Final Assessment User Guide

Oscars Database

Done by: Mohamed Ahmed Mohamed Yassin Harby

CREATE USER 'me'@'localhost' IDENTIFIED BY 'myUserPassword';

Type mysql -u me -p

Type the password, which is my user password

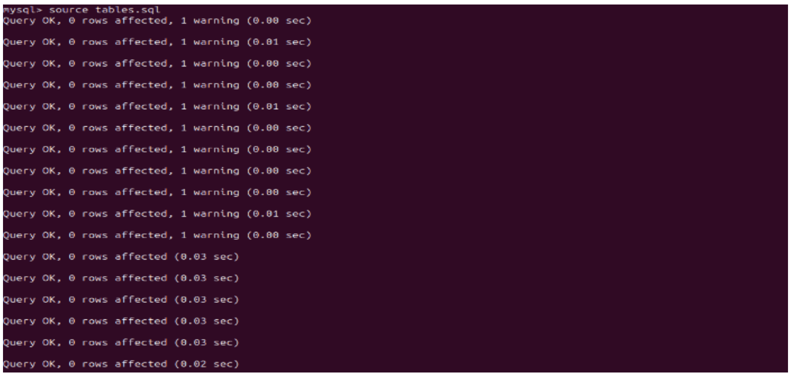
Create the database by typing:

CREATE DATABASE Oscars_21445116;

Then type: USE Oscars_21445116; to use the chosen database.

To create the tables type:

source tables.sql



```
mysql> source tables.sql
Query OK, 0 rows affected, 1 warning (0.00 sec)

Query OK, 0 rows affected, 1 warning (0.01 sec)

Query OK, 0 rows affected, 1 warning (0.00 sec)

Query OK, 0 rows affected, 1 warning (0.00 sec)

Query OK, 0 rows affected, 1 warning (0.01 sec)

Query OK, 0 rows affected, 1 warning (0.00 sec)

Query OK, 0 rows affected, 1 warning (0.00 sec)

Query OK, 0 rows affected, 1 warning (0.00 sec)

Query OK, 0 rows affected, 1 warning (0.00 sec)

Query OK, 0 rows affected, 1 warning (0.01 sec)

Query OK, 0 rows affected, 1 warning (0.00 sec)

Query OK, 0 rows affected (0.03 sec)

Query OK, 0 rows affected (0.03 sec)

Query OK, 0 rows affected (0.03 sec)

Query OK, 0 rows affected (0.03 sec)

Query OK, 0 rows affected (0.03 sec)

Query OK, 0 rows affected (0.02 sec)
```

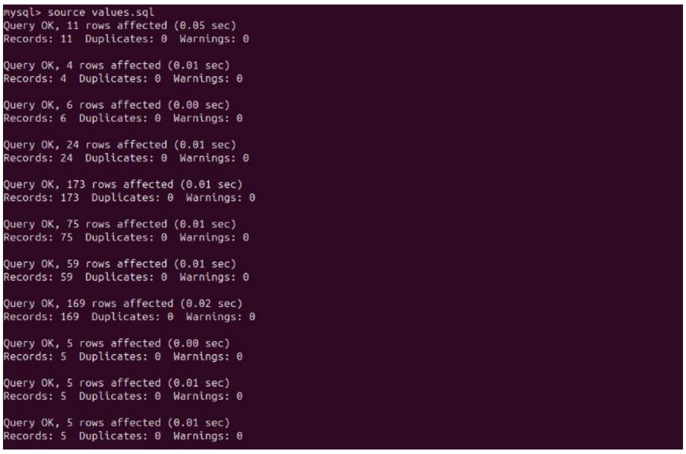
Type: SHOW TABLES;

To see all the defined tables

Type: DESC table name, for example DESC Actors;   to get the description of the Actors table

Then to populate the tables type:

