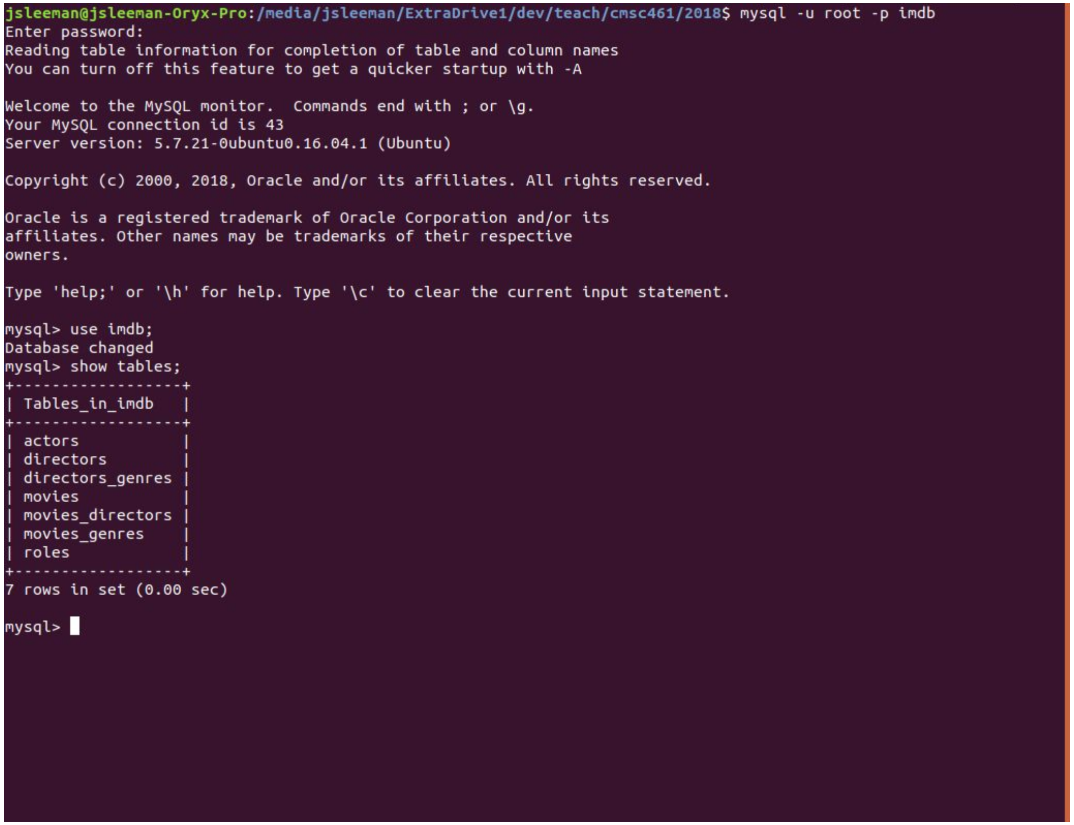
Homework 5

Due Friday 4/20/2018

In order to complete the questions in this homework assignment, you will need to create a local version of the following database:

https://relational.fit.cvut.cz/dataset/IMDb

1. Follow the steps described in class and create a local version of the IMDB database. Using mysqldump, create a ***myimdb.sql***. Create a local database called ***imdb***. Import ***myimdb.sql*** into your local database. Take a screenshot from the command line that includes your mysql login, the ***use imdb*** command, and the ***show tables*** command which results in the following listing. I want to see a screenshot of this activity. The screenshot should capture the information in the screenshot I provided. Do not use MySQL Workbench for this activity. (10 points)



```
jsleeman@jsleeman-Oryx-Pro:/media/jsleeman/ExtraDrive1/dev/teach/cmsc461/2018$ mysql -u root -p imdb
Enter password:
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 43
Server version: 5.7.21-0ubuntu0.16.04.1 (Ubuntu)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use imdb;
Database changed
mysql> show tables;
+-----------------+
| Tables_in_imdb  |
+-----------------+
| actors          |
| directors       |
| directors_genres |
| movies          |
| movies_directors |
| movies_genres   |
| roles           |
+-----------------+
7 rows in set (0.00 sec)

mysql> 
```

2. Follow the below steps in ***sequential order***, then answer the questions on the next page. Use MySQL-workbench to answer these questions. (40 points)

Step 1: Connect to your local version of IMDB. Create 5 query tabs.

