MySQL Create Table | How to Create Database in MySQL

Steps to Create Database in MySQL

Create Database in two ways

1) By executing a simple SQL query

2) By using forward engineering in MySQL Workbench

In this SQL Tutorial, you will learn-

- How to Create Database
- How to Create Table in MySQL
- Data types
- MySQL workbench ER diagram forward Engineering

As a SQL beginner, let's look into the query method first.

How to Create Database

Here is how to create a database in MySQL:

CREATE DATABASE is the SQL command used for creating a database in MySQL.

Imagine you need to create a database with the name "movies". You can create a database in MySQL by executing the following SQL command.

```
CREATE DATABASE movies;
```

Note: you can also use the command CREATE SCHEMA instead of CREATE DATABASE

Now let's improve our SQL query adding more parameters and specifications.

IF NOT EXISTS

A single MySQL server could have multiple databases. If you are not the only one accessing the same MySQL server or if you have to deal with multiple databases there is a probability of attempting to create a new database with the name of an existing database . **IF NOT EXISTS** let you instruct MySQL server to check the existence of a database with a similar name prior to creating the database.

When **IF NOT EXISTS** is used, the database is created only if the given name does not conflict with an existing database's name. Without the use of **IF NOT EXISTS** MySQL throws an error.

```
CREATE DATABASE IF NOT EXISTS movies;
```

Collation and Character Set

Collation is a set of **rules used in comparison.** Many people use MySQL to store data other than English. Data is stored in MySQL using a specific character set. The character set can be defined at different levels viz, server , database , table and columns.

You need to select the rules of collation which in turn depend on the character set chosen.

For instance, the Latin1 character set uses the `latin1_swedish_ci` collation which is the Swedish case insensitive order.

```
CREATE DATABASE IF NOT EXISTS movies CHARACTER SET latin1
COLLATE latin1_swedish_ci
```

The best practice while using local languages like Arabic , Chinese etc is to select a Unicode (utf-8) character set which has several collations or just stick to the default collation utf8-general-ci.

You can find the list of all collations and character sets.

You can see a list of existing databases by running the following SQL command.

```
SHOW DATABASES
```

How to Create Table in MySQL

CREATE TABLE command is used to create tables in a database

Tables can be created using the CREATE **TABLE** statement and it actually has the following syntax.

```
CREATE  TABLE [IF NOT EXISTS] `TableName` (`fieldname`
dataType [optional parameters]) ENGINE = storage Engine;
```

HERE

- "CREATE TABLE" is the one responsible for the creation of the table in the database.
- "[IF NOT EXISTS]" is optional and only creates the table if no matching table name is found.

- "`fieldName`" is the name of the field and "data Type" defines the nature of the data to be stored in the field.
- "[optional parameters]" additional information about a field such as " AUTO_INCREMENT" , NOT NULL etc.

MySQL Create Table Example

Below is a MySQL example to create a table in database:

```
CREATE  TABLE IF NOT EXISTS `MyFlixDB`.`Members` (
  `membership_number` INT  AUTOINCREMENT ,
  `full_names` VARCHAR(150) NOT NULL ,
  `gender` VARCHAR(6) ,
  `date_of_birth` DATE ,
  `physical_address` VARCHAR(255) ,
  `postal_address` VARCHAR(255) ,
  `contact_number` VARCHAR(75) ,
  `email` VARCHAR(255) ,
  PRIMARY KEY (`membership_number`) )
ENGINE = InnoDB;
```

Now let's see what the MySQL's data types are. You can use any of them depending on your need. You should always try to not to underestimate or overestimate potential range of data when creating a database.

DATA TYPES

Data types define the nature of the data that can be stored in a particular column of a table

MySQL has **3** main categories of data types namely

1. Numeric,
2. Text
3. Date/time.

Numeric Data types

Numeric data types are used to store numeric values. It is very important to make sure the range of your data is between lower and upper boundaries of numeric data types.

TINYINT( )	-128 to 127 normal 0 to 255 UNSIGNED.
SMALLINT ( )	-32768 to 32767 normal 0 to 65535 UNSIGNED.
MEDIUMINT( )	-8388608 to 8388607 normal 0 to 16777215 UNSIGNED.
INT( )	-2147483648 to 2147483647 normal 0 to 4294967295 UNSIGNED.
BIGINT( )	-9223372036854775808 to 9223372036854775807 normal 0 to 18446744073709551615 UNSIGNED.
FLOAT	A small approximate number with a floating decimal point.
DOUBLE(,)	A large number with a floating decimal point.
DECIMAL(,)	A DOUBLE stored as a string , allowing for a fixed decimal point. Choice for storing currency values.

