Description of Data

For the purpose of practicing writing SQL queries to master the fundamentals, the sakila sample database was used for this exercise. Queries were written and executed inside the MySQL workbench.

An example of the tables and keys is below:

actor

primary key: actor_id

address

primary key: address_id

foreign key: city_id

film

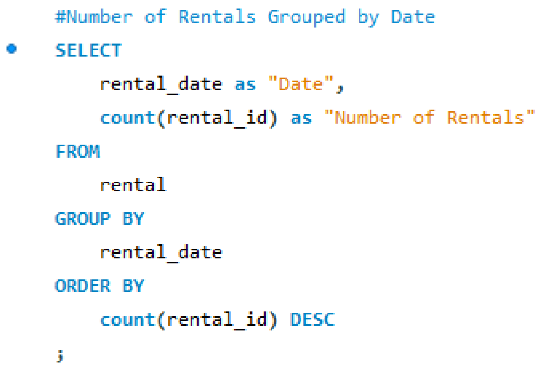
Primary key: film_id

Foreign key: language_id

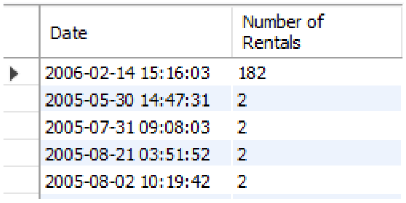
The sakila documentation can be viewed at the following address : https://dev.mysql.com/doc/sakila/en/

Limitations

Because this database features sample data some of the data is in a rather contrived format. This can be demonstrated with the below query. In a real life scenario it would be unlikely to observe 182 rentals recorded at the exact same moment. Because of the limitations of the rental dates the ability to do a robust time series analysis on rentals is not possible.



```
#Number of Rentals Grouped by Date
SELECT
    rental_date as "Date",
    count(rental_id) as "Number of Rentals"
FROM
    rental
GROUP BY
    rental_date
ORDER BY
    count(rental_id) DESC
;
```



Date	Number of Rentals
2006-02-14 15:16:03	182
2005-05-30 14:47:31	2
2005-07-31 09:08:03	2
2005-08-21 03:51:52	2
2005-08-02 10:19:42	2

This is just an example of how this sample database will not behave exactly like real-word data and is not mean to be a criticism of sakila as it is effective as a practice tool for beginners in SQL.

Methodology

The use command is first executed to access the tables in the sakila database

Example 1

The first few practice questions are simple SELCT FROM statements with one CONCAT, UPPER & IFNULL example.

Example 2

The second set of commands includes a WHERE = statement, two WHERE REGEXP examples and a WHERE IN statement used to filter results. ORDER BY is used to return results in the desired manner.

Example 3

The third group of queries uses ALTER TABLE, ADD COLUMN and DROP COLUMN statements as well as demonstrating the blob data type.

Example 4

The fourth block is several utilized GROUP BY, COUNT & HAVING to again filter results as well as using SET to demonstrate a way to alter values in a table.

Example 5

The fifth SQL code is an example of creating a schema in MySQL. Commands used include SHOW CREATE TABLE, CREATE TABLE IF NOT EXISTS, PRIMARY KEY, CONSTRAINT as well as others.

Example 6

The sixth collection of SQL requests used JOINS, MONTHS and YEARS to demonstrate pulling data from multiple tables into one result as well as SUM to find revenue.

Example 7

In the seventh example are code snippets that demonstrate using subqueries to filter and retrieve a desired result. This example does not introduce SQL commands not previously used but includes WHERE REGEXP and WHERE IN.

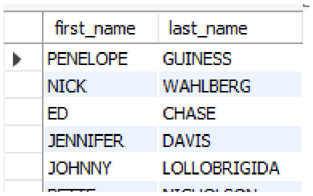
Example 8

The final few questions are an example of a fairly complicated JOIN, GROUP BY, ORDER BY & LIMIT query as well as indicating a way to save a query with CREATE VIEW for easy later refreshing of the information.