In quey tab 1, run the following query (limit 50000) 3 times. After the first run, log stats in bottom table on page 5 for the row labeled "tab 1". After each run, log the stats in the top table on page 5 for the row labeled "tab 1".

select movies.year, movies.name from movies where movies.year >=1990 and movies.year <= 2000 order by year;

Step 2: In query tab 2, run the following query (limit 50000) 3 times. After the first run, log stats in bottom table on page 5 for the row labeled "tab 2". After each run, log the stats in the top table on page 5 for the row labeled "tab 2".

select movies.year, movies.name from movies where movies.year >=1990 and movies.year <= 2000;

Step 3: In query tab 3, Create an index on ***year*** and show all indices on table ***movies***. Verify ***year*** is indexed.

Step 4: In query tab 4, run the following query (limit 50000) 3 times. After the first run, log stats in bottom table on page 5 for the row labeled "tab 4". After each run, log the stats in the top table on page 5 for the row labeled "tab 4".

select movies.year, movies.name from movies where movies.year >=1990 and movies.year <= 2000 order by year;

Step 5: In query tab 5, run the following query (limit 50000) 3 times. After the first run, log stats in bottom table on page 5 for the row labeled "tab 5". After each run, log the stats in the top table on page 5 for the row labeled "tab 5".

select movies.year, movies.name from movies where movies.year >=1990 and movies.year <= 2000;

Question 2 (cont.) Fill in the following table:

	Execution Time at the Client Run 1	Execution Time at the Server Run 1	Execution Time at the Client Run 2	Execution Time at the Server Run 2	Execution Time at the Client Run 3	Execution Time at the Server Run 3
Query Tab 1						
Query Tab 2						
Query Tab 4						
Query Tab 5						

	Number of Full Table Scans	Rows Examined	Joins Using Range	Sorted Rows	Sort Merge Passes	Sorts With Table Scans	Sorts With Ranges	Index Usage
Query Tab 1								
Query Tab 2								
Query Tab 4								
Query Tab 5								

Question 2 (cont.)

(1.) In which tab(s) did the query use the index on ***<u>year</u>***?

(2.) How did the performance differ between query tab 2 and query tab 5? Did indexing have any influence on performance? Why or why not?