Text Data Types

As data type category name implies these are used to store text values. Always make sure the length of your textual data does not exceed maximum lengths.

CHAR( )	A fixed section from 0 to 255 characters long.
VARCHAR( )	A variable section from 0 to 255 characters long.
TINYTEXT	A string with a maximum length of 255 characters.
TEXT	A string with a maximum length of 65535 characters.
BLOB	A string with a maximum length of 65535 characters.
MEDIUMTEXT	A string with a maximum length of 16777215 characters.
MEDIUMBLOB	A string with a maximum length of 16777215 characters.
LONGTEXT	A string with a maximum length of 4294967295 characters.
LONGBLOB	A string with a maximum length of 4294967295 characters.

Date / Time

DATE	YYYY-MM-DD
DATETIME	YYYY-MM-DD HH:MM:SS
TIMESTAMP	YYYYMMDDHHMMSS
TIME	HH:MM:SS

Apart from above there are some other data types in MySQL.

ENUM	To store text value chosen from a list of predefined text values
SET	This is also used for storing text values chosen from a list of predefined text values. It can have multiple values.
BOOL	Synonym for TINYINT(1), used to store Boolean values
BINARY	Similar to CHAR, the difference is texts are stored in binary format.
VARBINARY	Similar to VARCHAR, the difference is that texts are stored in binary format.

Now let's see a query for creating a table which has data of all data types. Study it and identify how each data type is defined in the below create table MySQL example.

```
CREATE TABLE`all_data_types` (
    `varchar` VARCHAR( 20 )  ,
    `tinyint` TINYINT  ,
    `text` TEXT  ,
    `date` DATE  ,
    `smallint` SMALLINT  ,
    `mediumint` MEDIUMINT  ,
    `int` INT  ,
    `bigint` BIGINT  ,
    `float` FLOAT( 10, 2 )  ,
    `double` DOUBLE  ,
    `decimal` DECIMAL( 10, 2 )  ,
    `datetime` DATETIME  ,
    `timestamp` TIMESTAMP  ,
    `time` TIME  ,
    `year` YEAR  ,
    `char` CHAR( 10 )  ,
    `tinyblob` TINYBLOB  ,
    `tinytext` TINYTEXT  ,
    `blob` BLOB  ,
    `mediumblob` MEDIUMBLOB  ,
    `mediumtext` MEDIUMTEXT  ,
    `longblob` LONGBLOB  ,
    `longtext` LONGTEXT  ,
    `enum` ENUM( '1', '2', '3' )  ,
    `set` SET( '1', '2', '3' )  ,
    `bool` BOOL  ,
    `binary` BINARY( 20 )  ,
    `varbinary` VARBINARY( 20 )
) ENGINE= MYISAM ;
```

Best practices

- Use upper case letters for SQL keywords i.e. "DROP SCHEMA IF EXISTS `MyFlixDB`;"
- End all your SQL commands using semicolons.

- Avoid using spaces in schema, table and field names. Use underscores instead to separate schema, table or field names.

