

MySQL Task sakila

Installation

Download from <https://downloads.mysql.com/docs/sakila-db.zip>

```
mysql> use sakila
Database changed
mysql> select database();
+-----+
| database() |
+-----+
| sakila     |
+-----+
1 row in set (0.00 sec)

mysql> desc actor;
+-----+
| Field      | Type                | Null | Key | Default | Extra           |
+-----+
| actor_id   | smallint unsigned   | NO   | PRI | NULL    | auto_increment |
| first_name | varchar(45)         | NO   |     | NULL    |                 |
| last_name  | varchar(45)         | NO   | MUL | NULL    |                 |
| last_update | timestamp           | NO   |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED on update CURRENT_TIMESTAMP |
+-----+
4 rows in set (2.32 sec)

mysql> show full tables;
+-----+
| Tables_in_sakila | Table_type |
+-----+
| actor             | BASE TABLE |
| actor_info        | VIEW        |
| address           | BASE TABLE |
| category          | BASE TABLE |
| city              | BASE TABLE |
| country           | BASE TABLE |
| customer           | BASE TABLE |
| customer_list     | VIEW        |
| film              | BASE TABLE |
| film_actor        | BASE TABLE |
| film_category     | BASE TABLE |
| film_list         | VIEW        |
| film_text         | BASE TABLE |
| inventory         | BASE TABLE |
| language          | BASE TABLE |
| nicer_but_slower_film_list | VIEW        |
| payment           | BASE TABLE |
| rental            | BASE TABLE |
| sales_by_film_category | VIEW        |
| sales_by_store    | VIEW        |
| staff             | BASE TABLE |
| staff_list        | VIEW        |
| store             | BASE TABLE |
+-----+
23 rows in set (0.10 sec)
```

```
mysql> select count(*) from film;
+-----+
| count(*) |
+-----+
| 1000     |
+-----+
1 row in set (3.14 sec)

mysql> select count(*) from film_text;
+-----+
| count(*) |
+-----+
| 1000     |
+-----+
1 row in set (0.38 sec)

mysql> select count * from film_text limit 5;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'from film_text limit 5' at line 1
mysql> select * from film_text limit 5;
+-----+
| file_id | title | description |
+-----+
| 1 | ACADEMY DINOSAUR | A Epic Drama of a Feminist And a Mad Scientist who must Battle a Teacher in The Canadian Rockies |
| 2 | ACE GOLDFINGER | A Astounding Epistle of a Database Administrator And a Explorer who must Find a Car in Ancient China |
| 3 | ADAPTATION HOLES | A Astounding Reflection of a Lumberjack And a Car who must Sink a Lumberjack in A Baloon Factory |
| 4 | AFFAIR PREJUDICE | A Fanciful Documentary of a Frisbee And a Lumberjack who must Chase a Monkey in A Shark Tank |
| 5 | AFRICAN EGG | A Fast-Paced Documentary of a Pastry Chef And a Dentist who must Pursue a Forensic Psychologist in The Gulf of Mexico |
+-----+
5 rows in set (0.00 sec)

mysql> select * from film limit 5;
+-----+
| file_id | title | description |
+-----+
| 1 | ACADEMY DINOSAUR | A Epic Drama of a Feminist And a Mad Scientist who must Battle a Teacher in The Canadian Rockies |
| 2 | ACE GOLDFINGER | A Astounding Epistle of a Database Administrator And a Explorer who must Find a Car in Ancient China |
| 3 | ADAPTATION HOLES | A Astounding Reflection of a Lumberjack And a Car who must Sink a Lumberjack in A Baloon Factory |
| 4 | AFFAIR PREJUDICE | A Fanciful Documentary of a Frisbee And a Lumberjack who must Chase a Monkey in A Shark Tank |
| 5 | AFRICAN EGG | A Fast-Paced Documentary of a Pastry Chef And a Dentist who must Pursue a Forensic Psychologist in The Gulf of Mexico |
+-----+
5 rows in set (0.00 sec)
```

1. Display the first and last name of each actor in a single column in upper case letters in alphabetic order. Name the column Actor Name.

```
Command Prompt - mysql -u root -p

mysql> select concat(upper(first_name), ' ', upper(last_name)) AS "Actor Name"
-> from actor
-> order by "Actor Name";
```