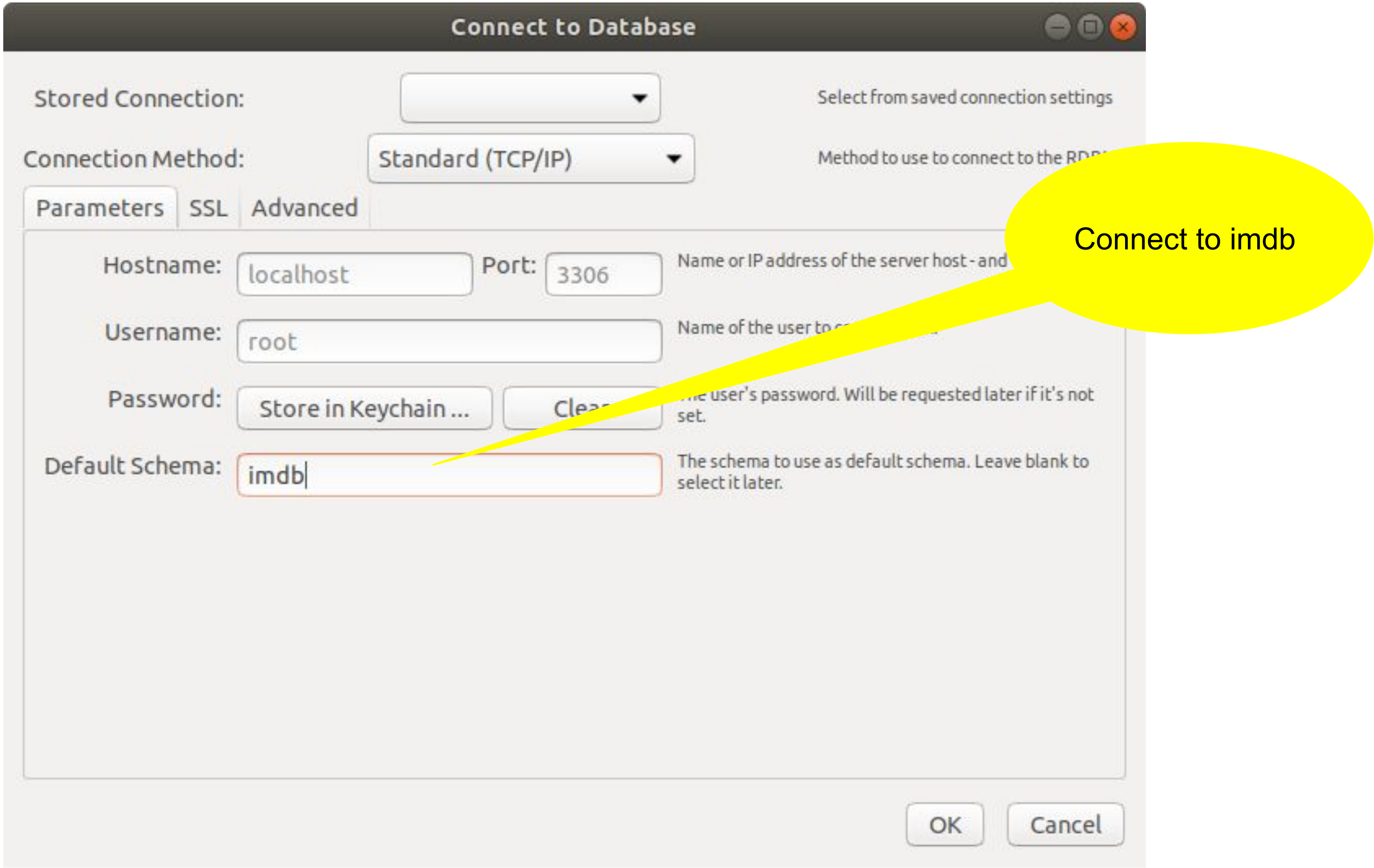
(3.) Did query tab 4 have a value for Sorted Rows? If not, why?

(4.) Did the result set order differ between query tab 1 and query tab 4? If so, why?

(5.) What does execution time in MySQL-Workbench tell us? What is the difference between client and server execution time? How accurate are they?

(6.) Any other observations?

How to view a query plan and statistics in MySQL-workbench



Connect to Database

Stored Connection:

Select from saved connection settings

Connection Method: Standard (TCP/IP)

Method to use to connect to the RDB

Hostname: localhost Port: 3306

Name or IP address of the server host - and

Username: root

Name of the user to c

Password:

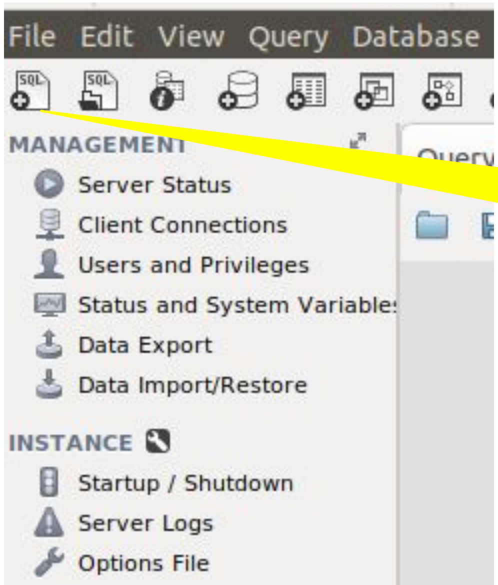
e user's password. Will be requested later if it's not set.

Default Schema: imdb

The schema to use as default schema. Leave blank to select it later.

