

## MySQL and MySQL Workbench Lab

The goal of this lab is creating a database from an ERD automatically. First, MySQL should be ready and it will be set as follows:

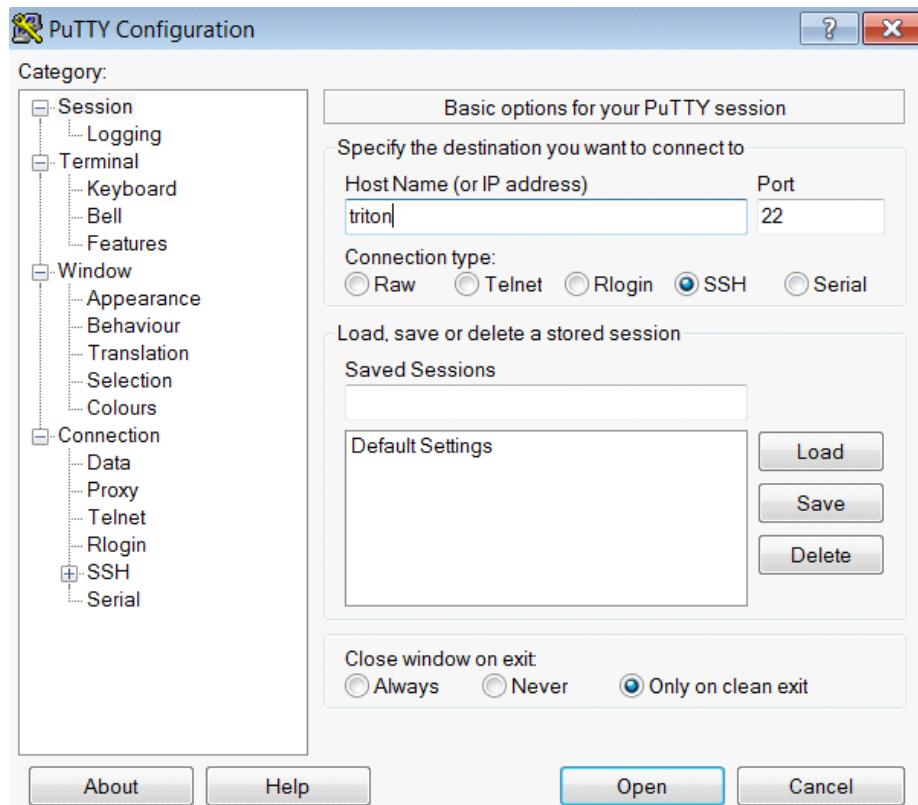
- MySQL preparation

Let's start putty program to connect to the triton server.

(start computer Science pgms ssh putty)

Host name: triton.towson.edu

Connect



You can login to the triton server by using your tiger user name and password.

Mac terminal

```
shell  
New remote connection  
ssh -p 22 user-name@triton.towson.edu
```

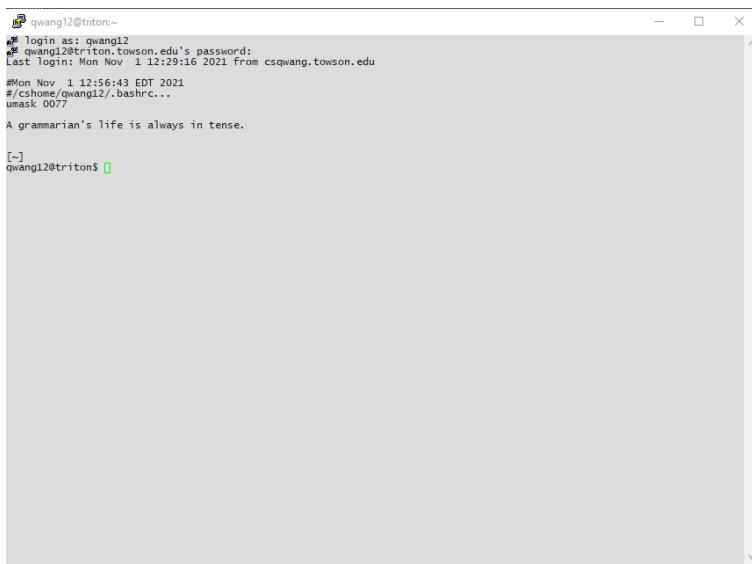
- Check your account status.

If you do not see your login name as a part of the prompt, you need to initialize your account.

([bsh\\$>/usr/cis/bin/getbashrc](#) )

Then type following for MySQL setting:

([/usr/cis/bin/getmycnf](#))



The screenshot shows a terminal window with the following text:  
qwang12@triton:~  
Login as: qwang12  
qwang12@triton.towson.edu's password:  
Last login: Mon Nov 1 12:29:16 2021 from csqwang.towson.edu  
#Mon Nov 1 12:56:43 EDT 2021  
#/cshome/qwang12/.bashrc...  
umask 0077  
A grammarian's life is always in tense.  
[~]  
qwang12@triton\$

- Preparation for MySQL

Please type following code:

[/usr/cis/etc/getbashrc](#)

Then type following code:

**qwang12@triton\$** [/usr/cis/etc/getmycnf](#)

Note: **qwang12@trition\$** is my prompt.

If you see bsh\$> then please let me know.

If it does not work then do following:

```
cp /usr/cis/bin/getmycnf ${HOME}
```

Then edit getmycnf file. (nano getmycnf)

```
cp /usr/cis/etc/tritondb.cnf ${HOME}/.my.cnf
```

Now, you can run getmycnf script

```
./getmycnf
```

- Start MySQL

```
qwang12@triton$ mysql -u username -p
```

insert the MySQL password (that is distributed at beginning of this class).

If everything is o.k. then you will see **MySQL>** prompt.

