an existing **Database** from the dropdown or click **New** to create a new database. Click **Next**.

SQL Server Import and Export Wizard

**Choose a Destination**  
Specify where to copy data to.

Destination: SQL Server Native Client 11.0

Server name:

Authentication

☒ Use Windows Authentication

☐ Use SQL Server Authentication

User name:

Password:

Database: <default> Refresh

New...

Help < Back Next > Finish >>| Cancel

SQL Server Import and Export Wizard

**Choose a Destination**  
Specify where to copy data to.

Destination: SQL Server Native Client 11.0

Server name:

Authentication

☒ Use Windows Authentication

☐ Use SQL Server Authentication

User name:

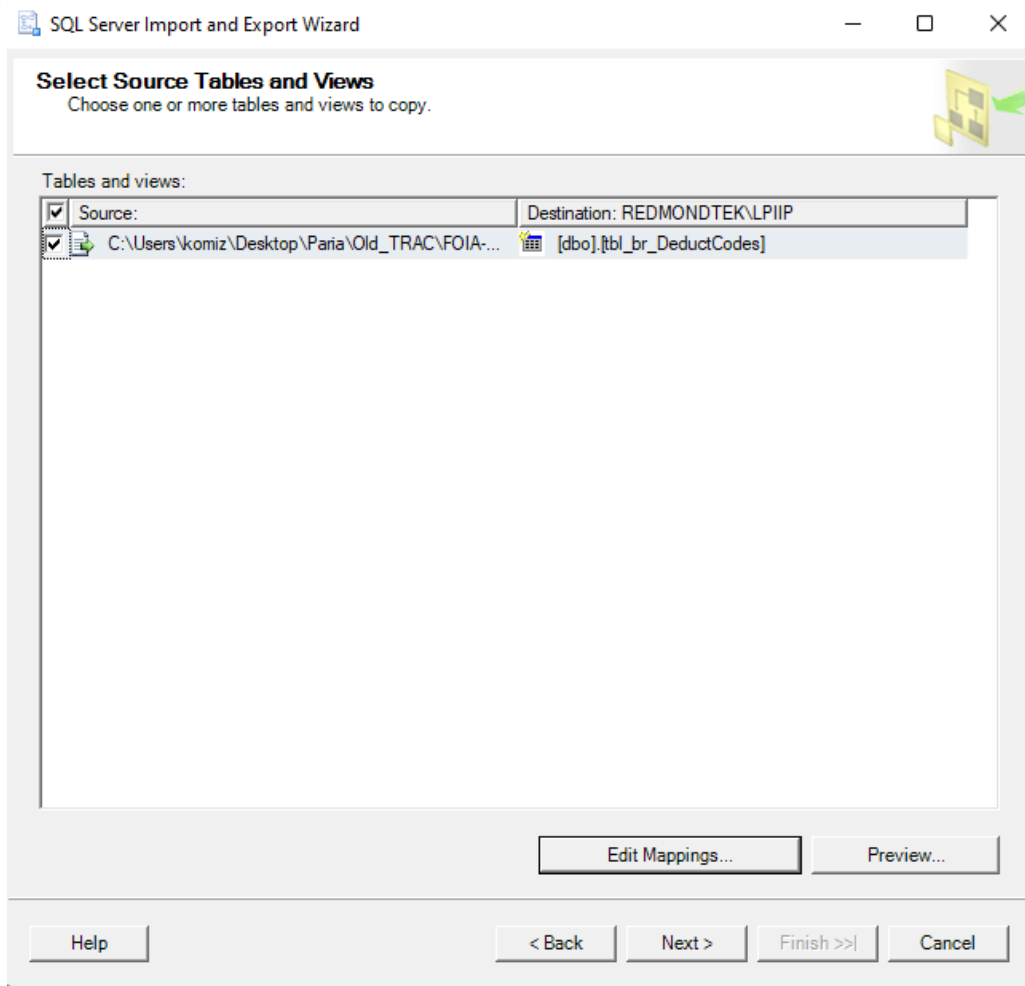
Password:

Database: <default> Refresh

New...

Help < Back Next > Finish >>| Cancel

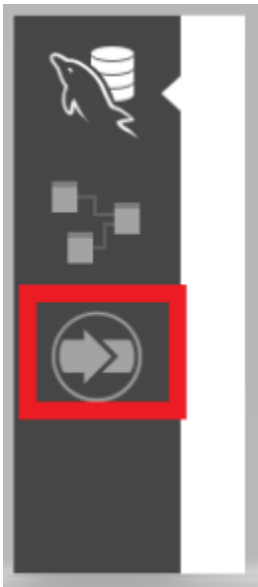
- Verify the Source, Destination, and Mappings then click **Next**.



