MOVIE RENTAL DATA ANALYSIS

Naguleshwari.S.R

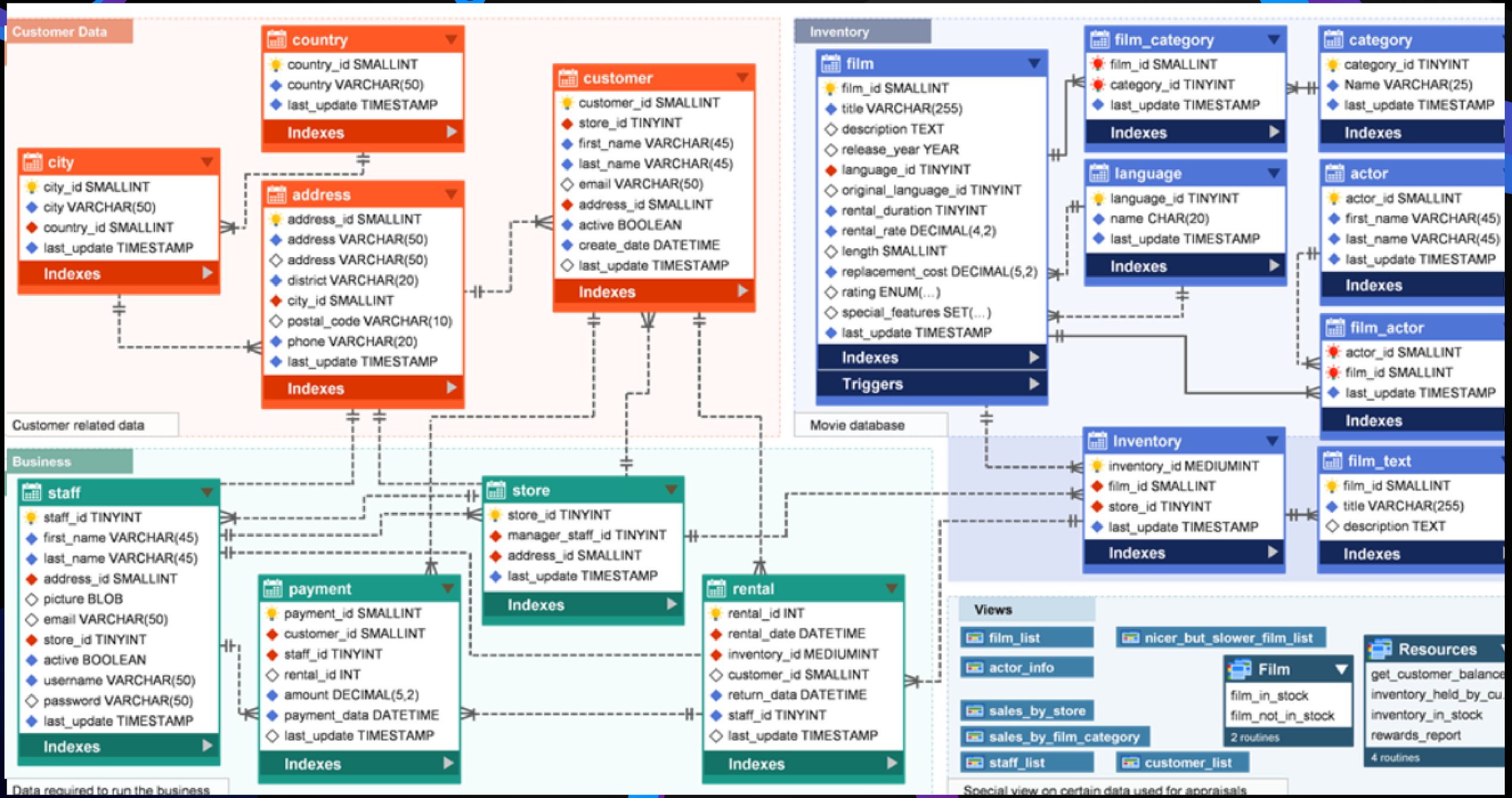
Sakila Database

- The sakila data base is a sample relational database, that is often used for educational and testing purposes.
- It is designed to simulate a fictional DVD rental store.
- Also it includes a set of tables that represent various aspects of the rental business, such as customers, films, actors, rentals and payments

Business objective

- Movie on rent is a chain of movie rental stores opearting in a certion couuntry. It has a vast collection of movies in DVD and blue ray disc formats.
- The management of the company wants to analyze what kind of movies are most often rented, which geners they belong to, and which actors appeared in them .
- It will help the management stock up the inventory as per audience’s preferences for improved business