Actor Name
PENELOPE GUINESS
NICK MAHLBERG
ED CHASE
JENNIFER DAVIS
JOHNNY LULLOBRIGIDA
BETTE NICHOLSON
GRACE MOSTEL
MATTHEW JOHANSSON
JOE SWANK
CHRISTIAN GABLE
ZERO CAGE
KARL BERRY
UPA WOOD
VIVIEN BERGEN
CUBA OLIVIER
FRED COSTNER
HELEN VOIGHT
DAN TORN
BOB FAKETT
LUCILLE TRACY
KIRSTEN PALTROW
ELVIS MARX
SANDRA KILMER
CAMERON STREEP
KEVIN BLOOM
RIP CRAWFORD
JULIA MCQUEEN
WOODY HOFFMAN
ALEC MAYNE
SANDRA PECK
SISSY SOBIESKI
TIM HACKMAN
MILLA PECK
AUDREY OLIVIER
JUDY DEAN
BURT DUKAKIS
VAL BOLGER
TONI MCKELLEN
GOLDIE BROODY
JOHNNY CAGE
JODIE DEGENERES
TONI MIRANDA
KIRK TOWNICH

Activate Windows
Go to Settings to activate Windows.

```
Command Prompt - mysql -u root -p
```

KIRK TOWNICH
NICK STALLONE
REESE KILMER
PARKER GOLDBERG
JULIA BARRYMORE
FRANKES DAY-LEWIS
ANNE CROWIN
NATALIE HOPKINS
GARY PHOENIX
CARMEN HUNT
RENA TEMPLE
PENELOPE PINKETT
FAY KILMER
DAN HARRIS
JUDE CRUISE
CHRISTIAN AKROYD
DUSTIN TAUTOU
HENRY BERRY
CHRISTIAN NEESON
JAYNE NEESON
CAMERON MRAY
RAY JOHANSSON
ANGELA HUDSON
MARY TANDY
JESSICA BAILEY
RIP WINSLET
KENNETH PALTROW
MICHELLE MCCONAUGHEY
ADAM GRANT
SEAN MCELLEANS
GARY PENN
MILLA KETTEL
BURT POSEY
ANGELINA ASTAIRE
GARY MCCONAUGHEY
GROUCHO SINATRA
MAE HOFFMAN
RALPH CRUZ
SCARLETT DAWON
WOODY JOLIE
BEN WILLIS
JAMES PITT
MICHAEL ZELLWEGER
GREG CHAPLIN
SPENKER PECK
KENNETH PESCI
CHARLIZE DENCH
SEAN GUINNESS
CHRISTOPHER BERRY
KIRSTEN AKROYD

Activate Windows
Go to Settings to activate Windows.

```
Command Prompt - mysql -u root -p

KIRSTEN AKROYD
ELLEN PRESLEY
KENNETH TORN
DARYL WAHLBERG
GENE WILLIS
PEG HAWKE
CHRIS BRIDGES
TIM POSTEL
SPENCER DEPP
SUSAN DAVIS
WALTER TORN
MATTHEW LEIGH
PENELOPE GRONYI
SIDNEY CROME
GROUCHO DUNST
GINA DEGENERES
WARREN NOLTE
SYLVESTER DERN
SUSAN DAVIS
CAMERON ZELLMER
RUSSELL BACALL
MORGAN HOPKINS
MORGAN MCDORMAND
HARRISON BALE
DAN STREEP
RENEE TRACY
CUBA ALLEN
WARREN JACKMAN
PENELOPE MONROE
LIZA BERGMAN
SALMA NOLTE
JULIANNE DENCH
SCARLETT BEHNING
ALBERT NOLTE
FRANKIE TOMEI
KEVIN GARLAND
CATE MCQUEEN
DARYL CRAWFORD
GRETA KETTEL
JANE JACKMAN
ADAM HOPPER
RICHARD PENN
GENE HOPKINS
KITA REYNOLDS
ED HANSFIELD
MORGAN WILLIAMS
LUCILLE DEE
EMAN GOODING
WHOOPI HURT
CATE HARRIS
```

```
Command Prompt - mysql -u root -p

CATE HARRIS
JADA POKER
RIVER DEAN
ANGELA WITHERSPOON
KIM ALLEN
ALBERT JOHANSSON
FAV KINSLET
EMILY DEE
RUSSELL TEMPLE
JAYNE NOLTE
GEOFFREY HESTON
BEN HARRIS
MINNIE KILMER
MERYL GIBSON
IAN TANDY
FAV WOOD
GRETA MALDEN
VIVIAN BASINGER
LAURA BRODY
CHRIS DEPP
HARVEY HOPE
OPRAH KILMER
CHRISTOPHER WEST
HUMPHREY WILLIS
AL GARLAND
NICK DEGENERES
LAURENCE BULLOCK
WILL WILSON
KENNETH HOFFMAN
RENA HOPPER
OLYMPIA PFETTER
GROUCHO WILLIAMS
ALAN DREYFUSS
MICHAEL BENING
WILLIAM JACKMAN
JON CHASE
GENE MCKELLEN
LISA MONROE
ED GUINNESS
JEFF SILVERSTONE
MATTHEW CARREY
DEBBIE AKROYD
RUSSELL CLOSE
HUMPHREY GARLAND
MICHAEL BOLGER
JULIA ZELLMER
RENEE BALL
ROCK DUKAKIS
CUBA BIRCH
AUDREY BATLEY
```

```
Command Prompt - mysql -u root -p

AUDREY BATLEY
GREGORY GOODING
JOHN SUVARI
BURT TEMPLE
MERYL ALLEN
JAYNE SILVERSTONE
BELA WALKER
REESE WEST
MARY KETTEL
JULIA FAWCETT
THORA TEMPLE

200 rows in set (0.41 sec)

mysql>
```