- Create a database

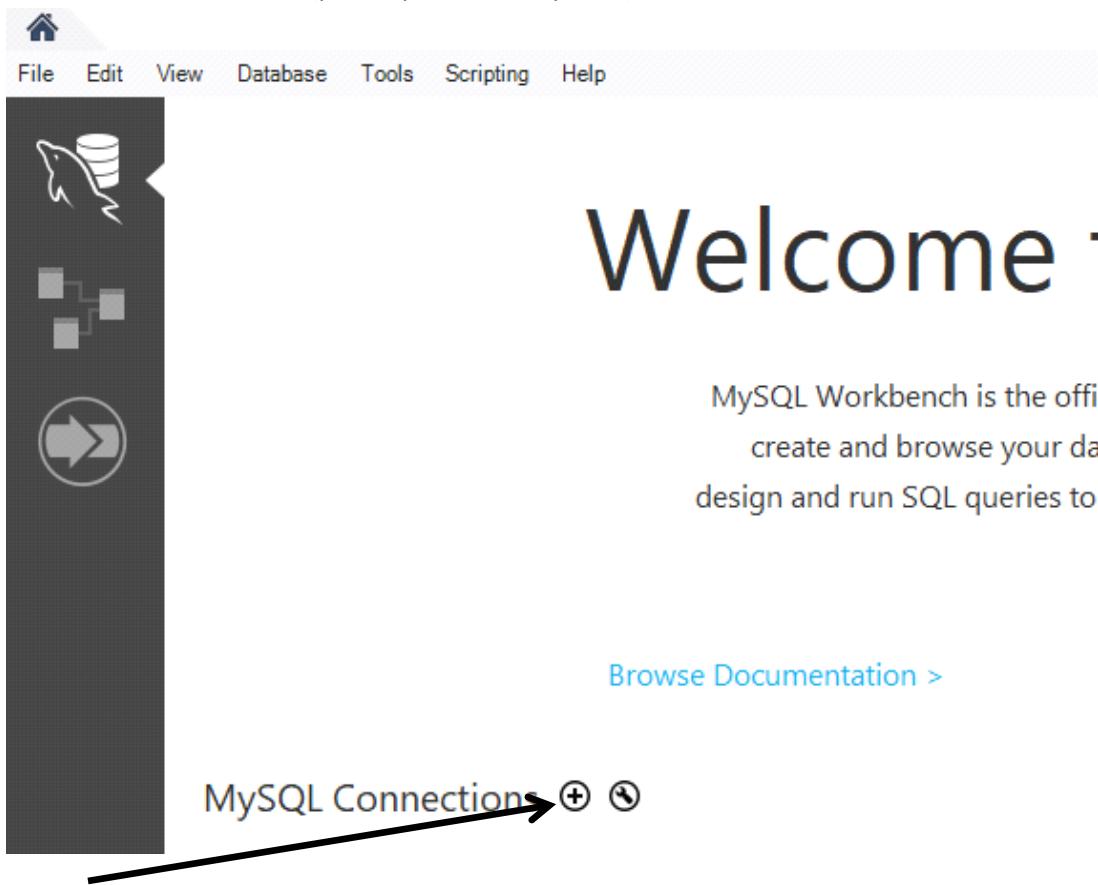
```
MySQL> create database databaseName; //you can find this databaseName in the distributed slip.
```

```
use databaseName;
```

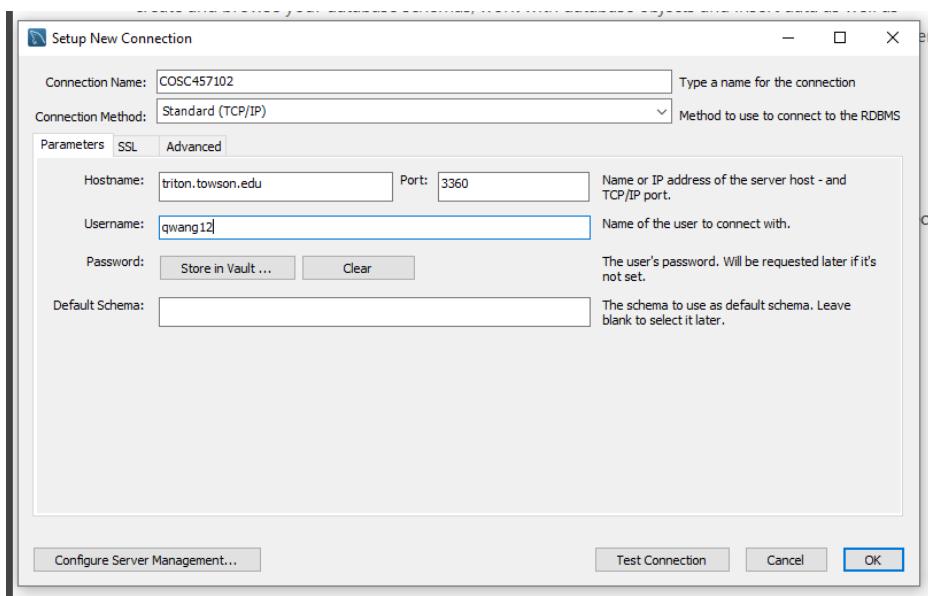
Now, let's start MySQL work bench.

- Open MySQL Work bench

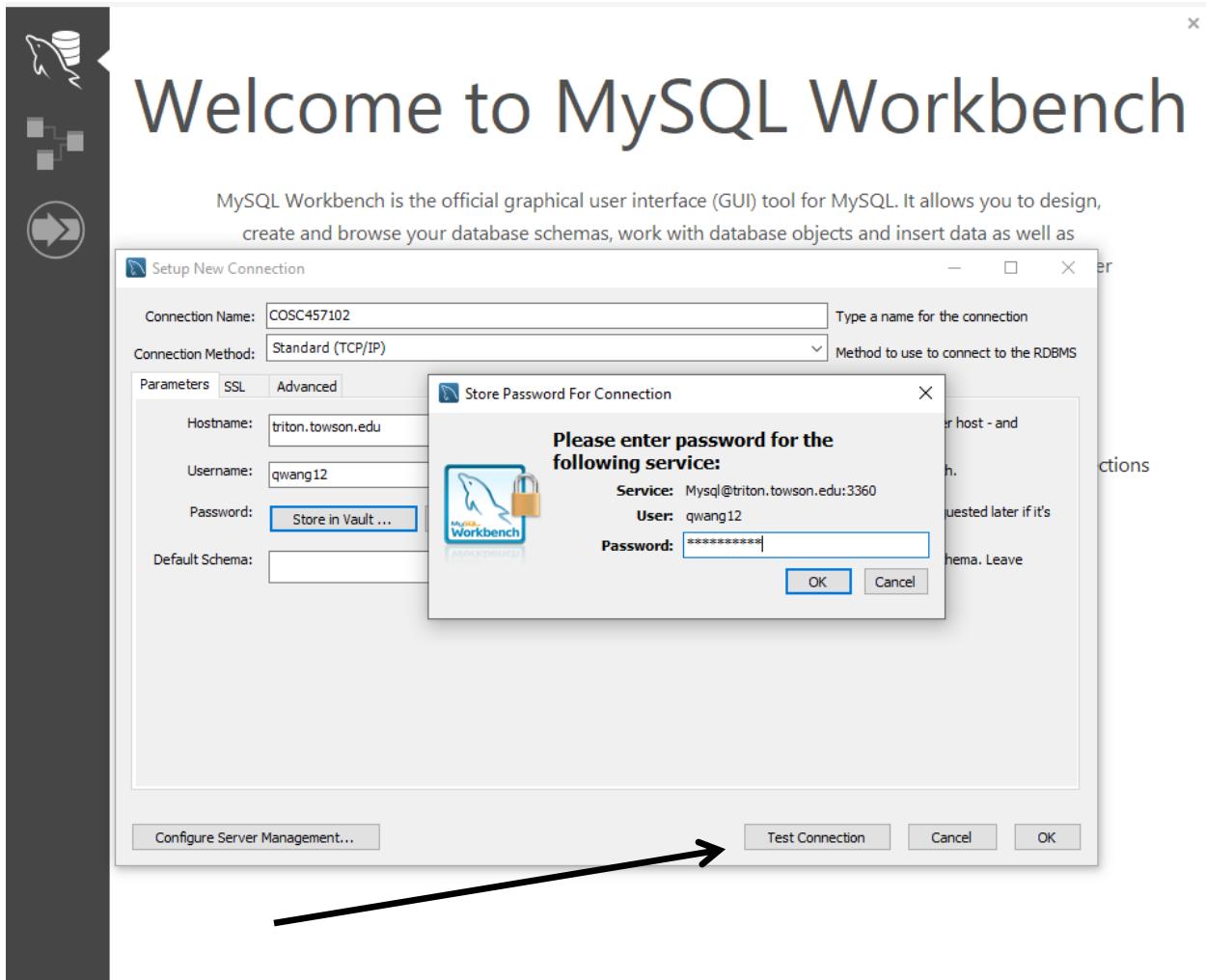
- Create a new connection (please press the + symbol)