Results

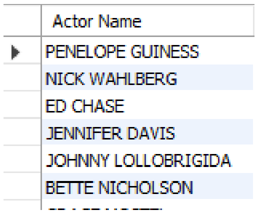
Example 1

First and last names from actor table.



first_name	last_name
PENELOPE	GUINESS
NICK	WAHLBERG
ED	CHASE
JENNIFER	DAVIS
JOHNNY	LOLLOBRIGIDA

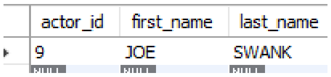
First and last names from actor table put into one table and named "Actor Name."



Actor Name
PENELOPE GUINESS
NICK WAHLBERG
ED CHASE
JENNIFER DAVIS
JOHNNY LOLLOBRIGIDA
BETTE NICHOLSON

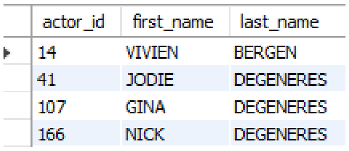
Example 2

Results from actor table with first name Joe.



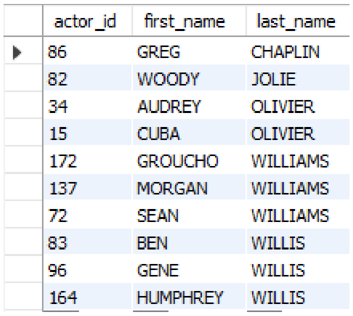
actor_id	first_name	last_name
9	JOE	SWANK

Actor ID, first name and last name from actor table for actors with GEN in their last name.



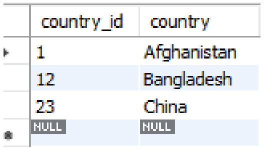
actor_id	first_name	last_name
14	VIVIEN	BERGEN
41	JODIE	DEGENERES
107	GINA	DEGENERES
166	NICK	DEGENERES

Actor ID, first name and last name from actor table for actors with LI in their last name; the result is sorted alphabetically.



actor_id	first_name	last_name
86	GREG	CHAPLIN
82	WOODY	JOLIE
34	AUDREY	OLIVIER
15	CUBA	OLIVIER
172	GROUCHO	WILLIAMS
137	MORGAN	WILLIAMS
72	SEAN	WILLIAMS
83	BEN	WILLIS
96	GENE	WILLIS
164	HUMPHREY	WILLIS

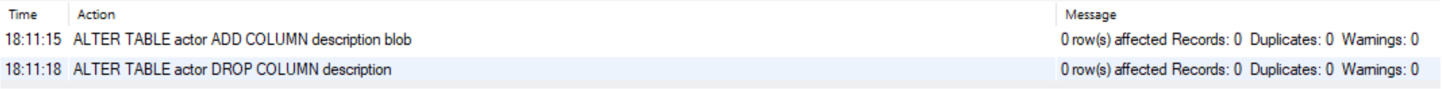
Country ID and country name results from the country table for the countries Afghanistan, Bangladesh, and China.



country_id	country
1	Afghanistan
12	Bangladesh
23	China
NULL	NULL

Example 3

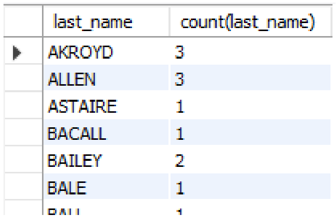
This group does not have a result table, so the action output is shown instead.



```
Time      Action                                                   Message
18:11:15  ALTER TABLE actor ADD COLUMN description blob            0 row(s) affected Records: 0  Duplicates: 0  Warnings: 0
18:11:18  ALTER TABLE actor DROP COLUMN description                0 row(s) affected Records: 0  Duplicates: 0  Warnings: 0
```

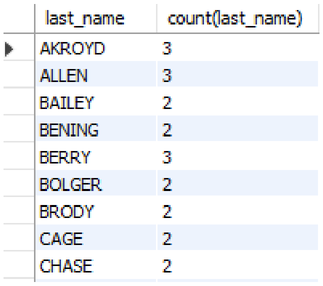
Example 4

Last name from actor table with count of actors with each.