Source values.sql



```
mysql> source values.sql
Query OK, 11 rows affected (0.05 sec)
Records: 11  Duplicates: 0  Warnings: 0

Query OK, 4 rows affected (0.01 sec)
Records: 4  Duplicates: 0  Warnings: 0

Query OK, 6 rows affected (0.00 sec)
Records: 6  Duplicates: 0  Warnings: 0

Query OK, 24 rows affected (0.01 sec)
Records: 24  Duplicates: 0  Warnings: 0

Query OK, 173 rows affected (0.01 sec)
Records: 173  Duplicates: 0  Warnings: 0

Query OK, 75 rows affected (0.01 sec)
Records: 75  Duplicates: 0  Warnings: 0

Query OK, 59 rows affected (0.01 sec)
Records: 59  Duplicates: 0  Warnings: 0

Query OK, 169 rows affected (0.02 sec)
Records: 169  Duplicates: 0  Warnings: 0

Query OK, 5 rows affected (0.00 sec)
Records: 5  Duplicates: 0  Warnings: 0

Query OK, 5 rows affected (0.01 sec)
Records: 5  Duplicates: 0  Warnings: 0

Query OK, 5 rows affected (0.01 sec)
Records: 5  Duplicates: 0  Warnings: 0
```

To run the pre-defined queries type:

source queries.sql

This will output the results of the 10 pre-defined queries.

Then type:

source AdvancedConcepts.sql to define the triggers, views and procedures.

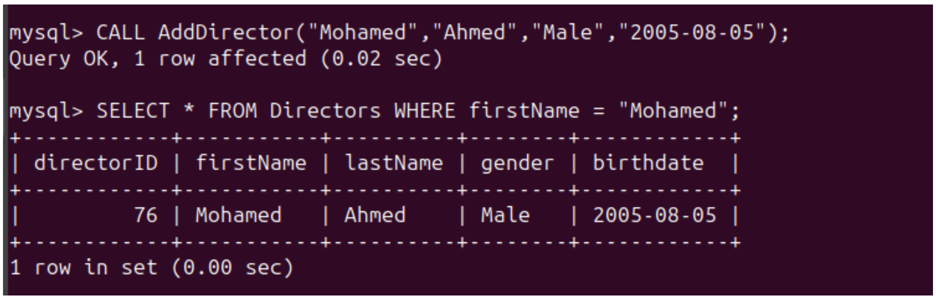
Type: SHOW PROCEDURE STATUS WHERE Db='Oscars_21445116';

To show the available procedures.

Calling procedures

First procedure

CALL AddDirector(firstName, lastName, gender, birthDate);



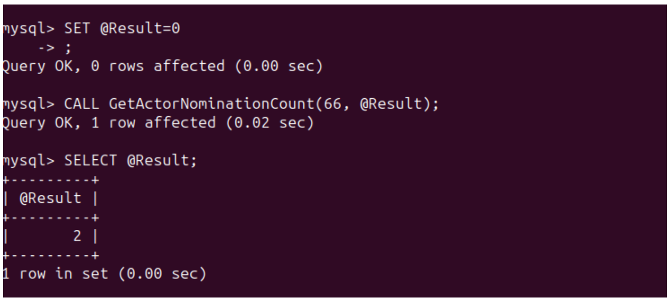
```
mysql> CALL AddDirector("Mohamed","Ahmed","Male","2005-08-05");
Query OK, 1 row affected (0.02 sec)

mysql> SELECT * FROM Directors WHERE firstName = "Mohamed";
+------------+-----------+----------+--------+------------+
| directorID | firstName | lastName | gender | birthdate  |
+------------+-----------+----------+--------+------------+
|         76 | Mohamed   | Ahmed    | Male   | 2005-08-05 |
+------------+-----------+----------+--------+------------+
1 row in set (0.00 sec)
```

Second procedure

SET @Result=0

CALL GetActorNominationCount (actorID, @Result);