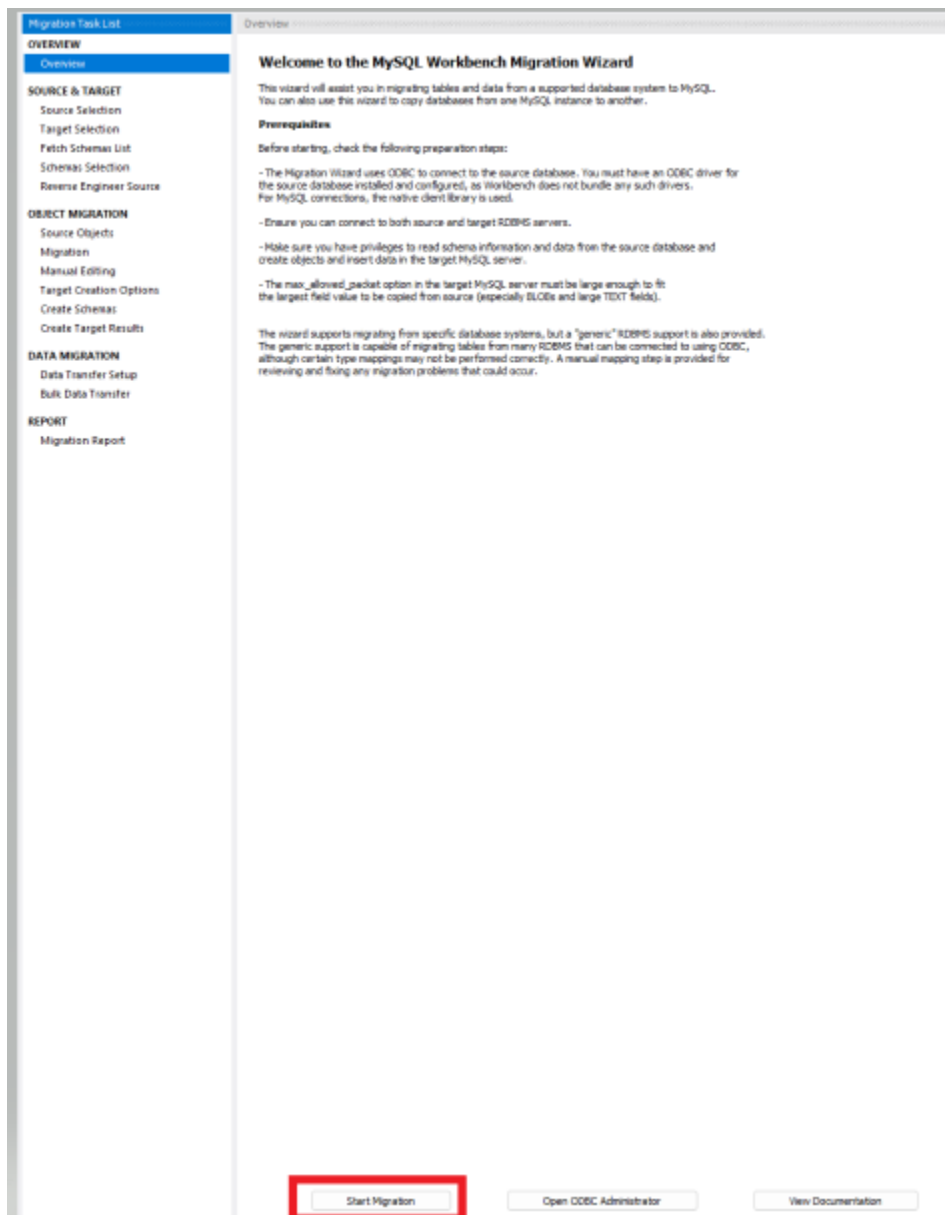
- Click **Finish** to complete the Import.
- Repeat all the steps above to import additional csv to your database.

### Migrate SQL to MySQL

- Launch MySQL Workbench.
- On the home page select Migration.



- Select Start Migration from the button.



- In the **Source Selection** section, Select **Microsoft SQL Server** for **Database System**.
- Leave the **Stored Connection** Blank.
- For **Connection Method**, Select **ODBC (native)** from dropdown.
- Configure the connection as follows:
  - Driver: SQL Server
  - Server: type your SQL server name here (e.x. localhost\SQLEXPRESS)
  - Username: type your SQL server username here (e.x. sa)
  - Password: add the password to your SQL server to the Vault.
  - Database: type the database that you want to migrate to MySQL.
- Click **Test Connection** at the bottom of the page.