2. Find all actors whose last name contain the letters GEN:

```
Command Prompt - mysql -u root -p

mysql>
mysql>
mysql> select * from actor where last_name like '%GEN%';
+-----+-----+-----+-----+
| actor_id | first_name | last_name | last_update |
+-----+-----+-----+-----+
| 14 | VIVIAN | BERGEN | 2006-02-15 04:34:33 |
| 41 | JODIE | DEGENERES | 2006-02-15 04:34:33 |
| 107 | GINA | DEGENERES | 2006-02-15 04:34:33 |
| 166 | NICK | DEGENERES | 2006-02-15 04:34:33 |
+-----+-----+-----+-----+
4 rows in set (0.56 sec)

mysql>
```

3. Using IN, display the country_id and country columns of the following countries:
Afghanistan, Bangladesh, and China:

```
Command Prompt - mysql -u root -p
mysql> select country_id, country
-> from country
-> where country in(
-> 'Afghanistan', 'Bangladesh', 'China');
+-----+-----+
| country_id | country |
+-----+-----+
| 1          | Afghanistan |
| 12         | Bangladesh |
| 23         | China      |
+-----+-----+
3 rows in set (3.02 sec)

mysql>
```

4. List the last names of actors, as well as how many actors have that last name.

```
Select Command Prompt - mysql -u root -p
mysql>
mysql>
mysql> select last_name, count(*) as actor_count
-> from actor
-> group by last_name
-> order by actor_count desc, last_name;
+-----+-----+
| last_name | actor_count |
+-----+-----+
| KILMER    | 5           |
| NOLTE     | 4           |
| TEMPLE    | 4           |
| AKROYD    | 3           |
| ALLEN     | 3           |
| BERRY     | 3           |
| DAVIS     | 3           |
| DEGENERES | 3           |
| GARLAND   | 3           |
| GUTNESS   | 3           |
| HARRIS    | 3           |
| HOFFMAN   | 3           |
| HOPKINS   | 3           |
| JOHANSSON | 3           |
| KETTEL    | 3           |
| PECK      | 3           |
| TORN      | 3           |
| WILLIAMS  | 3           |
| WILLIS    | 3           |
| ZELLMER   | 3           |
| BAILEY    | 2           |
| BENING    | 2           |
| BOLGER    | 2           |
| BROOKE    | 2           |
| CAGE      | 2           |
| CHASE     | 2           |
| CRAWFORD  | 2           |
| CROWNN    | 2           |
| DEAN      | 2           |
| DEE       | 2           |
| DENCH     | 2           |
| DEPP      | 2           |
| DUKAKIS   | 2           |
| FAWCETT   | 2           |
| GOODING   | 2           |
| HACKMAN   | 2           |
| HOPPER    | 2           |
| JACKMAN   | 2           |
| MCCONAUGHEY | 2         |
| MCKELLEN  | 2           |
| MCKEEN    | 2           |
| MONROE    | 2           |
| MOSTEL    | 2           |
| NEESON    | 2           |
| OLIVIER   | 2           |
| PALTROW   | 2           |
| PENN      | 2           |
| SILVERSTONE | 2         |
| STREEP    | 2           |
| TANDY     | 2           |
| TRACY     | 2           |
| WAHLBERG  | 2           |
| WEST      | 2           |
| WINSLET   | 2           |
| WOOD      | 2           |
| ASTAIRE   | 1           |
| BACALL    | 1           |
| BALE      | 1           |
| BALL      | 1           |
| BARRYMORE | 1           |
| BASTINGER | 1           |
| BERGEN    | 1           |
| BERGMAN   | 1           |
| BIRCH     | 1           |
| BLOOM     | 1           |
| BRIDGES   | 1           |
| BULLOCK   | 1           |
| CARREY    | 1           |
| CHAPLIN   | 1           |
| CLOSE     | 1           |
| COSTNER   | 1           |
| CROME     | 1           |
| CRUISE    | 1           |
| CRUZ      | 1           |
| DAMON     | 1           |
| DAY-LEWIS | 1           |
| DERN      | 1           |
| DREYFUSS  | 1           |
| DUNST     | 1           |
| GABLE     | 1           |
| GIBSON    | 1           |
| GOLDBERG  | 1           |
| GRANT     | 1           |
| HAWKE     | 1           |
| HESTON    | 1           |
| HOPE      | 1           |
| HUDSON    | 1           |
| HUNT      | 1           |
| HURT      | 1           |
| JOLIE     | 1           |
+-----+-----+

Activate Windows
Go to Settings to activate Windows.

16:46
```