Press this button.



Fill the boxes and put your Database password to the “Store in Vault”



Connection Name: *connection-name* (You can name this connection.)

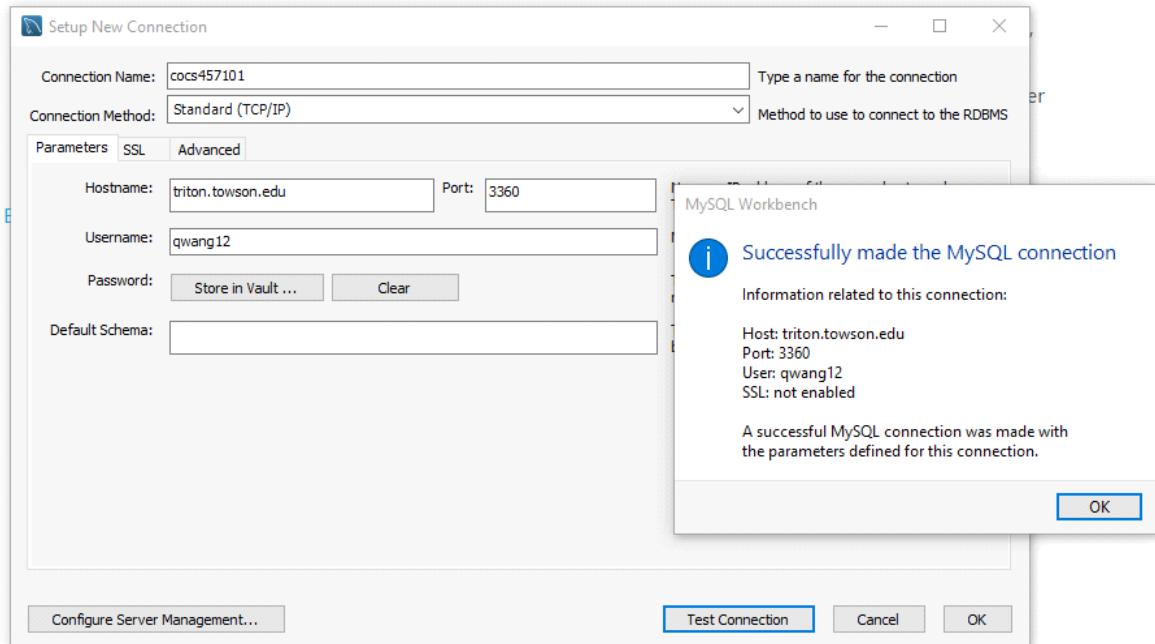
Host name: triton.towson.edu

Port: 3360

User name: *user name*

Password: *password* (Database Password)

- Store your password in a vault.