MySQL workbench ER diagram forward engineering

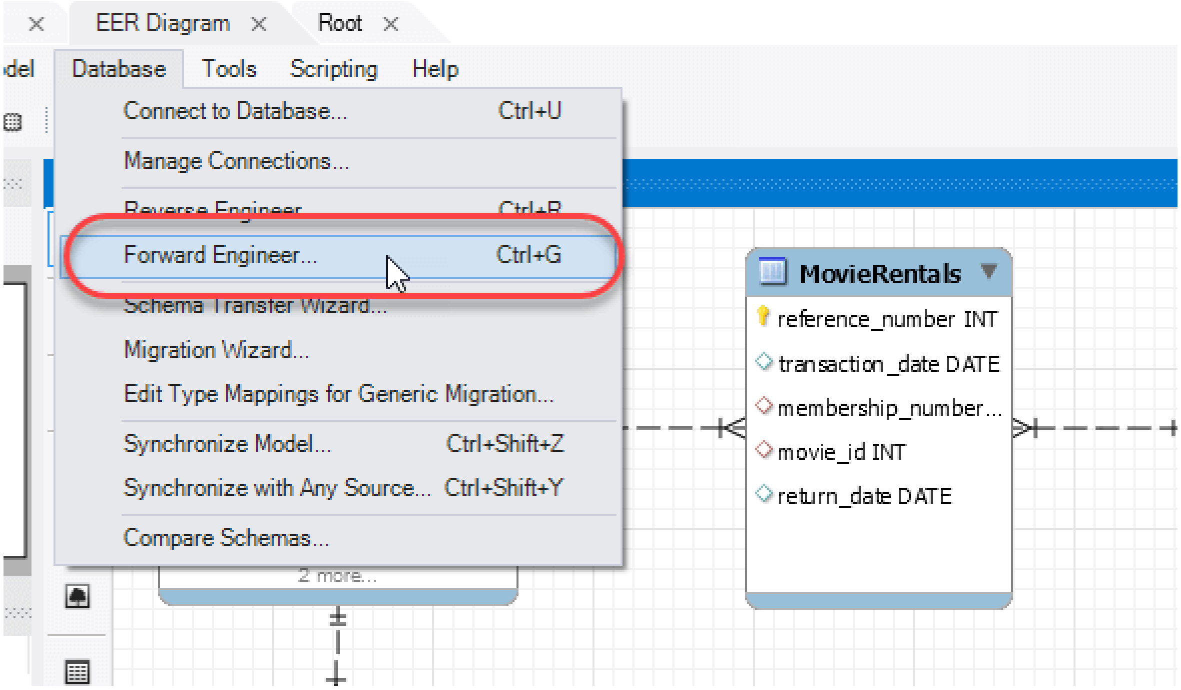
MySQL workbench has utilities that support forward engineering. **Forward engineering** is a technical term to describe the process of **translating a logical model into a physical implement automatically**.