Select Command Prompt - mysql -u root -p

JOLIE	1
KOVACH	1
LEIGH	1
LOLLORRIGIDA	1
MALDEN	1
MANSFIELD	1
MARX	1
MCDONNAND	1
MIRANDA	1
NICHOLSON	1
PESCI	1
PEIFFER	1
PHENIX	1
PINKETT	1
PITT	1
POSEY	1
PRESLEY	1
REYNOLDS	1
RYDER	1
SINATRA	1
SOBLESKI	1
STALLONE	1
SUARI	1
SHANK	1
TAUTOU	1
TOHEI	1
VOIGHT	1
WALKEN	1
WAYNE	1
WILSON	1
WETHERSPOON	1
WRAY	1

121 rows in set (0.12 sec)

mysql>

- List last names of actors and the number of actors who have that last name, but only for names that are shared by at least two actors

Command Prompt - mysql -u root -p

```
mysql> SELECT last_name, COUNT(*) as actor_count
--> FROM actor
--> GROUP BY last_name
--> HAVING actor_count >= 2
--> ORDER BY actor_count DESC, last_name;
```

last_name	actor_count
KILMER	5
NOLTE	4
TEPPLE	4
ANDRUD	3
ALLEN	3
BERRY	3
DAVIS	3
DEGENERES	3
GARLAND	3
GUINNESS	3
HARRIS	3
HOFFMAN	3
HOPKINS	3
JOHANSSON	3
KEITEL	3
PECK	3
TORN	3
WILLIAMS	3
WILLIS	3
ZELLMIEGER	3
BAILEY	2
BENING	2
BOLGER	2
BRODY	2
CAGE	2
CHASE	2
CRAWFORD	2
CROWNN	2
DEAN	2
DEE	2
DENCH	2
DEPP	2
DUKAKIS	2
FAMKETT	2
GOODING	2
HACKMAN	2
HOPPER	2
JACKMAN	2
MC CONAUGHEY	2
MC KELLLEN	2
MCQUEEN	2
MCQUEEN	2
MONROE	2
POSTEL	2
NEESON	2
OLIVIER	2
PALTRON	2
PENN	2
SILVERSTONE	2
STREEP	2
TANDY	2
TRACY	2
WAHLBERG	2
WEST	2
WINSLET	2
WOOD	2

55 rows in set (0.05 sec)

mysql>

- The actor HARPO WILLIAMS was accidentally entered in the actor table as GROUCHO WILLIAMS. Write a query to fix the record.

```

mysql> select * from actor where last_name = 'WILLIAMS';
+-----+-----+-----+-----+
| actor_id | first_name | last_name | last_update |
+-----+-----+-----+-----+
| 72 | SEAN | WILLIAMS | 2006-02-15 04:34:33 |
| 137 | MORGAN | WILLIAMS | 2006-02-15 04:34:33 |
| 172 | GROUCHO | WILLIAMS | 2024-03-06 17:02:47 |
+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> UPDATE actor
-> SET first_name = 'HARPO'
-> WHERE first_name = 'GROUCHO' AND last_name = 'WILLIAMS';
Query OK, 1 row affected (0.18 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> select * from actor where last_name = 'WILLIAMS';
+-----+-----+-----+-----+
| actor_id | first_name | last_name | last_update |
+-----+-----+-----+-----+
| 72 | SEAN | WILLIAMS | 2006-02-15 04:34:33 |
| 137 | MORGAN | WILLIAMS | 2006-02-15 04:34:33 |
| 172 | HARPO | WILLIAMS | 2024-03-06 17:06:03 |
+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql>