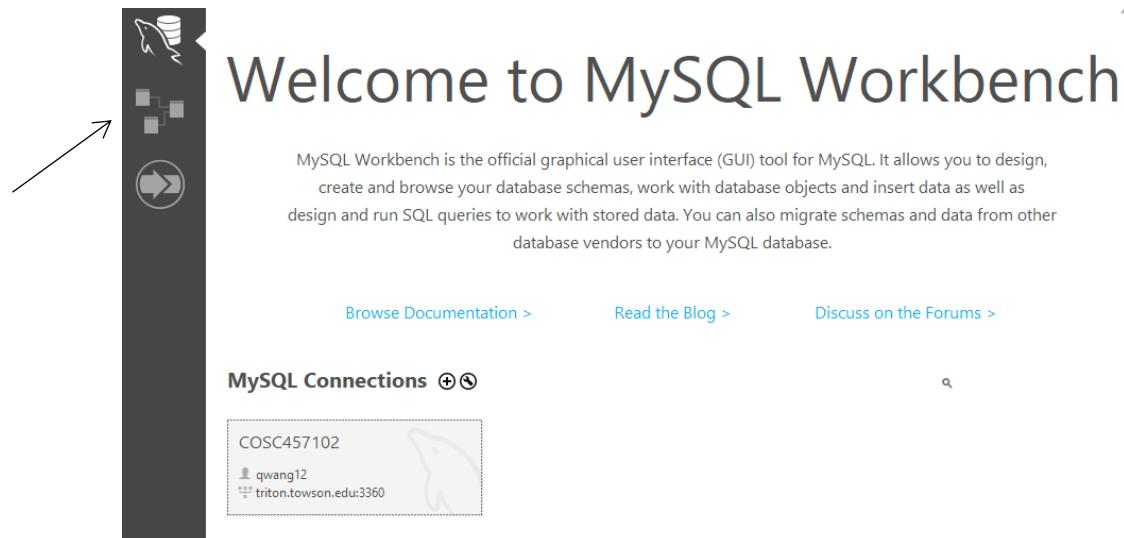


## 8.2 Test Connection (if it says o.k. then precede to ok)

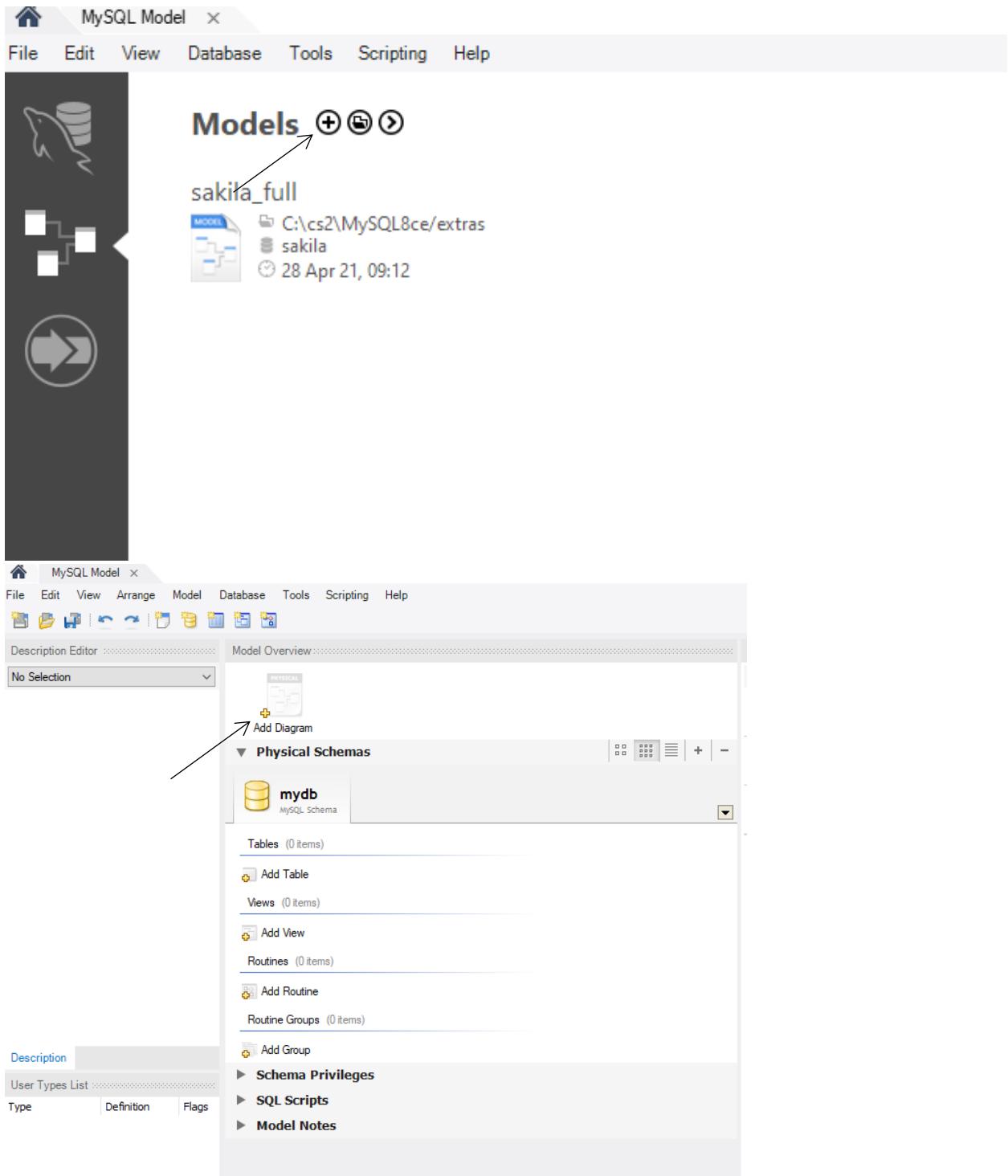
- Design an EERD

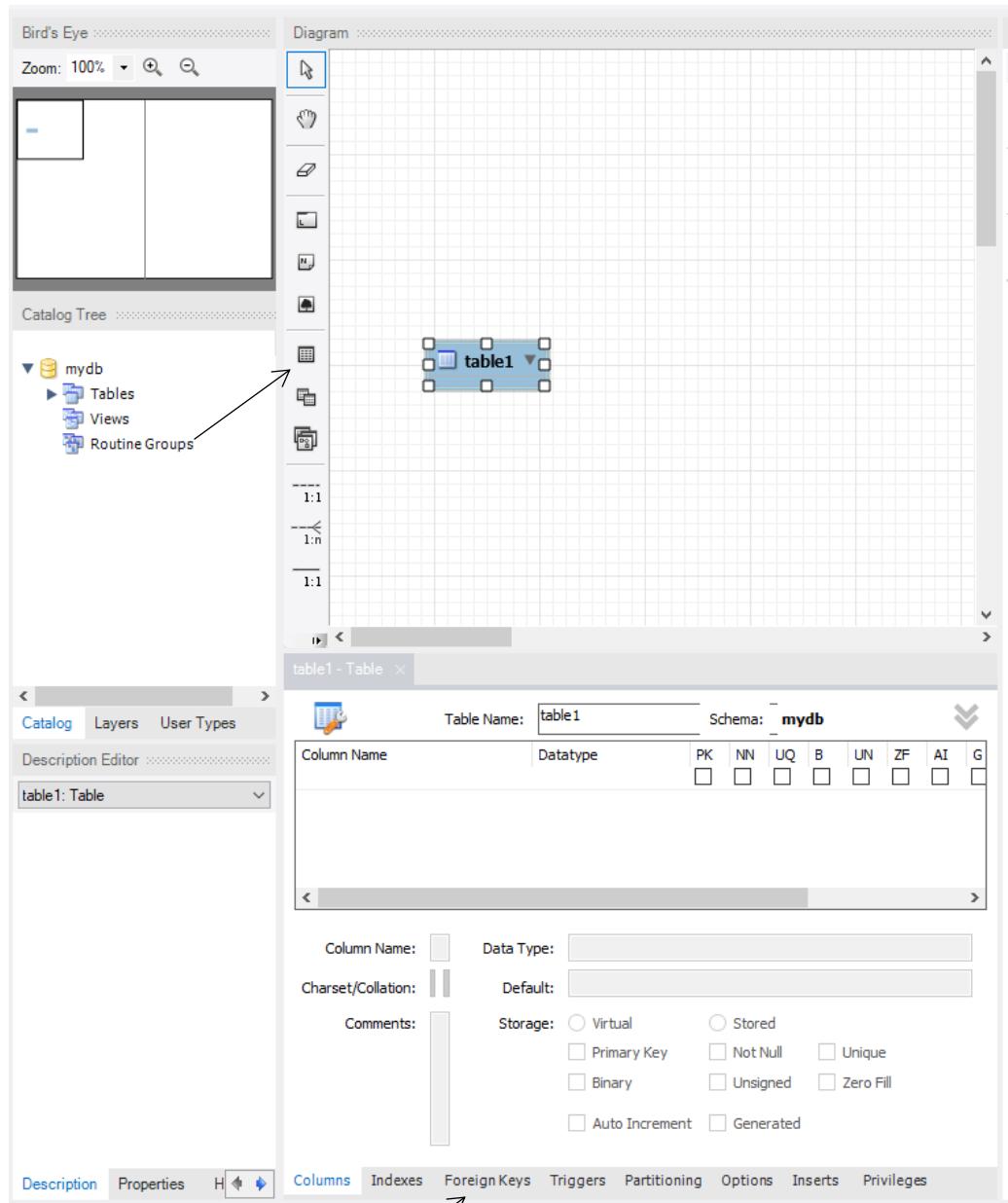
Please start an EERD with an existing database.