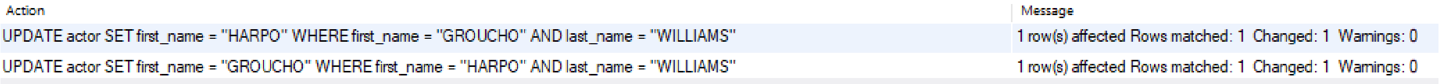
last_name	count(last_name)
AKROYD	3
ALLEN	3
ASTAIRE	1
BACALL	1
BAILEY	2
BALE	1
BALL	1

Same as prior example, but only including last names that are shared by at least two actors.



last_name	count(last_name)
AKROYD	3
ALLEN	3
BAILEY	2
BENING	2
BERRY	3
BOLGER	2
BRODY	2
CAGE	2
CHASE	2

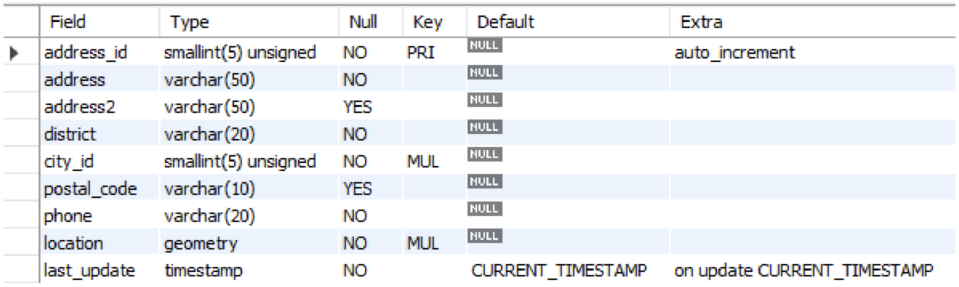
Action output for updating values in a table is shown below.



Action	Message
UPDATE actor SET first_name = "HARPO" WHERE first_name = "GROUCHO" AND last_name = "WILLIAMS"	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0
UPDATE actor SET first_name = "GROUCHO" WHERE first_name = "HARPO" AND last_name = "WILLIAMS"	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0

Example 5

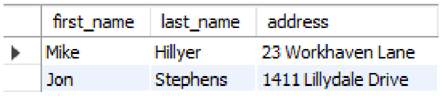
Describe result for address table is below.



Field	Type	Null	Key	Default	Extra
address_id	smallint(5) unsigned	NO	PRI	NULL	auto_increment
address	varchar(50)	NO		NULL	
address2	varchar(50)	YES		NULL	
district	varchar(20)	NO		NULL	
city_id	smallint(5) unsigned	NO	MUL	NULL	
postal_code	varchar(10)	YES		NULL	
phone	varchar(20)	NO		NULL	
location	geometry	NO	MUL	NULL	
last_update	timestamp	NO		CURRENT_TIMESTAMP	on update CURRENT_TIMESTAMP

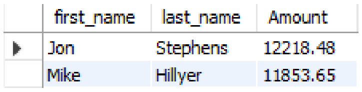
Example 6

First name, last name and addresses of staff members, taken from staff and address tables.



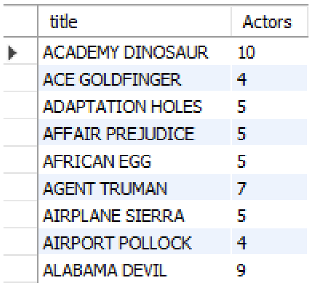
first_name	last_name	address
Mike	Hillyer	23 Workhaven Lane
Jon	Stephens	1411 Lillydale Drive

Revenue by staff member, taken from staff and payment tables.



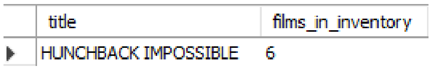
first_name	last_name	Amount
Jon	Stephens	12218.48
Mike	Hillyer	11853.65

Number of actors that appear in each movie, taken from film and film_actor tables.