```

- Use JOIN to display the first and last names, as well as the address, of each staff member. Use the tables staff and address:

```

mysql> select staff.first_name, staff.last_name, address.address
-> from staff join address on staff.address_id = address.address_id;
+-----+-----+-----+
| first_name | last_name | address |
+-----+-----+-----+
| Mike | Hillyer | 23 Workhaven Lane |
| Jon | Stephens | 1411 Lillydale Drive |
+-----+-----+-----+
2 rows in set (0.81 sec)

mysql>
mysql> select s.first_name as 'First Name', s.last_name as 'Last Name', a.address as 'Address'
-> from staff as s
-> join address as a
-> ON a.address_id = s.address_id;
+-----+-----+-----+
| First Name | Last Name | Address |
+-----+-----+-----+
| Mike | Hillyer | 23 Workhaven Lane |
| Jon | Stephens | 1411 Lillydale Drive |
+-----+-----+-----+
2 rows in set (0.00 sec)

mysql>

```

- List each film and the number of actors who are listed for that film. Use tables film_actor and film. Use inner join.

```

mysql> SELECT film.film_id, film.title, COUNT(film_actor.actor_id) as actor_count
-> FROM film
-> INNER JOIN film_actor ON film.film_id = film_actor.film_id
-> GROUP BY film.film_id, film.title
-> ORDER BY actor_count DESC, film.title limit 50;
+-----+-----+-----+
| film_id | title | actor_count |
+-----+-----+-----+
| 588 | LAMB'S CINCINATTI | 15 |
| 87 | BOONDOCK BALLROOM | 13 |
| 146 | CHITTY LOCK | 13 |
| 188 | CRAZY HOME | 13 |
| 249 | DRACULA CRYSTAL | 13 |
| 686 | MURPHY CREATURES | 13 |
| 714 | RANDOM GO | 13 |
| 34 | ARABIA DOGMA | 12 |
| 418 | HELIFIGHTERS STERRA | 12 |
| 517 | LESSON CLEDPATRA | 12 |
| 529 | LONELY ELEPHANT | 12 |
| 802 | SKY MIRACLE | 12 |
| 892 | TITANIC BOONDOCK | 12 |
| 312 | FIDDLER LOST | 11 |
| 342 | FUGITIVE MAGUIRE | 11 |
| 420 | HOLES BRANNIGAN | 11 |
| 453 | IMAGE PRINCESS | 11 |
| 463 | INSTINCT AIRPORT | 11 |
| 553 | PAKER GABLES | 11 |
| 561 | MASK PEACH | 11 |
| 649 | OZ LIAISONS | 11 |
| 680 | PINOCCHIO SIMON | 11 |
| 732 | RINGS HEARTBREAKERS | 11 |
| 827 | SPICE SORORITY | 11 |
| 830 | SPIRIT FLINTSTONES | 11 |
| 858 | SUBMARINE BED | 11 |
| 880 | TELEMARK HEARTBREAKERS | 11 |
| 1 | ACADREY DINOSAUR | 10 |
| 67 | BERETS AGENT | 10 |
| 144 | CHINATOWN GLADIATOR | 10 |
| 164 | COAST RAINBOW | 10 |
| 194 | CROW GRASS | 10 |
| 288 | DARES PLUTO | 10 |
| 410 | HEAVEN FREEDOM | 10 |
| 431 | HOOSIERS BIRDCAGE | 10 |
| 458 | INDIAN LOVE | 10 |
| 473 | JACKET FRISCO | 10 |
| 540 | LUCKY FLYING | 10 |
| 636 | OLEANDER CLUE | 10 |
| 688 | POLISH BROOKLYN | 10 |
| 694 | PREJUDICE OLEANDER | 10 |
+-----+-----+-----+

```

- How many copies of the film Hunchback Impossible exist in the inventory system?

```
Command Prompt - mysql -u root -p
mysql>
mysql> SELECT title, COUNT(inventory_id)
  -> FROM film f
  -> INNER JOIN inventory i
  -> ON f.film_id = i.film_id
  -> WHERE title = "Hunchback Impossible";
+-----+-----+
| title          | COUNT(inventory_id) |
+-----+-----+
| HUNCHBACK IMPOSSIBLE | 6 |
+-----+-----+
1 row in set (8.11 sec)

mysql>
```