Press the “ERD symbol “ and then “Create EER model from a database” .



Then you will see a design screen. Using click and click (no dragging), create a table. Double click the table. Then insert attributes with type and length. Set up a primary key.





- Setup a foreign key

Using the foreign key tab (lower part of the screen) set up the foreign key.

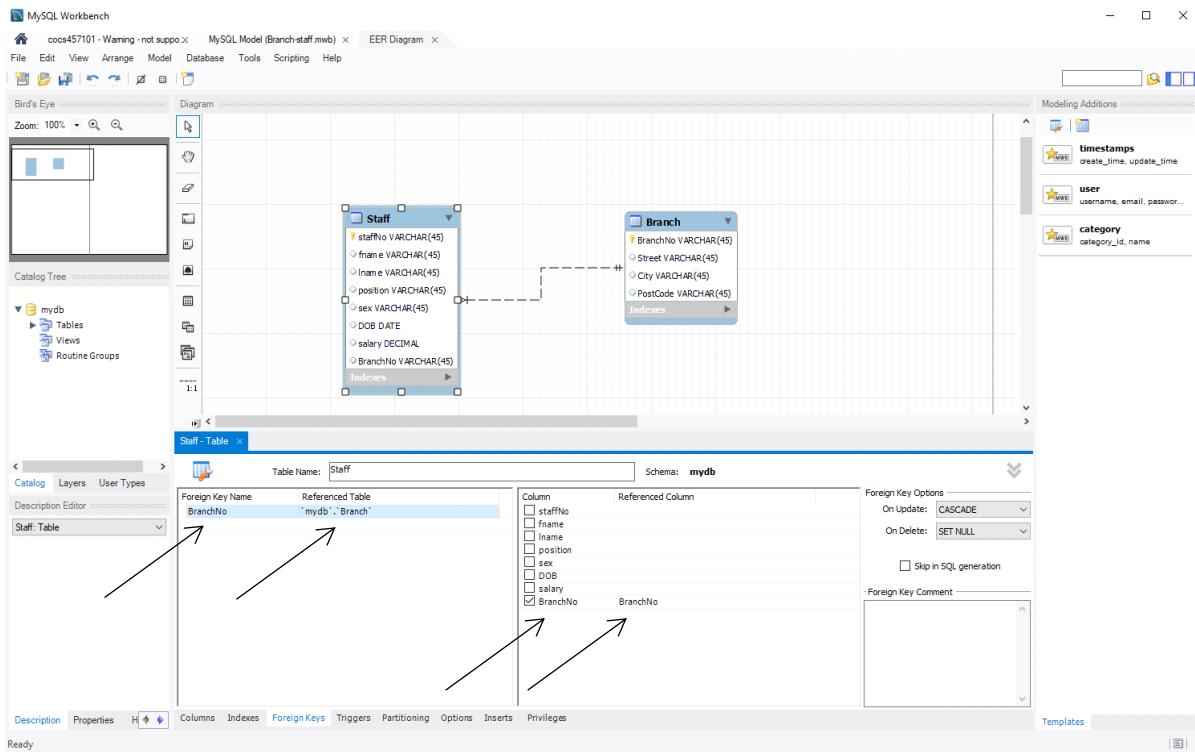
Branch			
branchNo	street	city	postcode
B002	56 Clover Dr	London	NW10 6EU
B003	163 Main St	Glasgow	G11 9QX
B004	32 Manse Rd	Bristol	BS99 1NZ
B005	22 Deer Rd	London	SW1 4EH

B007	16 Argyll St	Aberdeen	AB2 3SU
------	--------------	----------	---------

Staff							
staffNo	fName	IName	position	sex	DOB	salary	branchNO
SA9	Mary	Howe	Assistant	F	2/19/1970	9000	B007
SG14	David	Ford	Supervisor	M	3/24/1958	18000	B003
SG37	Ann	Beech	Assistant	F	11/10/1960	12000	B003
SG5	Susan	Brand	Manager	F	6/3/1940	24000	B003
SL21	John	White	Manager	M	10/1/1945	30000	B005
SL41	Julie	Lee	Assistant	F	6/13/1965	9000	B005

Select Staff entity. Then select the foreign key tab.

