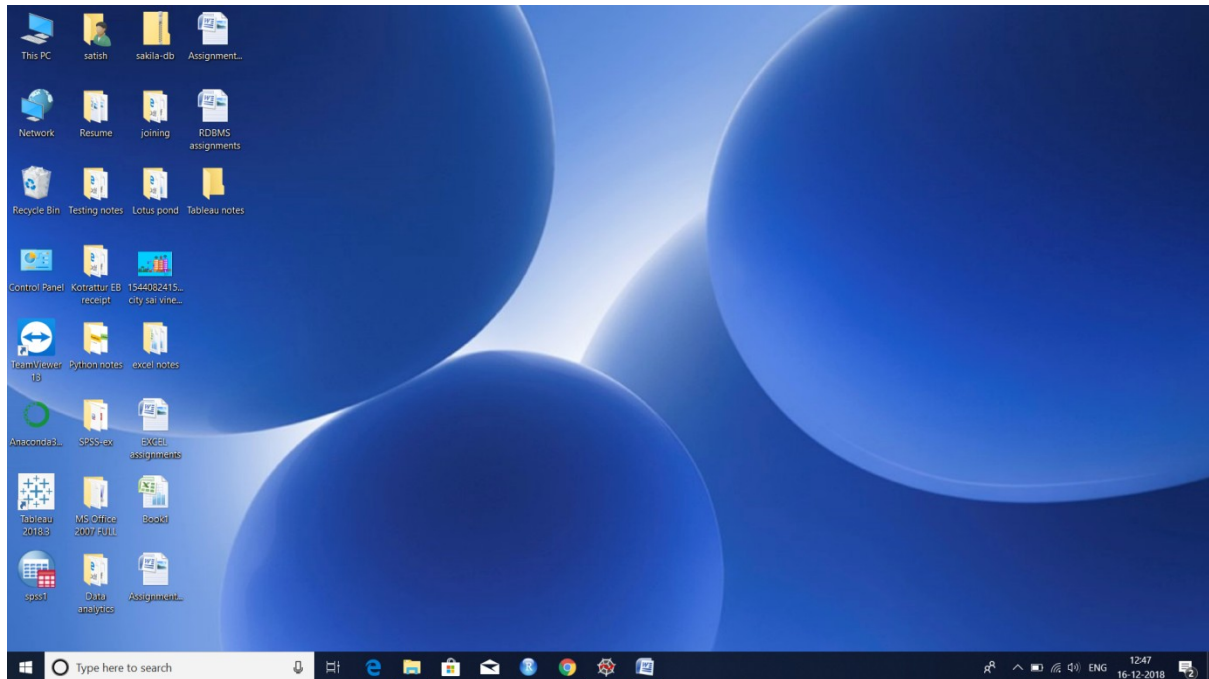


## Task 1

Set up/Install the database sakila. The database will be used for next MySQL Assignments as well. Refer the Links given above for help in setting up the database.

Downloaded Sakila database from the links to desktop



Moved Sakila database into MySQL

```
mysql> show databases;
+-----+
| Database |
+-----+
| employee |
| information_schema |
| mysql |
| performance_schema |
| sakila |
| sys |
+-----+
6 rows in set (0.00 sec)

mysql> use sakila database
Database changed
```

## Task 2

1. What are the names of all the countries in the databases which start with the letter "B"

(Sorted by their names)?

```
mysql> select country from country
-> where country like 'b%'
-> order by country;
+-----+
| country |
+-----+
| Bahrain |
| Bangladesh |
| Belarus |
| Bolivia |
| Brazil |
| Brunei |
| Bulgaria |
+-----+
7 rows in set (0.00 sec)
```

2. Return the first names (sorted) of all the actors with the last name "berry".

```
mysql> select first_name, last_name from actor
-> where last_name = "berry"
-> order by first_name;
+-----+-----+
| first_name | last_name |
+-----+-----+
| CHRISTOPHER | BERRY |
| HENRY | BERRY |
| KARL | BERRY |
+-----+-----+
3 rows in set (0.04 sec)
```

3. Find all the films whose length is more than 184 (inclusive). Order the results by the length  
( and for films with the same length order them by their name). Return their title and the length.

```
mysql> select title,length from film  
-> where length >=184  
-> order by length;
```

title	length
CONSPIRACY SPIRIT	184
CRYSTAL BREAKING	184
KING EVOLUTION	184
MOONWALKER FOOL	184
SMOOCHY CONTROL	184
SONS INTERVIEW	184
SORORITY QUEEN	184
THEORY MERMAID	184
CHICAGO NORTH	185
CONTROL ANTHEM	185
DARN FORRESTER	185
GANGS PRIDE	185
HOME PITY	185
MUSCLE BRIGHT	185
POND SEATTLE	185
SOLDIERS EVOLUTION	185
SWEET BROTHERHOOD	185
WORST BANGER	185

```
18 rows in set (0.00 sec)
```

### Task 3

1. Return the categories (names) of the longest film. NOTE that there may be several "longest" films (i.e. with the same length), so you might need to return more than one category. Return the duration as well.

```
mysql> select category,length,duration from film_lst where length = (select max(length) from film_lst);
```

category	length	duration
Games	185	6
Comedy	185	7
Action	185	7
Animation	185	4
Music	185	7
Travel	185	7
Animation	185	7
Sci-Fi	185	7
Travel	185	3
Action	185	4

```
10 rows in set, 1 warning (0.04 sec)
```

2. Find the movies whose total number of actors is above the average. Return the movie names And its number of actors ordered by the title. IMPORTANT NOTE: this query should return many movies. Please write in your submission only the first TOP-10 results.