We created an ER diagram on our ER modeling tutorial. We will now use that ER model to generate the SQL scripts that will create our database.

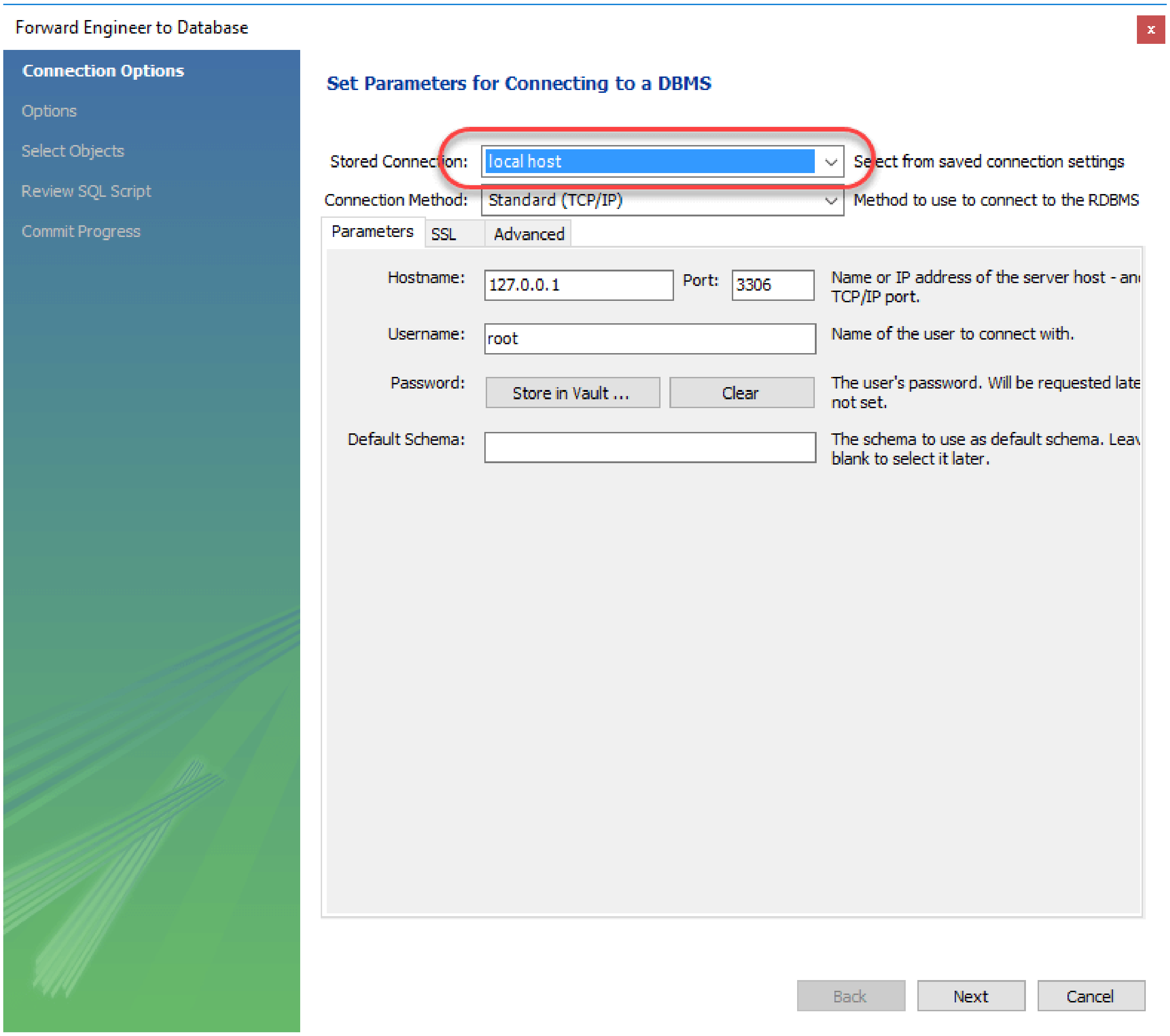
Creating the MyFlix database from the MyFlix ER model