#### Source RDBMS Connection Parameters

Database System:  Select a RDBMS from the list of supported systems

Stored Connection:  Select from saved connection settings

Connection Method:  Method to use to connect to the RDBMS

Parameters ☒ Advanced

Driver:	<input type="text" value="SQL Server"/>	The name of the ODBC driver you are using.
Server:	<input type="text" value="localhost\SQLEXPRESS"/>	Address\instance name of the server.
Username:	<input type="text" value="sa"/>	Name of the user to connect with.
Password:	<input type="button" value="Store in Vault ..."/> <input type="button" value="Clear"/>	The user's password. Leave blank to input when needed.
Database:	<input type="text"/>	The database to connect to. Leave blank to select it later.

- Once the connection is passed, Click **Next**.
- For Target Selection section, configure as follows:
  - Stored Connection: Select a connection to your MySQL server
  - Connection Method: Leave it to default (Standard (TCP/IP))
  - Hostname: Type your MySQL server name
  - Port: Leave it as default if you are using default port for your MySQL otherwise type the port.
  - Username: Type your MySQL username
  - Password: Store your MySQL password in Vault.
  - Default Schema: if you have already created your schema then type it here otherwise leave it blank to create it later.
- Click **Test Connection** at the bottom of the page.

#### Target RDBMS Connection Parameters

Stored Connection:  Select from saved connection settings

Connection Method:  Method to use to connect to the RDBMS

Parameters ☒ SSL ☐ Advanced

Hostname:	<input type="text" value="localhost"/>	Port:	<input type="text" value="3306"/>	Name or IP address of the server host - and TCP/IP port.
Username:	<input type="text" value="root"/>			Name of the user to connect with.
Password:	<input type="button" value="Store in Vault ..."/> <input type="button" value="Clear"/>			The user's password. Will be requested later if it's not set.
Default Schema:	<input type="text"/>			The schema to use as default schema. Leave blank to select it later.

Test Connection

- Once the connection is passed, Click **Next**.
- Click **Next** on **Fetch Schemas List** section.

The following tasks will now be performed. Please monitor the execution.

The names of available schemas will be retrieved from the source RDBMS. The account used for the connection will need to have appropriate privileges for listing and reading the schemas you want to migrate. Target RDBMS connection settings will also be checked for validity.

- ☒ Connect to source DBMS
- ☒ Check target DBMS connection
- ☒ Retrieve schema list from source

Finished performing tasks. Click [Next >] to continue.

- Select your source schema in the **Schema Selection** section. Then Select your Schema Name Mapping Method.

#### Schema Name Mapping Method

Choose how the reverse engineered schemas and objects should be mapped.

- ☐ Keep schemas as they are: Catalog.Schema.Table -> Schema.Table
- ☒ Only one schema: Catalog.Schema.Table -> Catalog.Table
- ☐ Only one schema, keep current schema names as a prefix: Catalog.Schema.Table -> Catalog.Schema\_Table

- Click **Next**.
- Click **Next** in **Reverse Engineer Source** section.



Selected schema metadata will now be fetched from the source RDBMS and reverse engineered so that its structure can be determined.

- ☒ Connect to source DBMS
- ☒ Reverse engineer selected schemas
- ☒ Post-processing of reverse engineered schemas

Finished performing tasks. Click [Next >] to continue.

- Verify the Source selection in **Source Objects** section then click **Next**.

You may select the objects to be migrated in the lists below.  
All tables will be migrated by default.

☒ Migrate Table objects  
4 total, 4 selected

Show Selection

Filter objects (wildcards chars \* and ? are allowed)

- Click **Next** on **Migration and Manual Editing** section.
- In the **Target Creation Options** section, select **Create schema in target RDBMS** then click **Next**.

Select options for the creation of the migrated schema in the target MySQL server and click [Next >] to execute.

Schema Creation

☒ Create schema in target RDBMS

- Click **Next** for the **Create Schemas** and **Create Target Results** section.
- Leave all configuration as default in the **Data Transfer Setup** section then click **Next** to start the migration.

Select options for the copy of the migrated schema tables in the target MySQL server and click [Next >] to execute.

**Data Copy**

☒ Online copy of table data to target RDBMS

☐ Create a batch file to copy the data at another time

Batch File:

You should edit this file to add the source and target server passwords before running it.

☐ Create a shell script to use native server dump and load abilities for fast migration

Bulk Data Copy Script:

Edit the generated file and change passwords at the top of the generated script.  
Run it on the source server to create a zip package containing a data dump as well as a load script.  
Copy this to the target server, extract it, and run the import script. See the script output for further details.

**Options**

☐ Truncate target tables (i.e. delete contents) before copying data

Worker tasks

☐ Enable debug output for table copy

☐ Driver sends data already encoded as UTF-8.

- A report of migration will be generated, once the migration is completed.