10. Using the tables payment and customer and the JOIN command, list the total paid by each customer. List the customers alphabetically by last name

```
Command Prompt - mysql -u root -p
mysql>
mysql> SELECT customer.last_name, customer.first_name, SUM(payment.amount) AS total_paid
  -> FROM customer
  -> JOIN payment ON customer.customer_id = payment.customer_id
  -> GROUP BY customer.customer_id
  -> ORDER BY customer.last_name, customer.first_name limit 20;
+-----+-----+-----+
| last_name | first_name | total_paid |
+-----+-----+-----+
| ADNEY     | RAFAEL    | 97.79      |
| ADAM      | NATHANIEL | 133.72     |
| ADAMS     | KATHLEEN  | 92.73      |
| ALEXANDER | DIANA     | 105.73     |
| ALLARD    | GORDON    | 160.68     |
| ALLEN     | SHERLEY   | 126.69     |
| ALVAREZ   | CHARLENE  | 114.73     |
| ANDERSON  | LISA      | 106.76     |
| ANDREW    | JOSE      | 96.75      |
| ANDREWS   | IDA       | 76.77      |
| AQUINO    | OSCAR     | 99.80      |
| ARCE      | HARRY     | 157.65     |
| ARCHULETA | JORDAN    | 132.70     |
| ARMSTRONG | MELANIE   | 92.75      |
| ARNOLD    | BEATRICE  | 119.74     |
| ARSENAULT | KEITH     | 134.73     |
| ARTIS     | CARL      | 106.77     |
| ASHCRAFT  | DARRYL    | 76.77      |
| ASHER     | TYRONE    | 112.76     |
| AUSTIN    | ALMA      | 151.65     |
+-----+-----+-----+
20 rows in set (0.10 sec)
```

11. The music of Queen and Kris Kristofferson have seen an unlikely resurgence. As an unintended consequence, films starting with the letters **K** and **Q** have also soared in popularity. Use subqueries to display the titles of movies starting with the letters **K** and **Q** whose language is English.

```
Command Prompt - mysql -u root -p
mysql>
mysql>
mysql> SELECT title FROM film
  -> WHERE language_id in
  -> (SELECT language_id
  -> FROM language
  -> WHERE name = "English" )
  -> AND (title LIKE "K%" OR (title LIKE "Q%"));
+-----+
| title          |
+-----+
| KANE EXORCIST  |
| KARATE MOON    |
| KENTUCKIAN GIANT |
| KICK SAVANNAH  |
| KILL BROTHERHOOD |
| KILLER INNOCENT |
| KING EVOLUTION |
| KISS GLORY     |
| KISSING DOLLS  |
| KNOCK WARLOCK  |
| KRAMER CHOCOLATE |
| KWAI HOMEWARD  |
| QUEEN LUKE     |
| QUEST MUSSOLINI |
| QUILLS BULL    |
+-----+
15 rows in set (3.87 sec)

mysql>
```

12. Use subqueries to display all actors who appear in the film *Alone Trip*.

```
Command Prompt - mysql -u root -p

mysql> SELECT last_name, first_name
-> FROM actor
-> WHERE actor_id in
-> (SELECT actor_id FROM film_actor
-> WHERE film_id in
-> (SELECT film_id FROM film
-> WHERE title = "Along Trip"));
+-----+
| last_name | first_name |
+-----+
| CHASE     | ED         |
| BERRY     | KARL       |
| WOOD      | UMA        |
| JOLIE     | WOODY      |
| DEPP      | SPENCER    |
| DEPP      | CHRIS      |
| BULLOCK   | LAURENCE   |
| BALL      | RENEE      |
+-----+
8 rows in set (0.24 sec)

mysql>
```

13. You want to run an email marketing campaign in Canada, for which you will need the names and email addresses of all Canadian customers. Use joins to retrieve this information.

```
Command Prompt - mysql -u root -p

mysql>
mysql>
mysql> SELECT country, last_name, first_name, email
-> FROM country c
-> LEFT JOIN customer cu
-> ON c.country_id = cu.customer_id
-> WHERE country = "Canada";
+-----+
| country | last_name | first_name | email |
+-----+
| Canada  | ROBINSON  | SHARON    | SHARON.ROBINSON@sakilacustomer.org |
+-----+
1 row in set (0.77 sec)

mysql>
```