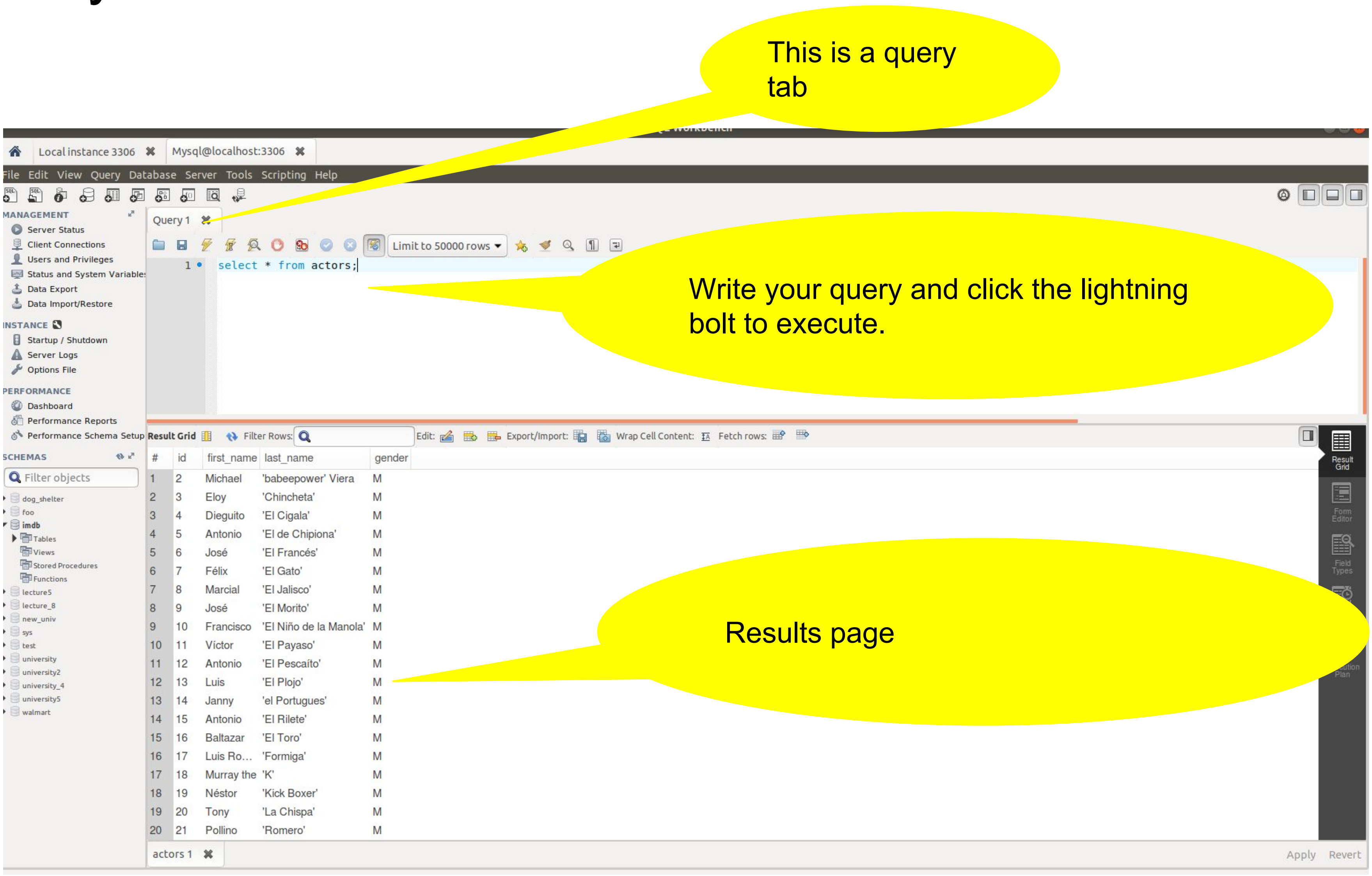
Connect to imdb

How to view a query plan and statistics in MySQL-workbench



Click on this button to create a new query tab

How to view a query plan and statistics in MySQL-workbench



This is a query tab

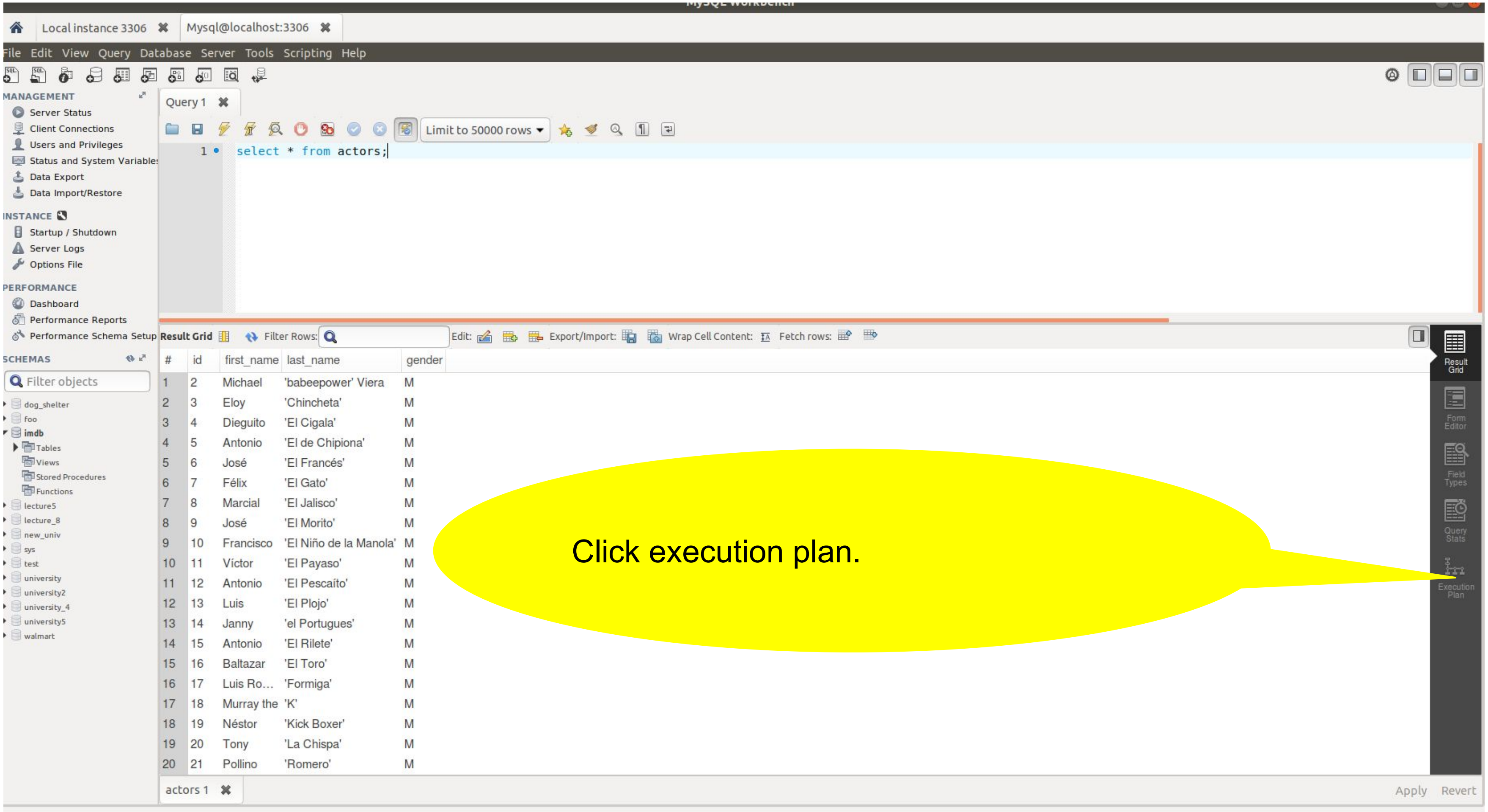
```sql
select * from actors;
```

Write your query and click the lightning bolt to execute.

#	id	first_name	last_name	gender
1	2	Michael	'babeepower' Viera	M
2	3	Eloy	'Chincheta'	M
3	4	Dieguito	'El Cigala'	M
4	5	Antonio	'El de Chipiona'	M
5	6	José	'El Francés'	M
6	7	Félix	'El Gato'	M
7	8	Marcial	'El Jalisco'	M
8	9	José	'El Morito'	M
9	10	Francisco	'El Niño de la Manola'	M
10	11	Víctor	'El Payaso'	M
11	12	Antonio	'El Pescaíto'	M
12	13	Luis	'El Plojo'	M
13	14	Janny	'el Portugues'	M
14	15	Antonio	'El Rilete'	M
15	16	Baltazar	'El Toro'	M
16	17	Luis Ro...	'Formiga'	M
17	18	Murray the	'K'	M
18	19	Néstor	'Kick Boxer'	M
19	20	Tony	'La Chispa'	M
20	21	Pollino	'Romero'	M