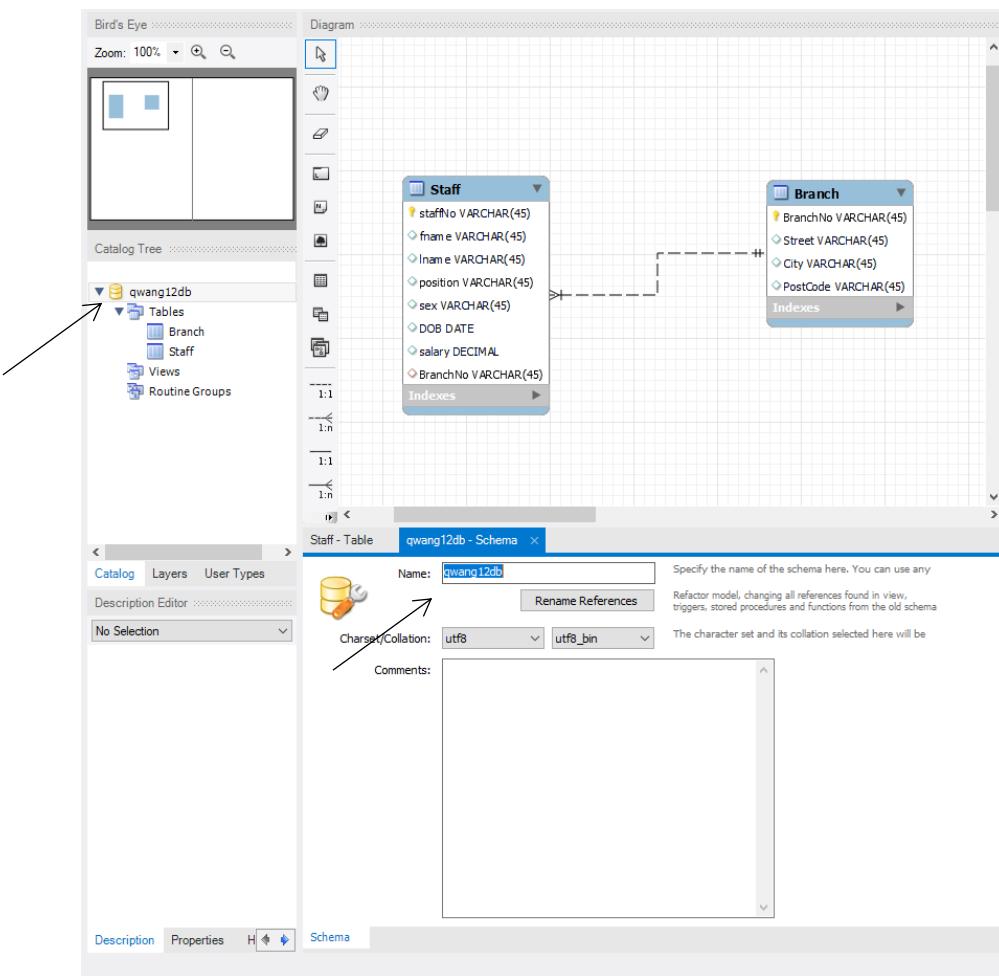
The screenshot shows the Oracle SQL Developer Data Modeler interface. On the left, the Catalog Tree shows a database named 'mydb' containing 'Tables', 'Views', and 'Routine Groups'. In the center, the 'Diagram' pane displays two entities: 'Staff' and 'Branch'. The 'Staff' entity has attributes: staffNo (PK), fName, Iname, position, sex, DOB, salary, and branchNo. It has a 1:n relationship with the 'Branch' entity, indicated by a dashed line connecting them. The 'Branch' entity has attributes: BranchNo (PK), Street, City, and PostCode. Below the diagram, the 'Staff - Table' properties are shown. The 'Description' tab is selected, displaying the table name 'Staff', schema 'mydb', and column definitions. The 'Columns' tab is also visible. At the bottom, the navigation tabs include 'Description', 'Properties', 'H', 'Columns', 'Indexes', 'Foreign Keys', 'Triggers', 'Partitioning', 'Options', 'Inserts', and 'Privileges'. A callout arrow points from the 'Description' tab in the Staff table's properties to the 'Foreign Keys' tab in the bottom navigation bar.

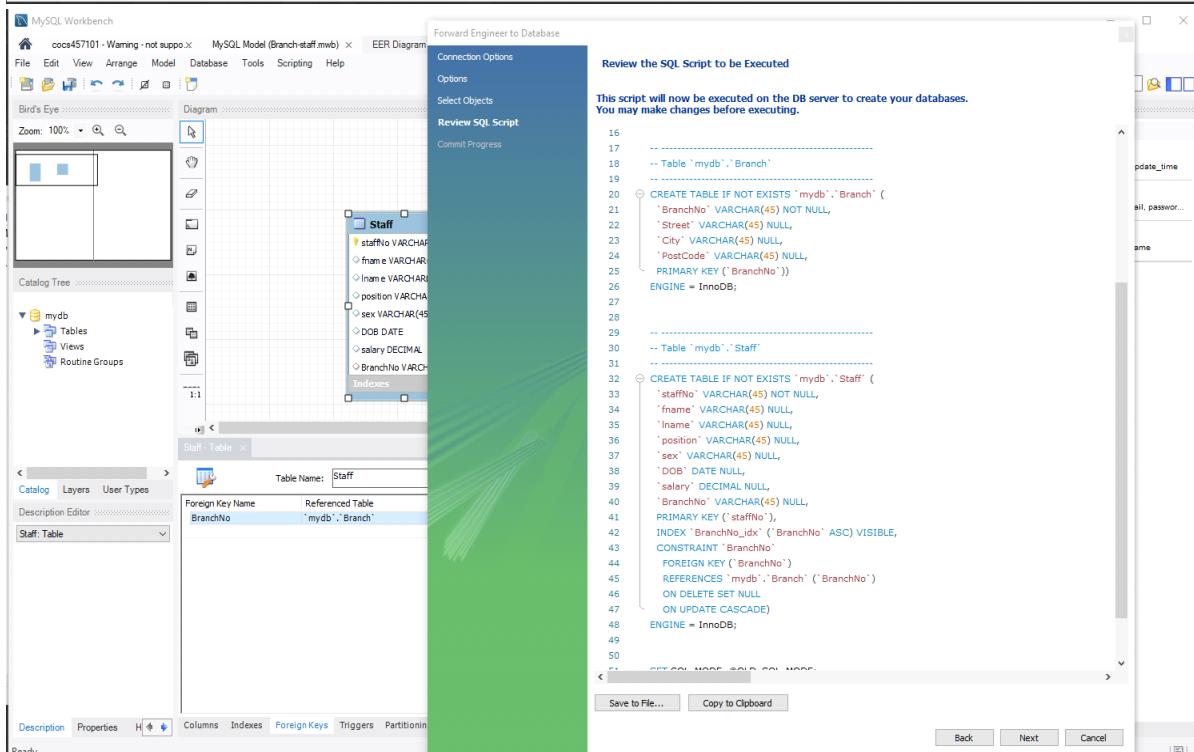
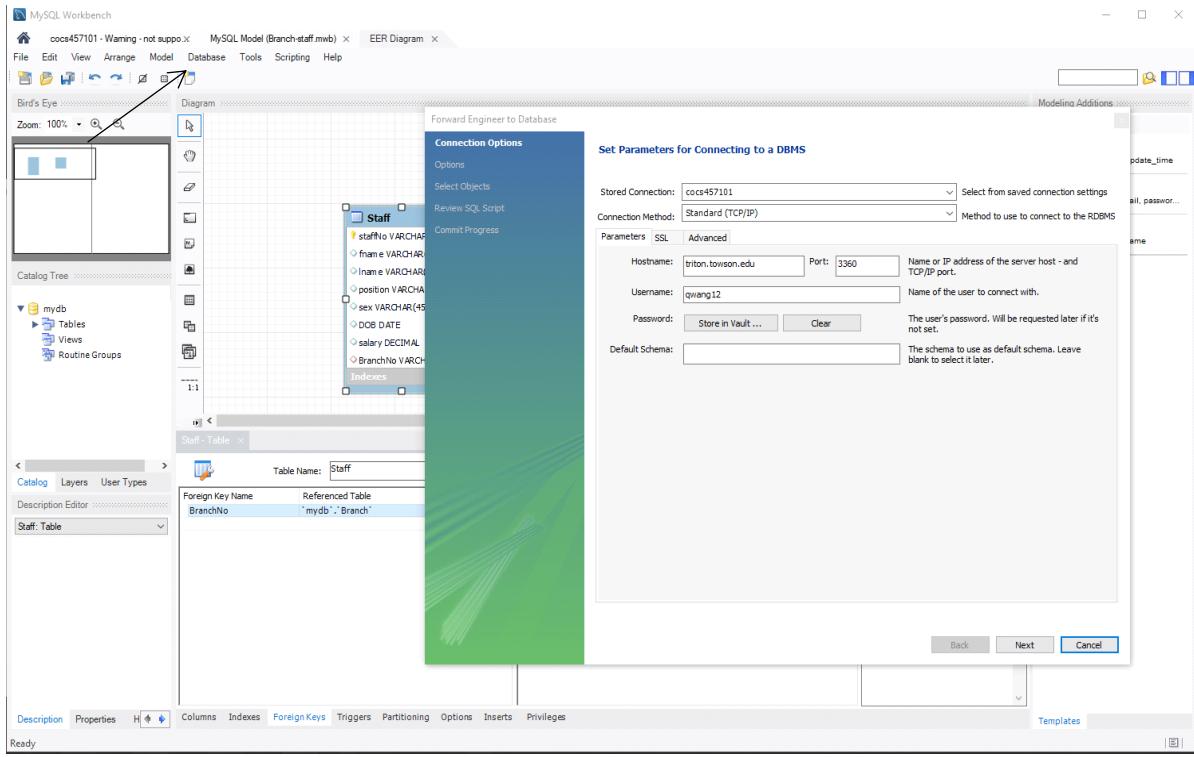