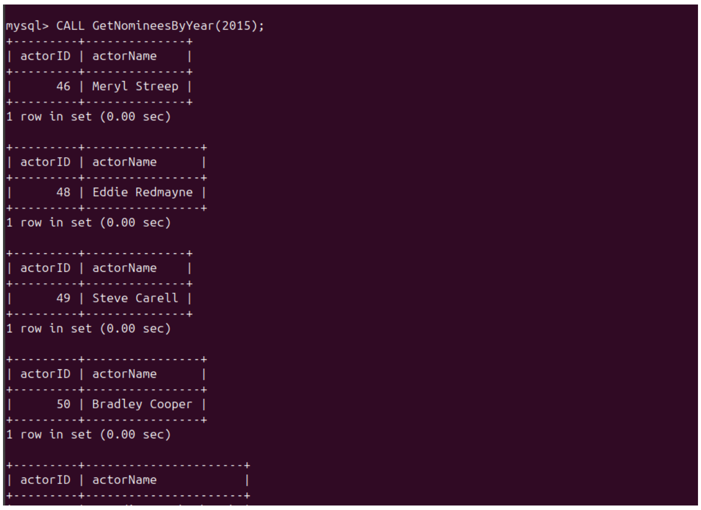
SELECT @Result;



```
mysql> SET @Result=0
    -> ;
Query OK, 0 rows affected (0.00 sec)

mysql> CALL GetActorNominationCount(66, @Result);
Query OK, 1 row affected (0.02 sec)

mysql> SELECT @Result;
+---------+
| @Result |
+---------+
|       2 |
+---------+
1 row in set (0.00 sec)
```

Third procedure

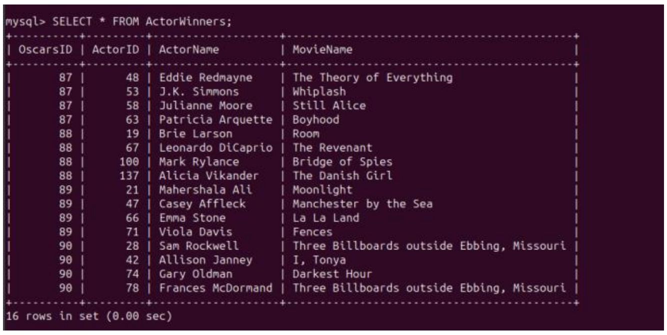
CALL GetNomineesByYear (year);



```
mysql> CALL GetNomineesByYear(2015);
+---------+--------------+
| actorID | actorName    |
+---------+--------------+
|      46 | Meryl Streep |
+---------+--------------+
1 row in set (0.00 sec)

+---------+----------------+
| actorID | actorName      |
+---------+----------------+
|      48 | Eddie Redmayne |
+---------+----------------+
1 row in set (0.00 sec)

+---------+--------------+
| actorID | actorName    |
+---------+--------------+
|      49 | Steve Carell |
+---------+--------------+
1 row in set (0.00 sec)

+---------+----------------+
| actorID | actorName      |
+---------+----------------+
|      50 | Bradley Cooper |
+---------+----------------+
1 row in set (0.00 sec)

+---------+----------------------+
| actorID | actorName            |
+---------+----------------------+
```