Results page

How to view a query plan and statistics in MySQL-workbench

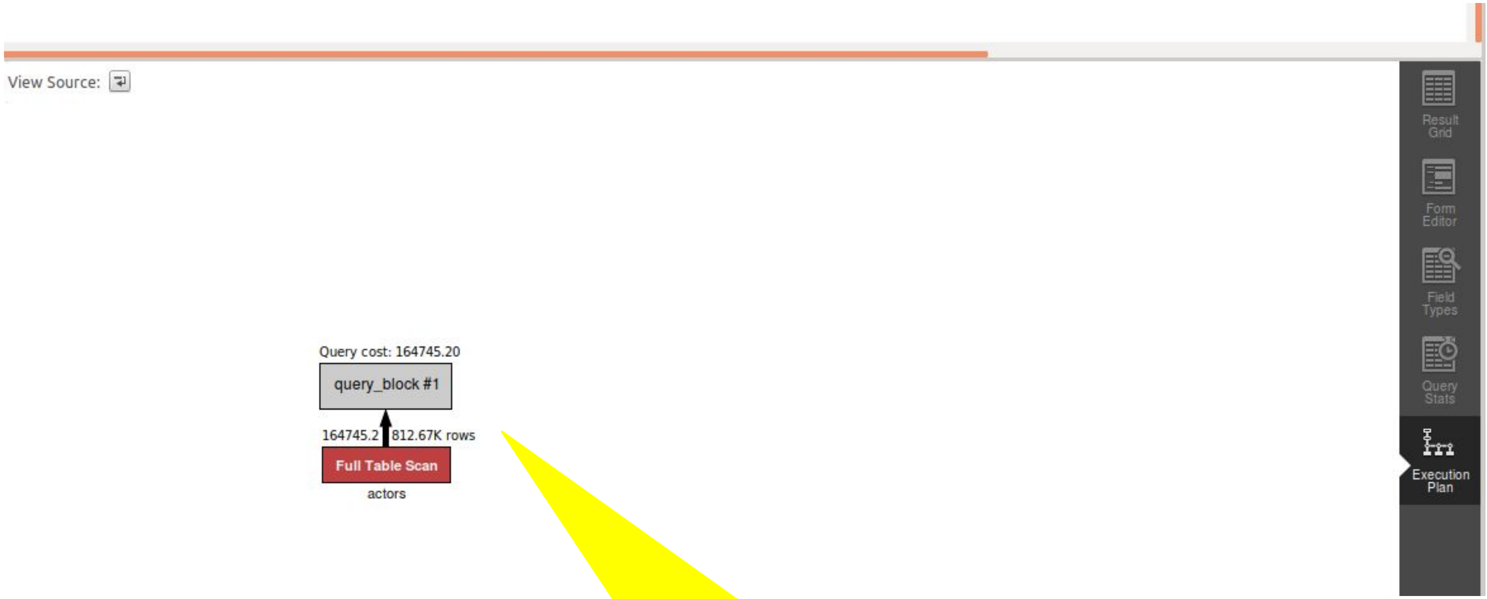


```
select * from actors;
```

#	id	first_name	last_name	gender
1	2	Michael	'babeepower' Viera	M
2	3	Eloy	'Chincheta'	M
3	4	Dieguito	'El Cigala'	M
4	5	Antonio	'El de Chipiona'	M
5	6	José	'El Francés'	M
6	7	Félix	'El Gato'	M
7	8	Marcial	'El Jalisco'	M
8	9	José	'El Morito'	M
9	10	Francisco	'El Niño de la Manola'	M
10	11	Víctor	'El Payaso'	M
11	12	Antonio	'El Pescaíto'	M
12	13	Luis	'El Plojo'	M
13	14	Janny	'el Portugues'	M
14	15	Antonio	'El Rilete'	M
15	16	Baltazar	'El Toro'	M
16	17	Luis Ro...	'Formiga'	M
17	18	Murray the	'K'	M
18	19	Néstor	'Kick Boxer'	M
19	20	Tony	'La Chispa'	M
20	21	Pollino	'Romero'	M