1. Open the ER model of MyFlix database that you created in an earlier tutorial.

2. Click on the database menu. Select forward engineer


EER Diagram
Root
del
Database
Tools
Scripting
Help
Connect to Database... Ctrl+U
Manage Connections...
Reverse Engineer... Ctrl+R
Forward Engineer... Ctrl+G
Schema Transfer Wizard...
Migration Wizard...
Edit Type Mappings for Generic Migration...
Synchronize Model... Ctrl+Shift+Z
Synchronize with Any Source... Ctrl+Shift+Y
Compare Schemas...
2 more...
MovieRentals
reference_number INT
transaction_date DATE
membership_number...
movie_id INT
return_date DATE

3. The next window allows you to connect to an instance of MySQL server. Click on the stored connection drop down list and select the local host. Click Execute



Forward Engineer to Database

Set Parameters for Connecting to a DBMS

Stored Connection: local host — Select from saved connection settings

Connection Method: Standard (TCP/IP) — Method to use to connect to the RDBMS

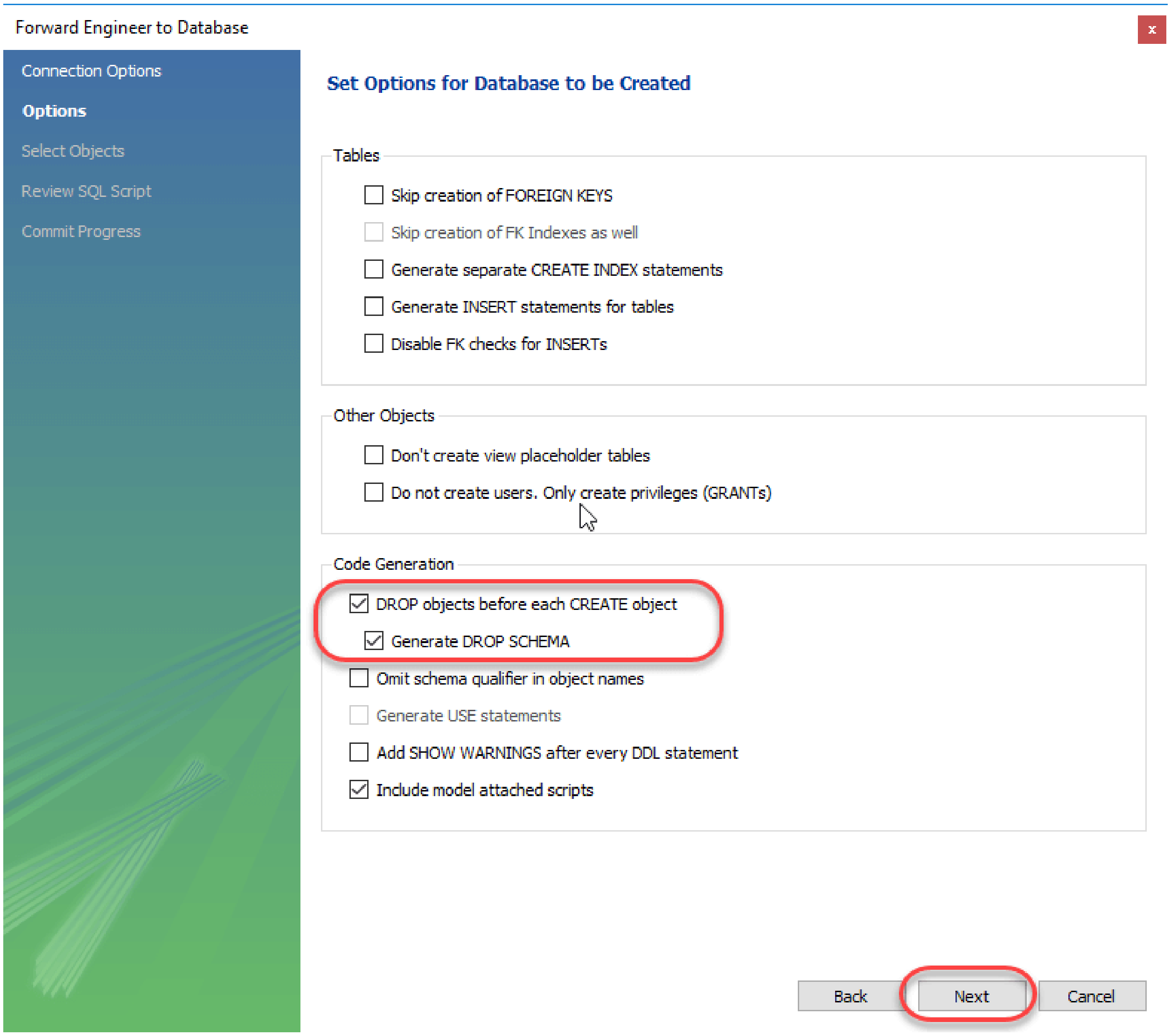
Hostname: 127.0.0.1 Port: 3306 — Name or IP address of the server host - an TCP/IP port.