Ans: In the tables there is no column with number of actors. Hence average cannot be found. Hence movie names and its number of actors ordered by the title cannot be found.

3. Who is the customer who spent the most on rental movies? Return his/her customer id, first name and the amount spent.

Most on rental movies = max rental rate. Amount spent = rental rate

```
mysql> select customer_id,first_name,rental_rate from customer_rental where rental_rate = (select max(rental_rate) from customer_rental);
```

customer_id	first_name	rental_rate
2	PATRICIA	4.99
7	MARIA	4.99
8	SUSAN	4.99
10	DOROTHY	4.99
13	KAREN	4.99
20	SHARON	4.99
21	MICHELLE	4.99
28	CYNTHIA	4.99
31	BRENDA	4.99
32	AMY	4.99
44	MARIE	4.99
45	JANET	4.99
46	CATHERINE	4.99
47	FRANCES	4.99
48	ANN	4.99
60	MILDRED	4.99
61	KATHERINE	4.99
65	ROSE	4.99
68	NICOLE	4.99
70	CHRISTINA	4.99
71	KATHY	4.99
74	DENISE	4.99
75	TAMMY	4.99
77	JANE	4.99
81	ANDREA	4.99
84	SARA	4.99
86	JACQUELINE	4.99
88	BONNIE	4.99
92	TINA	4.99
93	PHYLLIS	4.99
95	PAULA	4.99
98	LILLIAN	4.99
100	ROBIN	4.99
102	CRYSTAL	4.99
103	GLADYS	4.99
112	ROSA	4.99
113	CINDY	4.99
117	EDITH	4.99
120	SYLVIA	4.99
123	SHANNON	4.99
124	SHEILA	4.99
126	ELLEN	4.99
127	ELAINE	4.99
131	MONICA	4.99
133	PAULINE	4.99

133	PAULINE	4.99
134	EMMA	4.99
139	AMBER	4.99
141	DEBBIE	4.99
144	CLARA	4.99
145	LUCILLE	4.99
151	MEGAN	4.99
156	BERTHA	4.99
159	JILL	4.99
161	GERALDINE	4.99
165	LORRAINE	4.99
167	SALLY	4.99
170	BEATRICE	4.99
171	DOLORES	4.99
172	BERNICE	4.99
173	AUDREY	4.99
174	YVONNE	4.99
177	SAMANTHA	4.99
179	DANA	4.99
182	RENEE	4.99
183	IDA	4.99
190	YOLANDA	4.99
192	LAURIE	4.99
195	VANESSA	4.99
202	CARLA	4.99
203	TARA	4.99
207	GERTRUDE	4.99
210	ELLA	4.99
211	STACEY	4.99
212	WILMA	4.99
214	KRISTIN	4.99
215	JESSIE	4.99
216	NATALIE	4.99
217	AGNES	4.99
219	WILLIE	4.99
222	DELORES	4.99
224	PEARL	4.99
226	MAUREEN	4.99
227	COLLEEN	4.99
229	TAMARA	4.99
239	MINNIE	4.99
243	LYDIA	4.99
244	VIOLA	4.99
246	MARIAN	4.99
248	CAROLINE	4.99
251	VICKIE	4.99
253	TERRY	4.99
255	IRMA	4.99
256	MABEL	4.99

251	VICKIE	4.99	
253	TERRY	4.99	
255	IRMA	4.99	
256	MABEL	4.99	
260	CHRISTY	4.99	
263	HILDA	4.99	
265	JENNIE	4.99	
267	MARGIE	4.99	
268	NINA	4.99	
271	PENNY	4.99	
272	KAY	4.99	
276	BRANDY	4.99	
277	OLGA	4.99	
278	BILLIE	4.99	
279	DIANNE	4.99	
284	SONIA	4.99	
289	VIOLET	4.99	
294	SHELLY	4.99	
300	JOHN	4.99	
307	JOSEPH	4.99	
310	DANIEL	4.99	
312	MARK	4.99	
313	DONALD	4.99	
316	STEVEN	4.99	
320	ANTHONY	4.99	
321	KEVIN	4.99	
323	MATTHEW	4.99	
324	GARY	4.99	
327	LARRY	4.99	
330	SCOTT	4.99	
334	RAYMOND	4.99	
336	JOSHUA	4.99	
338	DENNIS	4.99	
340	PATRICK	4.99	
342	HAROLD	4.99	
346	ARTHUR	4.99	
347	RYAN	4.99	
350	JUAN	4.99	
354	JUSTIN	4.99	
359	WILLIE	4.99	
365	BRUCE	4.99	
369	FRED	4.99	
371	BILLY	4.99	
372	STEVE	4.99	
381	BOBBY	4.99	
384	ERNEST	4.99	
385	PHILIP	4.99	