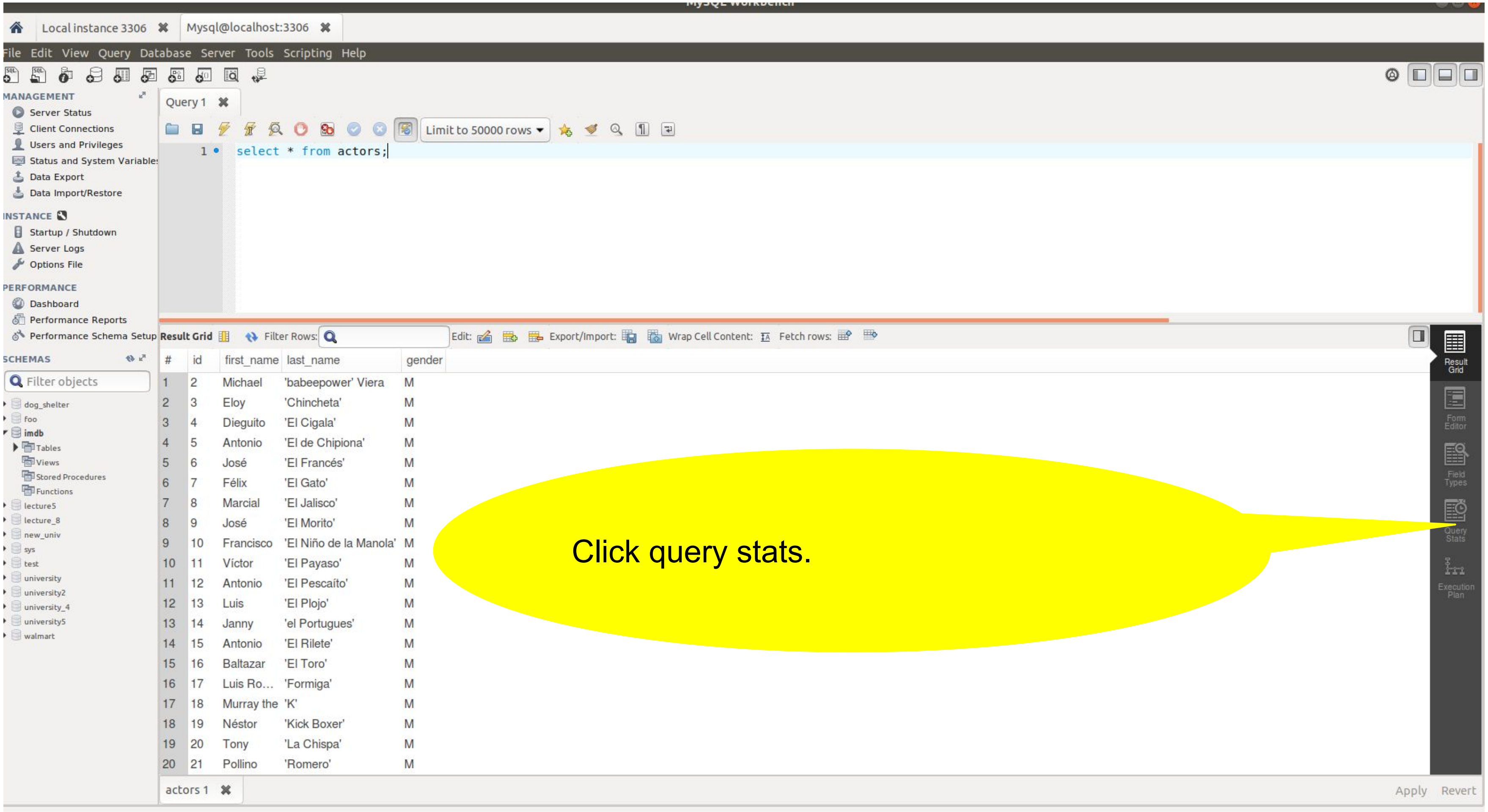
Click execution plan.

How to view a query plan and statistics in MySQL-workbench



This page will show you the execution plan.