Username: root — Name of the user to connect with.

Password: — The user's password. Will be requested late not set.

Default Schema: — The schema to use as default schema. Leav blank to select it later.

4. Select the options shown below in the wizard that appears. Click next



Set Options for Database to be Created

Tables

□ Skip creation of FOREIGN KEYS
□ Skip creation of FK Indexes as well
□ Generate separate CREATE INDEX statements
□ Generate INSERT statements for tables
□ Disable FK checks for INSERTs

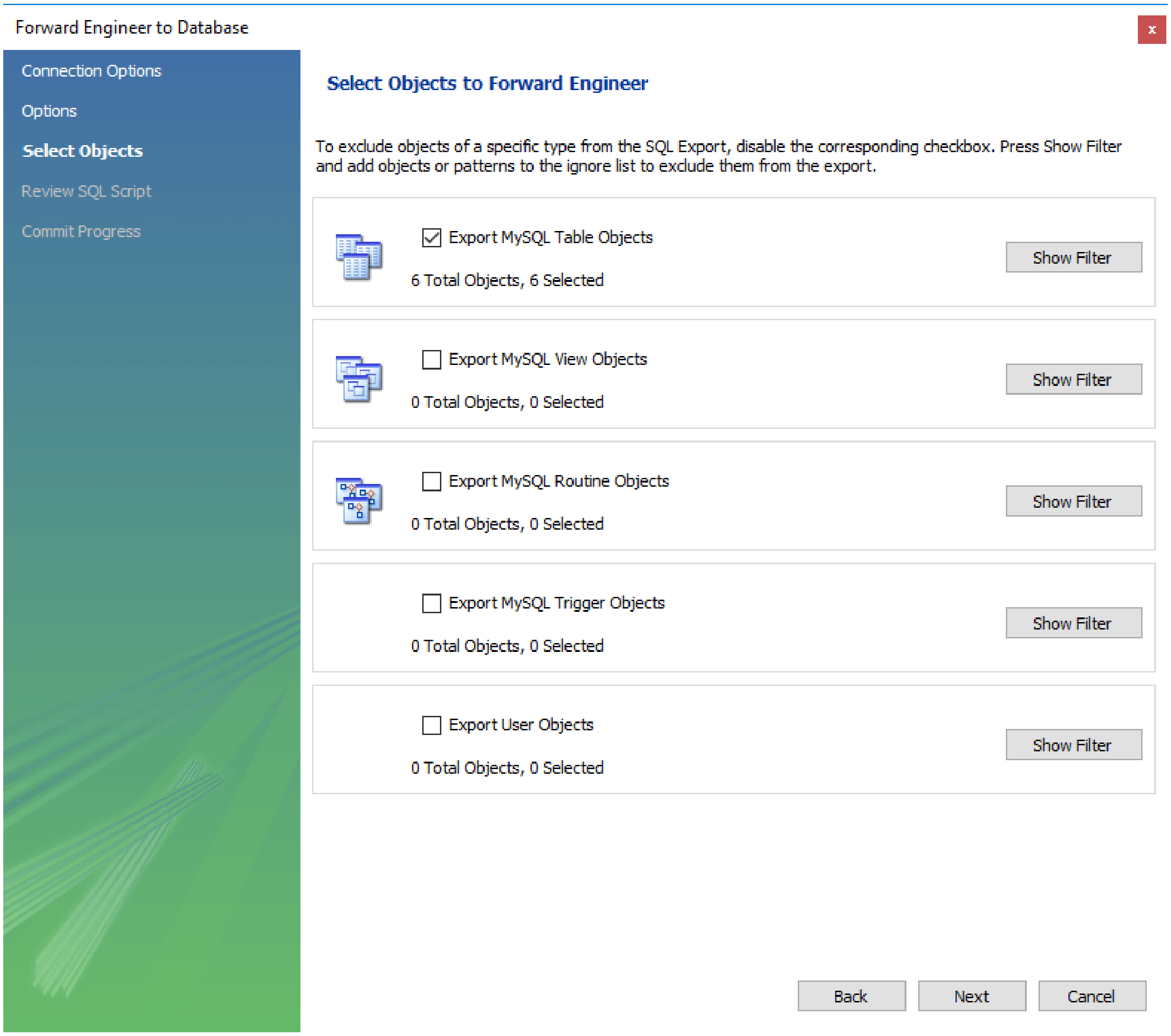
Other Objects

□ Don't create view placeholder tables
□ Do not create users. Only create privileges (GRANTs)

Code Generation

☑ DROP objects before each CREATE object
☑ Generate DROP SCHEMA
□ Omit schema qualifier in object names
□ Generate USE statements
□ Add SHOW WARNINGS after every DDL statement
☑ Include model attached scripts

5. The next screen shows the summary of objects in our EER diagram. Our MyFlix DB has 5 tables. Keep the selection default and click Next.



Forward Engineer to Database

Select Objects to Forward Engineer

To exclude objects of a specific type from the SQL Export, disable the corresponding checkbox. Press Show Filter and add objects or patterns to the ignore list to exclude them from the export.