14. Sales have been lagging among young families, and you wish to target all family movies for a promotion. Identify all movies categorized as famiy films.

```
Command Prompt - mysql -u root -p

mysql>
mysql> SELECT title, category
-> FROM film_list
-> WHERE category = "Family";
+-----+
| title | category |
+-----+
| AFRICAN EGG | Family |
| APACHE DIVINE | Family |
| ATLANTIS CAUSE | Family |
| BAKED CLEOPATRA | Family |
| BANG KMAI | Family |
| BEDAZZLED MARRIED | Family |
| BILKO ANONYMOUS | Family |
| BLANKET BEVERLY | Family |
| BLOOD ARGONAUTS | Family |
| BLUES INSTINCT | Family |
| BRAVEHEART HUMAN | Family |
| CHASTING FIGHT | Family |
| CHISUM BEHAVIOR | Family |
| CHOCOLAT HARRY | Family |
| CONFUSED CANDLES | Family |
| CONVERSATION DOWNHILL | Family |
| DATE SPEED | Family |
| DINOSAUR SECRETARY | Family |
| DUMBO LUST | Family |
| EARRING INSTINCT | Family |
| EFFECT GLADIATOR | Family |
| FELD FROGREN | Family |
| FINDING ANACONDA | Family |
| GABLES METROPOLIS | Family |
| GANDHI KMAI | Family |
| GLADIATOR WESTWARD | Family |
| GREASE YOUTH | Family |
| HALF OUTFIELD | Family |
| HOCUS FRIDA | Family |
| HOMICIDE PEACH | Family |
| HOUSE DYNAMITE | Family |
| HUNTING MUSKETEERS | Family |
| INDIAN LOVE | Family |
| JASON TRAP | Family |
| JEDI BENEATH | Family |
| KILLER INNOCENT | Family |
| KING EVOLUTION | Family |
| LOLITA WORLD | Family |
| LOUISIANA HARRY | Family |
| MAGUIRE APACHE | Family |
| MANCHURIAN CURTAIN | Family |
| MOVIE SHAKESPEARE | Family |
+-----+

Activate Windows
Go to Settings to activate Windows.
```



```

Command Prompt - mysql -u root -p

| MOVIE SHAKESPEARE | Family |
| MUSIC BOONDOCK | Family |
| NATURAL STOCK | Family |
| NETWORK PEAK | Family |
| ODDS BOOGIE | Family |
| OPPOSITE NECKLACE | Family |
| PILOT HOOSIERS | Family |
| PITTSBURGH HUNKBACK | Family |
| PRESIDENT BANG | Family |
| PRIX UNDEFEATED | Family |
| RAGE GAMES | Family |
| RANGE POORMALKER | Family |
| REMEMBER DIARY | Family |
| RESURRECTION SILVERADO | Family |
| ROBBERY BRIGHT | Family |
| RUSH GOODFELLAS | Family |
| SECRETS PARADISE | Family |
| SENSIBILITY REAR | Family |
| SIEGE MADRE | Family |
| SILMS DUCK | Family |
| SOUP WISDOM | Family |
| SPARTACUS CHEAPER | Family |
| SPINAL ROCKY | Family |
| SPLASH GUMP | Family |
| SUNSET RACER | Family |
| SUPER MYOTING | Family |
| VIRTUAL SPOTTERS | Family |
| WILLOW TRACY | Family |
-----
89 rows in set (2.63 sec)

mysql>

```

15. Create a Stored procedure to get the count of films in the input category (IN category_name, OUT count)

```

Command Prompt - mysql -u root -p

mysql> DELIMITER //
mysql>
mysql> CREATE PROCEDURE GetFilmCountByCategory(
  -> IN category_name VARCHAR(255),
  -> OUT film_count INT
  -> )
  -> BEGIN
  -> SELECT COUNT(*) INTO film_count
  -> FROM films
  -> WHERE category = category_name;
  -> END //
Query OK, 0 rows affected (2.00 sec)

mysql>
mysql> DELIMITER ;
mysql>

```

16. Display the most frequently rented movies in descending order.

```

Command Prompt - mysql -u root -p

mysql> SELECT f.title, COUNT(f.title) as rentals from film f
  -> JOIN
  -> (SELECT r.rental_id, i.film_id FROM rental r
  -> JOIN
  -> inventory i ON i.inventory_id = r.inventory_id) a
  -> ON a.film_id = f.film_id GROUP BY f.title ORDER BY rentals DESC limit 30;
-----
| title | rentals |
-----
| BUCKET BROTHERHOOD | 34 |
| ROCKETEER MOTHER | 33 |
| FORWARD TEMPLE | 32 |
| SCALWAG DUCK | 32 |
| JUGGLER HARDLY | 32 |
| RIDGEMONT SUBMARINE | 32 |
| GRIT CLOCKWORK | 32 |
| GOODFELLAS SALUTE | 31 |
| TIMBERLAND SKY | 31 |
| ROBBERS TOON | 31 |
| ZORRO ARK | 31 |
| NETWORK PEAK | 31 |
| HOBBIT ALIEN | 31 |
| WIFE TURN | 31 |
| APACHE DIVINE | 31 |
| RUSH GOODFELLAS | 31 |
| SUSPECTS QUILLS | 30 |
| RUGRATS SHAKESPEARE | 30 |
| GRAFFITI LOVE | 30 |
| MARRIED GO | 30 |
| MUSCLE BRIGHT | 30 |
| IDOLS SNATCHERS | 30 |
| PASSARE USUAL | 30 |
| ENGLISH RULMORTH | 30 |
| DOGMA FAMILY | 30 |
| CAT CONEHEADS | 30 |
| SHOCK CABIN | 30 |
| HARRY IDHO | 30 |
| FROST HEAD | 30 |
| BUTTERFLY CHOCOLAT | 30 |
-----
30 rows in set (0.06 sec)

mysql>