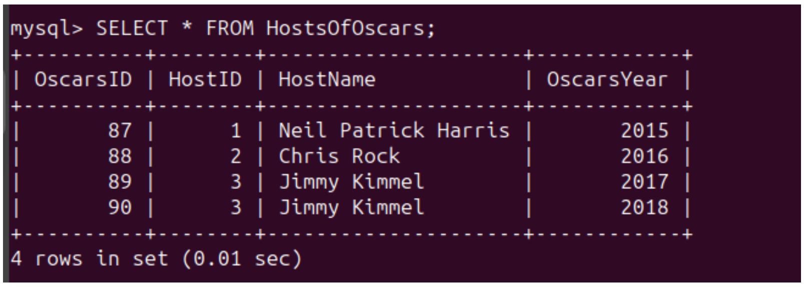
Calling views

First view
SELECT * FROM ActorWinners;



```
mysql> SELECT * FROM ActorWinners;
+----------+---------+-------------------+-------------------------------------------+
| OscarsID | ActorID | ActorName         | MovieName                                 |
+----------+---------+-------------------+-------------------------------------------+
|       87 |      48 | Eddie Redmayne    | The Theory of Everything                  |
|       87 |      53 | J.K. Simmons      | Whiplash                                  |
|       87 |      58 | Julianne Moore    | Still Alice                               |
|       87 |      63 | Patricia Arquette | Boyhood                                   |
|       88 |      19 | Brie Larson       | Room                                      |
|       88 |      67 | Leonardo DiCaprio | The Revenant                              |
|       88 |     100 | Mark Rylance      | Bridge of Spies                           |
|       88 |     137 | Alicia Vikander   | The Danish Girl                           |
|       89 |      21 | Mahershala Ali    | Moonlight                                 |
|       89 |      47 | Casey Affleck     | Manchester by the Sea                     |
|       89 |      66 | Emma Stone        | La La Land                                |
|       89 |      71 | Viola Davis       | Fences                                    |
|       90 |      28 | Sam Rockwell      | Three Billboards outside Ebbing, Missouri |
|       90 |      42 | Allison Janney    | I, Tonya                                  |
|       90 |      74 | Gary Oldman       | Darkest Hour                              |
|       90 |      78 | Frances McDormand | Three Billboards outside Ebbing, Missouri |
+----------+---------+-------------------+-------------------------------------------+
16 rows in set (0.00 sec)
```

Second view
SELECT * FROM HostsOfOscar;



```
mysql> SELECT * FROM HostsOfOscars;
+----------+--------+--------------------+------------+
| OscarsID | HostID | HostName           | OscarsYear |
+----------+--------+--------------------+------------+
|       87 |      1 | Neil Patrick Harris |       2015 |
|       88 |      2 | Chris Rock         |       2016 |
|       89 |      3 | Jimmy Kimmel       |       2017 |
|       90 |      3 | Jimmy Kimmel       |       2018 |
+----------+--------+--------------------+------------+
4 rows in set (0.01 sec)
```

To exit the mysql

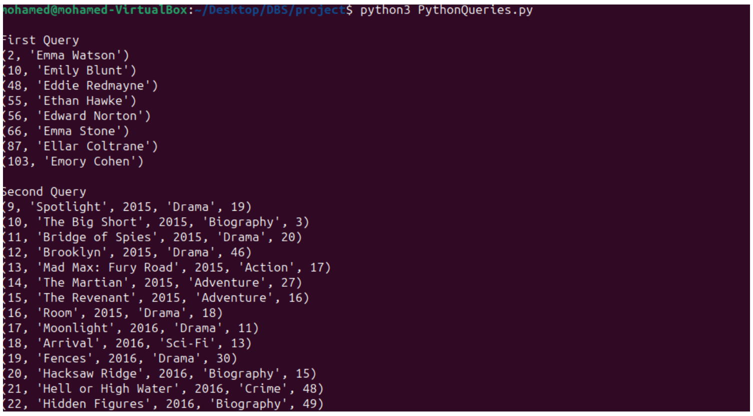
Type: exit

Connecting to python

Install python connector on the terminal.