☑ Export MySQL Table Objects
6 Total Objects, 6 Selected

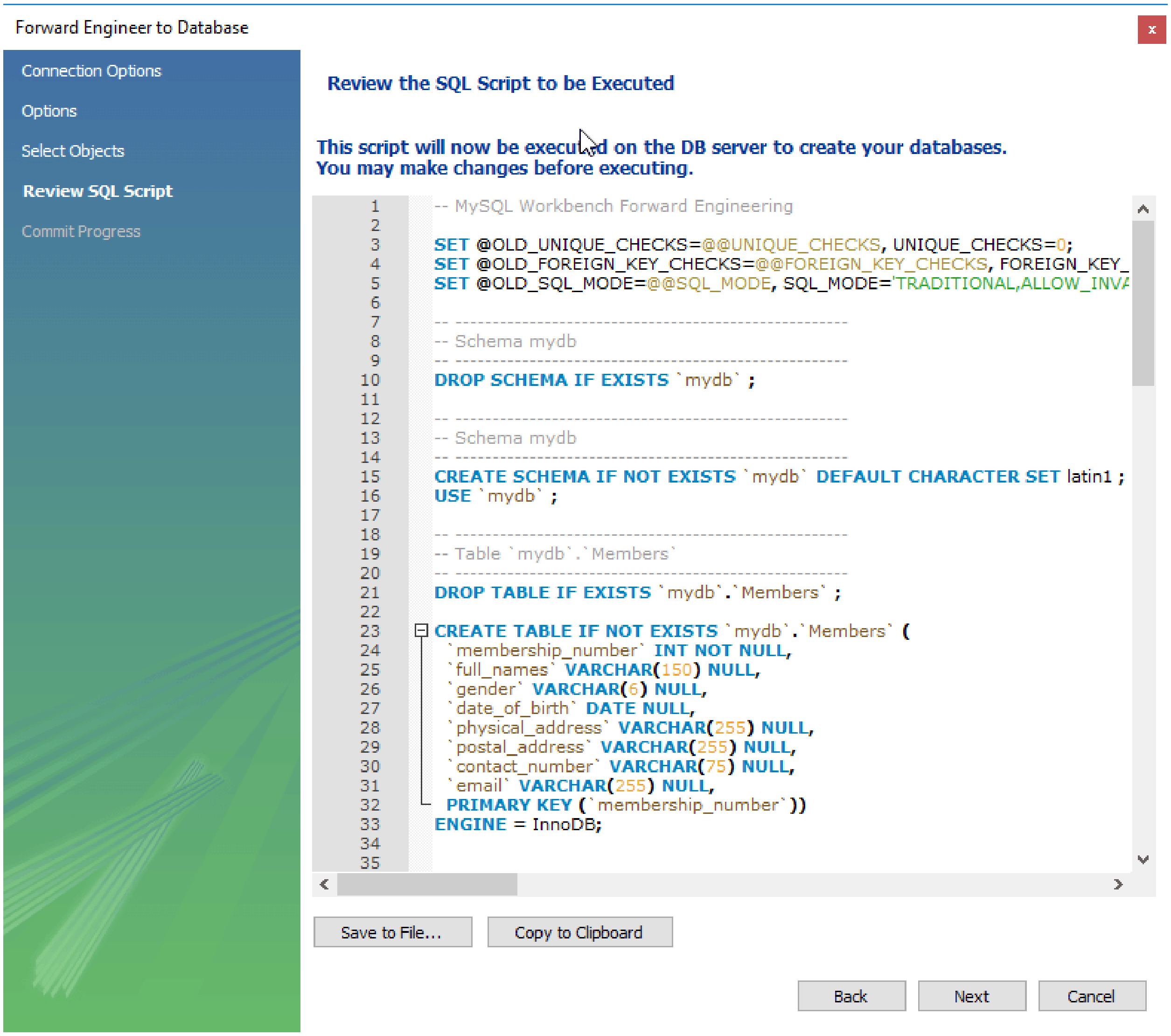
☐ Export MySQL View Objects
0 Total Objects, 0 Selected

☐ Export MySQL Routine Objects
0 Total Objects, 0 Selected

☐ Export MySQL Trigger Objects
0 Total Objects, 0 Selected

☐ Export User Objects
0 Total Objects, 0 Selected

6.. The window shown below appears. This window allows you to preview the SQL script to create our database. We can save the scripts to a *.sql" file or copy the scripts to the clipboard. Click on next button