How to view a query plan and statistics in MySQL-workbench



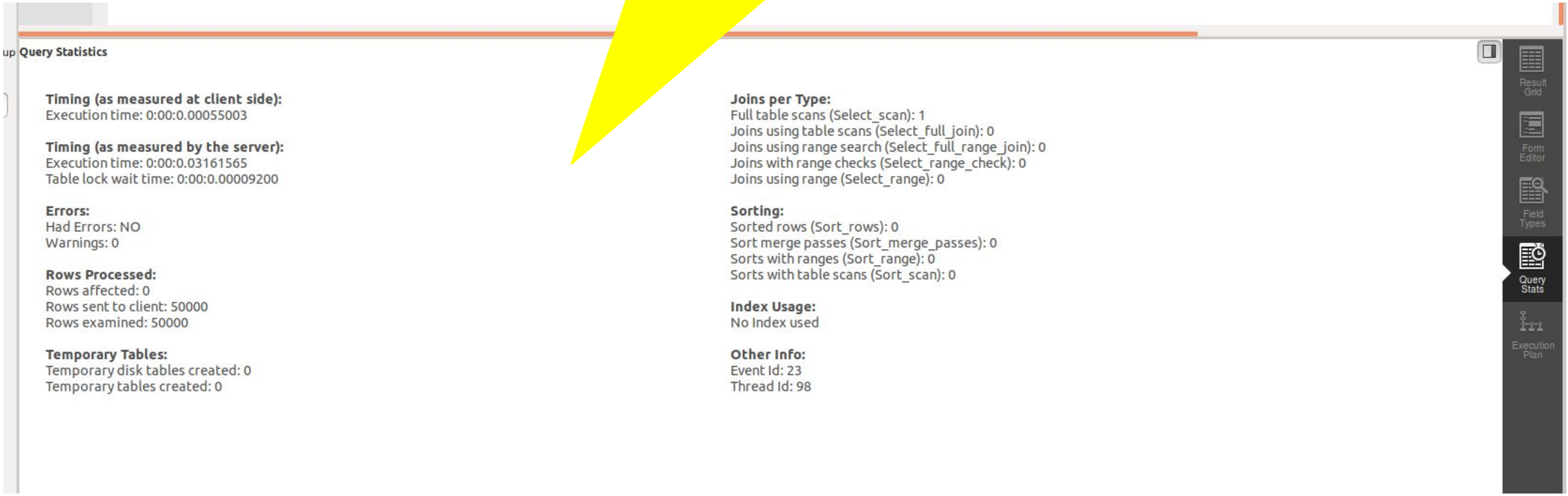
```
select * from actors;
```

#	id	first_name	last_name	gender
1	2	Michael	'babeepower' Viera	M
2	3	Eloy	'Chincheta'	M
3	4	Dieguito	'El Cigala'	M
4	5	Antonio	'El de Chipiona'	M
5	6	José	'El Francés'	M
6	7	Félix	'El Gato'	M
7	8	Marcial	'El Jalisco'	M
8	9	José	'El Morito'	M
9	10	Francisco	'El Niño de la Manola'	M
10	11	Víctor	'El Payaso'	M
11	12	Antonio	'El Pescaíto'	M
12	13	Luis	'El Plojo'	M
13	14	Janny	'el Portugues'	M
14	15	Antonio	'El Rilete'	M
15	16	Baltazar	'El Toro'	M
16	17	Luis Ro…	'Formiga'	M
17	18	Murray the	'K'	M
18	19	Néstor	'Kick Boxer'	M
19	20	Tony	'La Chispa'	M
20	21	Pollino	'Romero'	M

Click query stats.

How to view a query plan and statistics in MySQL-workbench

The columns in the table on page 5 correlate to the attributes on this page.



Query Statistics

Timing (as measured at client side):
Execution time: 0:00:0.00055003

Timing (as measured by the server):
Execution time: 0:00:0.03161565
Table lock wait time: 0:00:0.00009200

Errors:
Had Errors: NO
Warnings: 0

Rows Processed:
Rows affected: 0
Rows sent to client: 50000
Rows examined: 50000

Temporary Tables:
Temporary disk tables created: 0
Temporary tables created: 0

Joins per Type:
Full table scans (Select_scan): 1
Joins using table scans (Select_full_join): 0
Joins using range search (Select_full_range_join): 0
Joins with range checks (Select_range_check): 0
Joins using range (Select_range): 0

Sorting:
Sorted rows (Sort_rows): 0
Sort merge passes (Sort_merge_passes): 0
Sorts with ranges (Sort_range): 0
Sorts with table scans (Sort_scan): 0

Index Usage:
No Index used

Other Info:
Event Id: 23
Thread Id: 98