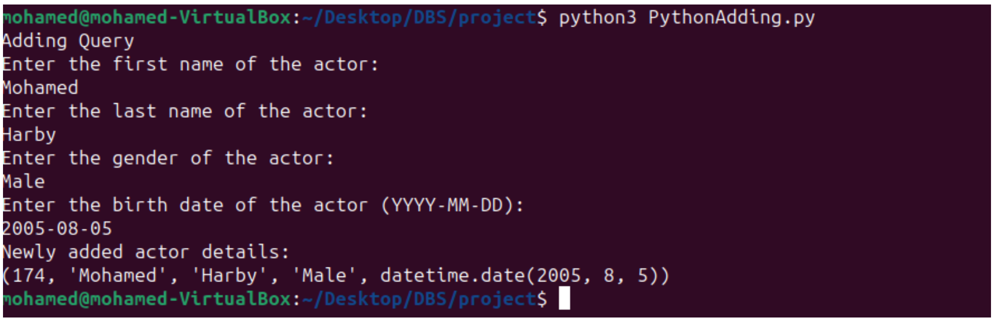
Type: pip install mysql-connector-python

Type: python3 PythonQueries.py



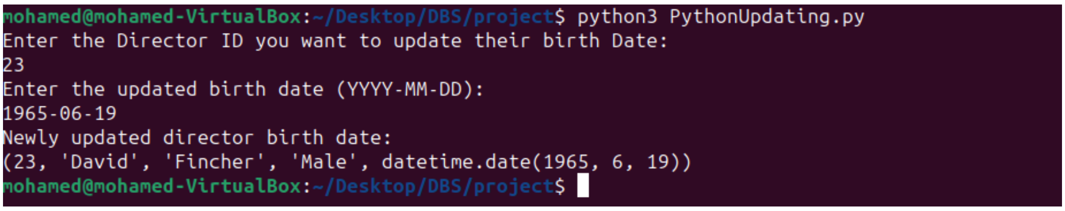
```
mohamed@mohamed-VirtualBox:~/Desktop/DBS/project$ python3 PythonQueries.py

First Query
(2, 'Emma Watson')
(10, 'Emily Blunt')
(48, 'Eddie Redmayne')
(55, 'Ethan Hawke')
(56, 'Edward Norton')
(66, 'Emma Stone')
(87, 'Ellar Coltrane')
(103, 'Emory Cohen')

Second Query
(9, 'Spotlight', 2015, 'Drama', 19)
(10, 'The Big Short', 2015, 'Biography', 3)
(11, 'Bridge of Spies', 2015, 'Drama', 20)
(12, 'Brooklyn', 2015, 'Drama', 46)
(13, 'Mad Max: Fury Road', 2015, 'Action', 17)
(14, 'The Martian', 2015, 'Adventure', 27)
(15, 'The Revenant', 2015, 'Adventure', 16)
(16, 'Room', 2015, 'Drama', 18)
(17, 'Moonlight', 2016, 'Drama', 11)
(18, 'Arrival', 2016, 'Sci-Fi', 13)
(19, 'Fences', 2016, 'Drama', 30)
(20, 'Hacksaw Ridge', 2016, 'Biography', 15)
(21, 'Hell or High Water', 2016, 'Crime', 48)
(22, 'Hidden Figures', 2016, 'Biography', 49)
```

Type: python3 PythonAdding.py



```
mohamed@mohamed-VirtualBox:~/Desktop/DBS/project$ python3 PythonAdding.py
Adding Query
Enter the first name of the actor:
Mohamed
Enter the last name of the actor:
Harby
Enter the gender of the actor:
Male
Enter the birth date of the actor (YYYY-MM-DD):
2005-08-05
Newly added actor details:
(174, 'Mohamed', 'Harby', 'Male', datetime.date(2005, 8, 5))
mohamed@mohamed-VirtualBox:~/Desktop/DBS/project$ 
```

Type: python3 PythonUpdating.py



```
mohamed@mohamed-VirtualBox:~/Desktop/DBS/project$ python3 PythonUpdating.py
Enter the Director ID you want to update their birth Date:
23
Enter the updated birth date (YYYY-MM-DD):
1965-06-19
Newly updated director birth date:
(23, 'David', 'Fincher', 'Male', datetime.date(1965, 6, 19))
mohamed@mohamed-VirtualBox:~/Desktop/DBS/project$ 
```