```

17. Write a query to display for each store its store ID, city, and country.

```
mysql> SELECT d.store_id, d.address, d.city, country.country from country
-> JOIN
-> (SELECT b.store_id, b.address, c.city, c.country_id FROM city c
-> JOIN
-> (SELECT s.store_id, a.address, a.city_id from store s
-> JOIN address a ON s.address_id = a.address_id) b ON (b.city_id = c.city_id)
-> ) d ON (d.country_id = country.country_id);
+-----+-----+-----+-----+
| store_id | address          | city       | country |
+-----+-----+-----+-----+
| 1        | 47 MySakila Drive | Lethbridge | Canada  |
| 2        | 28 MySQL Boulevard | Woodridge  | Australia |
+-----+-----+-----+-----+
2 rows in set (8.87 sec)

mysql>
```

18. List the genres and its gross revenue.

```
mysql> SELECT
-> c.name AS genre,
-> SUM(p.amount) AS gross_revenue
-> FROM
-> category c
-> JOIN
-> film_category fc ON c.category_id = fc.category_id
-> JOIN
-> film f ON fc.film_id = f.film_id
-> JOIN
-> inventory i ON f.film_id = i.film_id
-> JOIN
-> rental r ON i.inventory_id = r.inventory_id
-> JOIN
-> payment p ON r.rental_id = p.rental_id
-> GROUP BY
-> c.name
-> ORDER BY
-> gross_revenue DESC;
+-----+-----+
| genre | gross_revenue |
+-----+-----+
| Sports | 5314.21 |
| Sci-Fi | 4756.98 |
| Animation | 4656.30 |
| Drama | 4587.39 |
| Comedy | 4383.58 |
| Action | 4375.85 |
| New | 4351.62 |
| Games | 4281.23 |
| Foreign | 4270.67 |
| Family | 4226.07 |
| Documentary | 4217.52 |
| Horror | 3722.54 |
| Children | 3655.55 |
| Classics | 3639.59 |
| Travel | 3549.64 |
| Music | 3417.72 |
+-----+-----+
16 rows in set (0.27 sec)
```

19. Create a View for the above query(18)

```
mysql> ^C
mysql> CREATE VIEW genre_gross_revenue_view AS
-> SELECT
-> c.name AS genre,
-> SUM(p.amount) AS gross_revenue
-> FROM
-> category c
-> JOIN
-> film_category fc ON c.category_id = fc.category_id
-> JOIN
-> film f ON fc.film_id = f.film_id
-> JOIN
-> inventory i ON f.film_id = i.film_id
-> JOIN
-> rental r ON i.inventory_id = r.inventory_id
-> JOIN
-> payment p ON r.rental_id = p.rental_id
-> GROUP BY
-> c.name
-> ORDER BY
-> gross_revenue DESC;
Query OK, 0 rows affected (0.37 sec)

mysql> SELECT * FROM genre_gross_revenue_view;
+-----+-----+
| genre | gross_revenue |
+-----+-----+
| Sports | 5314.21 |
| Sci-Fi | 4756.98 |
| Animation | 4656.30 |
| Drama | 4587.39 |
| Comedy | 4383.58 |
| Action | 4375.85 |
| New | 4351.62 |
| Games | 4281.23 |
| Foreign | 4270.67 |
| Family | 4226.07 |
| Documentary | 4217.52 |
| Horror | 3722.54 |
| Children | 3655.55 |
| Classics | 3639.59 |
| Travel | 3549.64 |
| Music | 3417.72 |
+-----+-----+
16 rows in set (0.54 sec)

mysql>
```

Activate Windows
Go to Settings to activate Windows.

20. Select top 5 genres in gross revenue view.

```
Command Prompt - mysql -u root -p
mysql>
mysql>
mysql> SELECT *
    -> FROM genre_gross_revenue_view
    -> ORDER BY gross_revenue DESC
    -> LIMIT 5;
+-----+-----+
| genre | gross_revenue |
+-----+-----+
| Sports | 5314.21 |
| Sci-Fi | 4756.98 |
| Animation | 4656.30 |
| Drama | 4587.39 |
| Comedy | 4383.58 |
+-----+-----+
5 rows in set (0.26 sec)

mysql> _
```