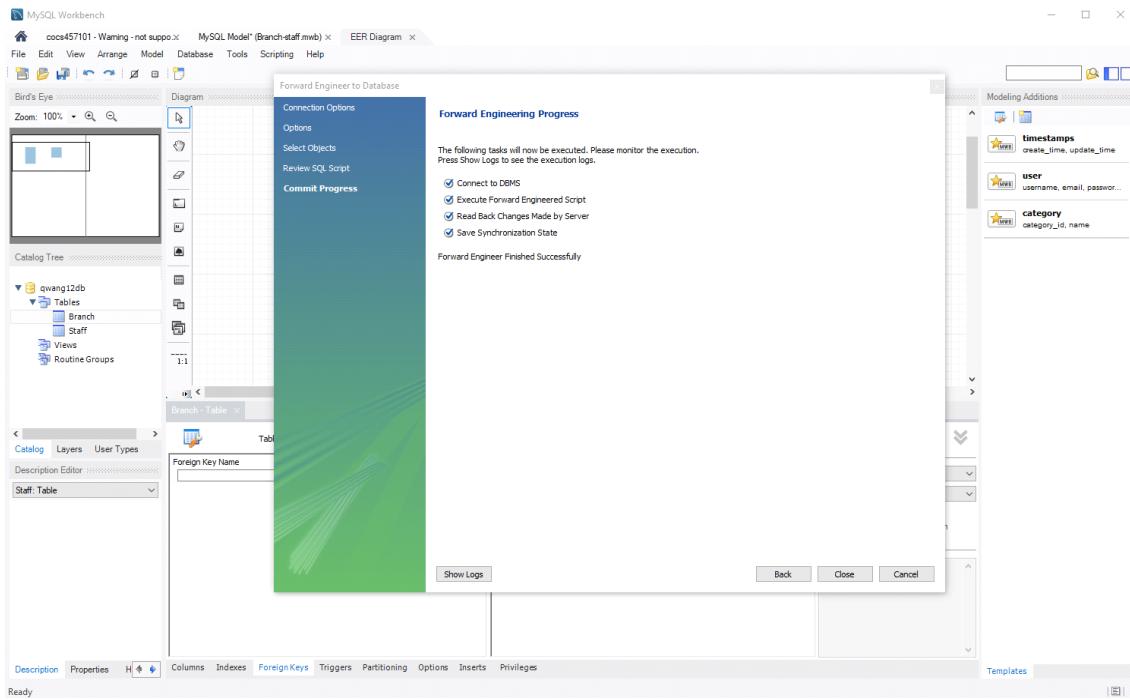
- Synchronize Database: forward engineering (create tables from an ERD), reverse engineering (Create an ERD from the database tables), and synchronize model (create or update ERD and create tables from an ERD at the same time)

#### Database Forward engineering

(Follow the wizard)







- Check out the tables in MySQL (Putty)

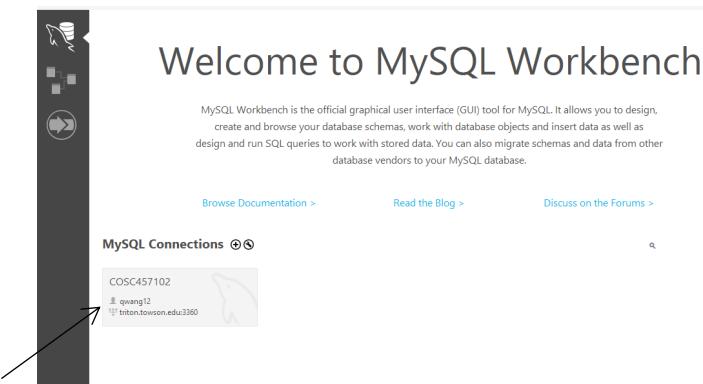
If the automatic process is successful then you will see the tables that are created in Workbench.

`MySQL>use database_name`

`MySQL>show tables;`

- Data Input (Please do not insert data from your EERD design. It does not save the entries)

Database Connect database, (right click the table in left side table tree), Select rows edit  
(when you finish entering data), click 'apply' in the right bottom



Screenshot of SQL Server Management Studio (SSMS) showing the results of a query against the 'Branch' table.