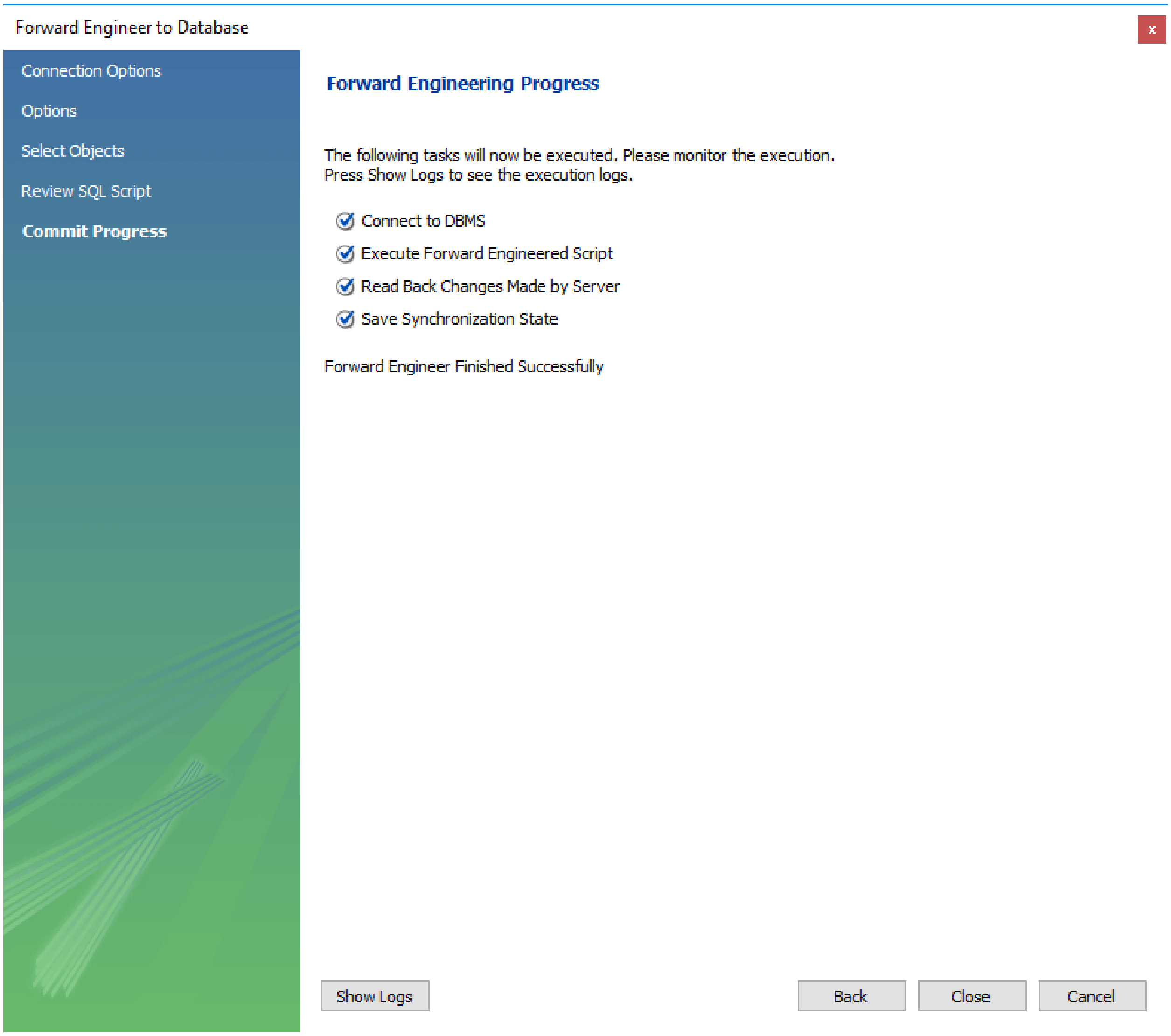


Review the SQL Script to be Executed

This script will now be executed on the DB server to create your databases.
You may make changes before executing.

```
-- MySQL Workbench Forward Engineering

SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_
SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='TRADITIONAL,ALLOW_INVA

-- -----------------------------------------------------
-- Schema mydb
-- -----------------------------------------------------
DROP SCHEMA IF EXISTS `mydb` ;

-- -----------------------------------------------------
-- Schema mydb
-- -----------------------------------------------------
CREATE SCHEMA IF NOT EXISTS `mydb` DEFAULT CHARACTER SET latin1 ;
USE `mydb` ;

-- -----------------------------------------------------
-- Table `mydb`.`Members`
-- -----------------------------------------------------
DROP TABLE IF EXISTS `mydb`.`Members` ;

CREATE TABLE IF NOT EXISTS `mydb`.`Members` (
  `membership_number` INT NOT NULL,
  `full_names` VARCHAR(150) NULL,
  `gender` VARCHAR(6) NULL,
  `date_of_birth` DATE NULL,
  `physical_address` VARCHAR(255) NULL,
  `postal_address` VARCHAR(255) NULL,
  `contact_number` VARCHAR(75) NULL,
  `email` VARCHAR(255) NULL,
  PRIMARY KEY (`membership_number`))
ENGINE = InnoDB;


```

7. The window shown below appears after successfully creating the database on the selected MySQL server instance.



Forward Engineering Progress

The following tasks will now be executed. Please monitor the execution.
Press Show Logs to see the execution logs.

- ☑ Connect to DBMS
- ☑ Execute Forward Engineered Script
- ☑ Read Back Changes Made by Server
- ☑ Save Synchronization State

Forward Engineer Finished Successfully

Summary

- Creating a database involves translating the logical database design model into the physical database.
- MySQL supports a number of data types for numeric, dates and strings values.
- CREATE DATABASE command is used to create a database
- CREATE TABLE command is used to create tables in a database

- MySQL workbench supports forward engineering which involves automatically generating SQL scripts from the logical database model that can be executed to create the physical database

The Database along with Dummy Data is attached. We will be using this DB for all our further tutorials. Simple import the DB in MySQL Workbench to get started