385	PHILLIP	4.99	
386	TODD	4.99	
390	SHAWN	4.99	
392	SEAN	4.99	
396	EARL	4.99	
398	ANTONIO	4.99	
403	MIKE	4.99	
405	LEONARD	4.99	
408	MANUEL	4.99	
409	RODNEY	4.99	
411	NORMAN	4.99	
412	ALLEN	4.99	
420	JACOB	4.99	
421	LEE	4.99	
422	MELVIN	4.99	
426	BRADLEY	4.99	
435	RICKY	4.99	
439	ALEXANDER	4.99	
444	MARCUS	4.99	
446	THEODORE	4.99	
451	JIM	4.99	
455	JON	4.99	
456	RONNIE	4.99	
459	TOMMY	4.99	
460	LEON	4.99	
465	FLOYD	4.99	
466	LEO	4.99	
469	WESLEY	4.99	
470	GORDON	4.99	
472	GREG	4.99	
485	CLYDE	4.99	
486	GLEN	4.99	
487	HECTOR	4.99	
499	MARC	4.99	
500	REGINALD	4.99	
501	RUBEN	4.99	
506	LESLIE	4.99	
508	MILTON	4.99	
510	BEN	4.99	
512	CECIL	4.99	
515	ANDRE	4.99	
518	GABRIEL	4.99	
521	ROLAND	4.99	
527	CORY	4.99	
532	NEIL	4.99	
543	LANCE	4.99	
551	CLAYTON	4.99	



543	LANCE	4.99
551	CLAYTON	4.99
556	ARMANDO	4.99
563	KEN	4.99
565	JAIME	4.99
568	ALBERTO	4.99
570	IVAN	4.99
575	ISAAC	4.99
578	WILLARD	4.99
579	DARYL	4.99
580	ROSS	4.99
583	MARSHALL	4.99
587	SERGIO	4.99
588	MARION	4.99
591	KENT	4.99
596	ENRIQUE	4.99
597	FREDDIE	4.99

-----  
197 rows in set (0.01 sec)

4. Give an interesting query of your own that is not already in the assignment. The query should involve at least two joins, HAVING clause and aggregation operation. Give the answer.

```
mysql> select actor.first_name, actor.last_name, category.name AS name, film.description AS description, film.rental_rate, film.length, count(length)
-> from actor
-> join category on actor.actor_id = category.category_id
-> join film on film.film_id = category.category_id
-> GROUP BY film.length
-> Having length > 85
-> Order by length;
```

first_name	last_name	name	description	rental_rate	length	count(length)
PENELOPE	GUINNESS	Action	A Epic Drama of a Feminist And a Mad Scientist who must Battle a Teacher in The Canadian Rockies	0.99	86	1
VIVIEN	BERGEN	Sci-fi	A Emotional Drama of a A Shark And a Database Administrator who must Vanquish a Pioneer in Soviet Georgia	0.99	94	1
JOE	SWANK	Foreign	A Thoughtful Panorama of a Database Administrator And a Mad Scientist who must Outgun a Mad Scientist in A Jet Boat	2.99	114	1
JENNIFER	DAVIS	Classics	A Fanciful Documentary of a Frisbee And a Lumberjack who must Chase a Monkey in A Shark Tank	2.99	117	1
ZERO	CAGE	Horror	A Boring Epistle of a Butler And a Cat who must Fight a Pastry Chef in A MySQL Convention	0.99	126	1
JOHNNY	LOLLOBRIGIDA	Comedy	A Fast-Paced Documentary of a Pastry Chef And a Dentist who must Pursue a Forensic Psychologist in The Gulf of Mexico	2.99	130	1
KARL	BERRY	Music	A Fanciful Saga of a Hunter And a Pastry Chef who must Vanquish a Boy in Australia	0.99	136	1
UMA	WOOD	New	A Action-Packed Drama of a Dentist And a Crocodile who must Battle a Feminist in The Canadian Rockies	4.99	150	1
BETTE	NICHOLSON	Documentary	A Intrepid Panorama of a Robot And a Boy who must Escape a Sumo Wrestler in Ancient China	2.99	169	1
FRED	COSTNER	Travel	A Fast-Paced Drama of a Robot And a Composer who must Battle a Astronaut in New Orleans	2.99	180	1

10 rows in set (0.00 sec)

**Give the English explanation:**

The above query selects columns first name, last name, category name, film description, film rental rate and film length

From **table 1 (Actor)**

To combine the output result from three tables, we use **two joins** as below.

le **join <table 1 (Actor).column name(actor\_id) and table 2 (Category).column name(category\_id)>**

**Join <table 3 (Film).column name(film\_id) and table 2 (Category) .column name(category\_id)>**

Used **aggregate operator (count)** (under each category name only 1 record is displayed, so count is1)

Used **having clause (Having length > 85)**