**Navigator:**

- SCHEMAS
  - qwang12db
    - Tables
      - Branch
      - Staff
      - Views
      - Stored Procedures
      - Functions

**Query 1:** Branch

```
1 •  SELECT * FROM qwang12db.Branch;
```

**Result Grid:**

BranchNo	Street	City	PostCode
NULL	NULL	NULL	NULL

**SQL Additions:**

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

**Object Info:**

Administration Schemas

Information

Table: Branch

Columns:

- BranchNo varchar(45) PK
- Street varchar(45)
- City varchar(45)
- PostCode varchar(45)

**Branch 1:**

Action Output

#	Time	Action	Message	Duration / Fetch
1	15:13:22	SELECT * FROM qwang12db.Branch LIMIT 0, 1000	0 row(s) returned	0.000 sec / 0.000 sec

**Query Completed:**

Screenshot of SQL Server Management Studio (SSMS) showing the results of a query against the 'Branch' table.

**Navigator:**

- SCHEMAS
  - qwang12db
    - Tables
      - Branch
      - Staff
      - Views
      - Stored Procedures
      - Functions

**Query 1:** Branch

```
1 •  SELECT * FROM qwang12db.Branch;
```

**Result Grid:**

BranchNo	Street	City	PostCode
B007	16 Argyll St	Aberdeen	AB2 3SU
NULL	NULL	NULL	NULL

**SQL Additions:**

d

c

**Object Info:**

Administration Schemas

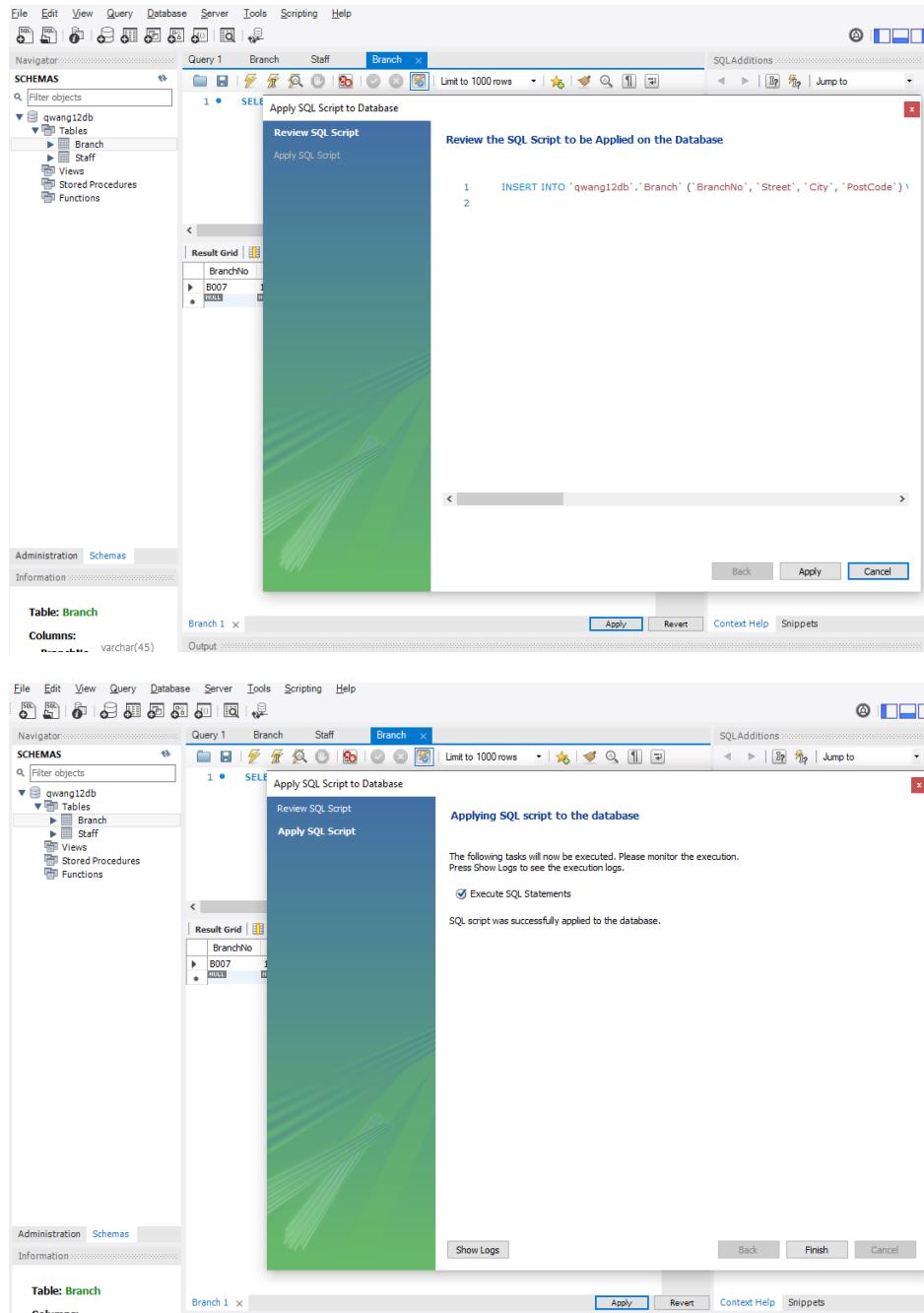
Information

Table: Branch

Columns:

**Branch 1:**

Apply Revert Con



- **Query Database**

Database connect database

Connect to the database. Then you will see the query editor.

Check out the content of the tables.

Screenshot of a SQL database management tool interface showing a query execution process.

The interface includes a top menu bar with File, Edit, View, Query, Database, Server, Tools, Scripting, Help, and a toolbar with various icons.

The Navigator pane on the left shows the schema structure:

- SCHEMAS: qwang12db
- Tables: Branch, Staff
- Views
- Stored Procedures
- Functions

The main workspace displays a query window titled "Query 1" with the tab "Branch" selected. The query entered is:

```
1 • SELECT * FROM qwang12db.Branch;
```

The results grid shows the data from the "Branch" table:

BranchNo	Street	City	PostCode
B007	16 Argyll St	Aberdeen	AB2 3SU
*	HULL	HULL	HULL

On the right side, there is a vertical toolbar with the following options:

- Result Grid (selected)
- Form Editor
- Field Types
- Query Stats
- Execution Plan

At the bottom, there are tabs for Administration and Schemas, and a section for Information.