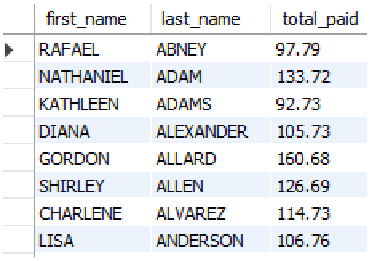
title	Actors
ACADEMY DINOSAUR	10
ACE GOLDFINGER	4
ADAPTATION HOLES	5
AFFAIR PREJUDICE	5
AFRICAN EGG	5
AGENT TRUMAN	7
AIRPLANE SIERRA	5
AIRPORT POLLOCK	4
ALABAMA DEVIL	9

Copies of film “Hunchback Impossible” in inventory, taken from film and inventory tables.



title	films_in_inventory
HUNCHBACK IMPOSSIBLE	6

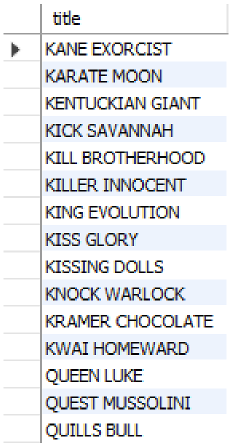
Amount paid by each customer, sorted alphabetically by last name, taken from customer and payment tables.



first_name	last_name	total_paid
RAFAEL	ABNEY	97.79
NATHANIEL	ADAM	133.72
KATHLEEN	ADAMS	92.73
DIANA	ALEXANDER	105.73
GORDON	ALLARD	160.68
SHIRLEY	ALLEN	126.69
CHARLENE	ALVAREZ	114.73
LISA	ANDERSON	106.76

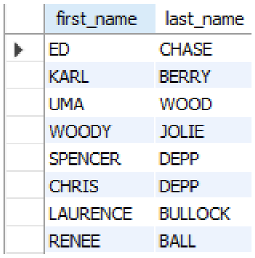
Example 7

Films in English that begin with K or Q, taken from film and language tables.



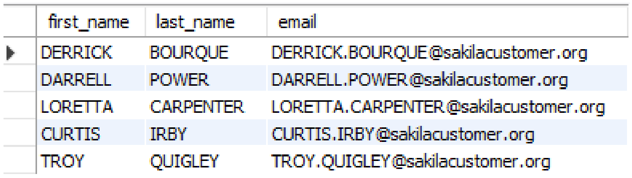
title
KANE EXORCIST
KARATE MOON
KENTUCKIAN GIANT
KICK SAVANNAH
KILL BROTHERHOOD
KILLER INNOCENT
KING EVOLUTION
KISS GLORY
KISSING DOLLS
KNOCK WARLOCK
KRAMER CHOCOLATE
KWAI HOMEWARD
QUEEN LUKE
QUEST MUSSOLINI
QUILLS BULL

Actors that appear in the film “Alone Trip”, taken from actor, film_actor and film tables.



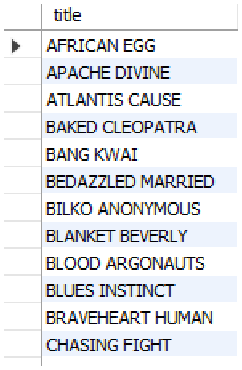
first_name	last_name
ED	CHASE
KARL	BERRY
UMA	WOOD
WOODY	JOLIE
SPENCER	DEPP
CHRIS	DEPP
LAURENCE	BULLOCK
RENEE	BALL

First and last name and email of customer in Canada, taken from customer, address, city and country tables.



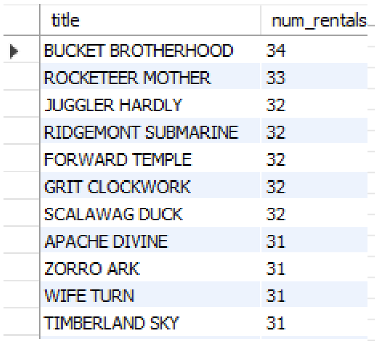
first_name	last_name	email
DERRICK	BOURQUE	DERRICK.BOURQUE@sakilacustomer.org
DARRELL	POWER	DARRELL.POWER@sakilacustomer.org
LORETTA	CARPENTER	LORETTA.CARPENTER@sakilacustomer.org
CURTIS	IRBY	CURTIS.IRBY@sakilacustomer.org
TROY	QUIGLEY	TROY.QUIGLEY@sakilacustomer.org

Films in the family category, taken from the film, film_category and category tables.

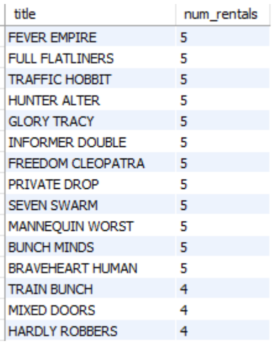


title
AFRICAN EGG
APACHE DIVINE
ATLANTIS CAUSE
BAKED CLEOPATRA
BANG KWAI
BEDAZZLED MARRIED
BILKO ANONYMOUS
BLANKET BEVERLY
BLOOD ARGONAUTS
BLUES INSTINCT
BRAVEHEART HUMAN
CHASING FIGHT

Most frequently rented films in descending order, both the top and bottom of result. Information retrieved from film, inventory and rental tables.

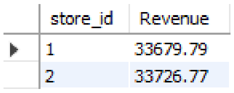


title	num_rentals
BUCKET BROTHERHOOD	34
ROCKETEER MOTHER	33
JUGGLER HARDLY	32
RIDGEMONT SUBMARINE	32
FORWARD TEMPLE	32
GRIT CLOCKWORK	32
SCALAWAG DUCK	32
APACHE DIVINE	31
ZORRO ARK	31
WIFE TURN	31
TIMBERLAND SKY	31



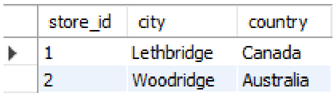
title	num_rentals
FEVER EMPIRE	5
FULL FLATLINERS	5
TRAFFIC HOBBIT	5
HUNTER ALTER	5
GLORY TRACY	5
INFORMER DOUBLE	5
FREEDOM CLEOPATRA	5
PRIVATE DROP	5
SEVEN SWARM	5
MANNEQUIN WORST	5
BUNCH MINDS	5
BRAVEHEART HUMAN	5
TRAIN BUNCH	4
MIXED DOORS	4
HARDLY ROBBERS	4

Revenue per store, taken from store, inventory, rental and payment tables.



store_id	Revenue
1	33679.79
2	33726.77

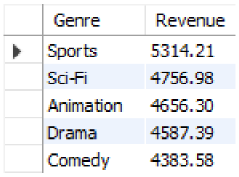
Store ID, city and country for each store, taken from the store, address, city and country tables.



store_id	city	country
1	Lethbridge	Canada
2	Woodridge	Australia

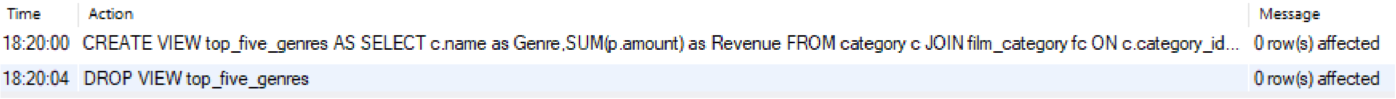
Example 8

Top five genres by revenue, taken from category, film_category, inventory, rental and payment tables.



Genre	Revenue
Sports	5314.21
Sci-Fi	4756.98
Animation	4656.30
Drama	4587.39
Comedy	4383.58

Action output for saving and deleting view is shown below



Time	Action	Message
18:20:00	CREATE VIEW top_five_genres AS SELECT c.name as Genre,SUM(p.amount) as Revenue FROM category c JOIN film_category fc ON c.category_id...	0 row(s) affected
18:20:04	DROP VIEW top_five_genres	0 row(s) affected

Conclusions

It can be observed that the two stores in this sample data are bringing in almost identical revenue and as they are in different countries for this hypothetical example there is a lot of opportunity to grow into different countries or districts with the same countries.

“Bucket Brotherhood” was rented the most times, so it would be beneficial to have more in inventory if there have been stockouts and movies similar to it should the target for purchases of new releases. Conversely, the films with low rentals would likely be more beneficial to be sold as used items rather than taking up rack space in the stores.

Future Analysis

Analyzing the results of amount spent by customers and sorting by revenue rather that customer name would yield a beneficial list to target for promotions or possibly even a monthly membership. Including the contact information for those customers would be necessary for any kind of outreach.

By joining the actor, film and film tables with revenue could produce a list of actors that are frequently appearing in films that bring in higher revenue than average.