ER Diagram


Customer Data
country
country_id SMALLINT
country VARCHAR(50)
last_update TIMESTAMP
Indexes
customer
customer_id SMALLINT
store_id TINYINT
first_name VARCHAR(45)
last_name VARCHAR(45)
email VARCHAR(50)
address_id SMALLINT
active BOOLEAN
create_date DATETIME
last_update TIMESTAMP
Indexes
city
city_id SMALLINT
city VARCHAR(50)
country_id SMALLINT
last_update TIMESTAMP
Indexes
address
address_id SMALLINT
address VARCHAR(50)
address VARCHAR(50)
district VARCHAR(20)
city_id SMALLINT
postal_code VARCHAR(10)
phone VARCHAR(20)
last_update TIMESTAMP
Indexes
Customer related data
Inventory
film
film_id SMALLINT
title VARCHAR(255)
description TEXT
release_year YEAR
language_id TINYINT
original_language_id TINYINT
rental_duration TINYINT
rental_rate DECIMAL(4,2)
length SMALLINT
replacement_cost DECIMAL(5,2)
rating ENUM(...)
special_features SET(...)
last_update TIMESTAMP
Indexes
Triggers
film_category
film_id SMALLINT
category_id TINYINT
last_update TIMESTAMP
Indexes
category
category_id TINYINT
Name VARCHAR(25)
last_update TIMESTAMP
Indexes
language
language_id TINYINT
name CHAR(20)
last_update TIMESTAMP
Indexes
actor
actor_id SMALLINT
first_name VARCHAR(45)
last_name VARCHAR(45)
last_update TIMESTAMP
Indexes
film_actor
actor_id SMALLINT
film_id SMALLINT
last_update TIMESTAMP
Indexes
Movie database
Business
staff
staff_id TINYINT
first_name VARCHAR(45)
last_name VARCHAR(45)
address_id SMALLINT
picture BLOB
email VARCHAR(50)
store_id TINYINT
active BOOLEAN
username VARCHAR(50)
password VARCHAR(50)
last_update TIMESTAMP
Indexes
store
store_id TINYINT
manager_staff_id TINYINT
address_id SMALLINT
last_update TIMESTAMP
Indexes
payment
payment_id SMALLINT
customer_id SMALLINT
staff_id TINYINT
rental_id INT
amount DECIMAL(5,2)
payment_data DATETIME
last_update TIMESTAMP
Indexes
rental
rental_id INT
rental_date DATETIME
inventory_id MEDIUMINT
customer_id SMALLINT
return_data DATETIME
staff_id TINYINT
last_update TIMESTAMP
Indexes
Inventory
inventory_id MEDIUMINT
film_id SMALLINT
store_id TINYINT
last_update TIMESTAMP
Indexes
film_text
film_id SMALLINT
title VARCHAR(255)
description TEXT
Indexes
Views
film_list
nicer_but_slower_film_list
actor_info
sales_by_store
sales_by_film_category
staff_list
customer_list
Film
film_in_stock
film_not_in_stock
2 routines
Resources
get_customer_balance
inventory_held_by_cu
inventory_in_stock
rewards_report
4 routines
Data required to run the business
Special view on certain data used for appraisals

SQL Analysis and output

```
-- Task 1---

Select * from actor;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

actor_id	first_name	last_name	last_update
1	PENELOPE	GUINESS	2006-02-15 04:34:33
2	NICK	WAHLBERG	2006-02-15 04:34:33
3	ED	CHASE	2006-02-15 04:34:33
4	JENNIFER	DAVIS	2006-02-15 04:34:33
5	JOHNNY	LOLLOBRIGIDA	2006-02-15 04:34:33
6	BETTE	NICHOLSON	2006-02-15 04:34:33
7	GRACE	MOSTEL	2006-02-15 04:34:33

Here i want to display the full names of all the actors in the database. so, i used this query to fetch all the actors name. There are 200 rows

```sql
-- Task 2i---
SELECT first_name, COUNT(*) AS Count FROM actor GROUP BY first_name;


-- Task 2ii---
SELECT first_name, COUNT(*) AS Count FROM actor GROUP BY first_name HAVING COUNT(*) =
```

Result Grid | Filter R

first_name	Count
JENNIFER	1
BETTE	1
GRACE	1
JOE	1
ZERO	1
KARL	1
UMA	1

Task 2.1: 72 actors

Result Grid | Filter

first_name	Count
PENELOPE	4
NICK	3
ED	3
JENNIFER	1
JOHNNY	2
BETTE	1
GRACE	1

Task 2.2: 76 rows

Task 2.1: Here i need to find the common first name using the count function

Task 2.2: Here i need to find the unique first name of the actors using the count function with given a condition the count as 1

```sql
-- Task 3i---
Select last_name, COUNT(*) AS Count FROM actor Group By last_name;

-- Task 3ii---
SELECT last_name, COUNT(*) AS Count FROM actor GROUP BY last_name HAVING COUNT(*) = 1;
```

Result Grid | Filter Ro

last_name	Count
AKROYD	3
ALLEN	3
ASTAIRE	1
BACALL	1
BAILEY	2
BALE	1
BALL	1

Task 3.1: 121 actors

Result Grid | Filter R

last_name	Count
ASTAIRE	1
BACALL	1
BALE	1
BALL	1
BARRYMORE	1
BASINGER	1
BERGEN	1

Task 3.2: 66 rows

Task 3.1: Here i need to find the common lastname using the count function

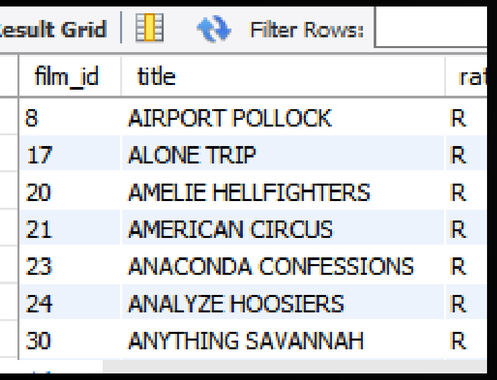
Task 3.2: Here i need to find the unique lastname of the actors using the count function with given a condition the count as 1

```sql
-- Task 4i---
SELECT film_id, title, rating FROM film WHERE rating = 'R';

-- Task 4ii---
Select film_id, title, rating From film where NOT rating = "R";

-- Task 4iii---
Select film_id, title, rating From film where rating = "G";
```

Task 4.1: Here i need to find the title starts with rating ‘R’ by using where function

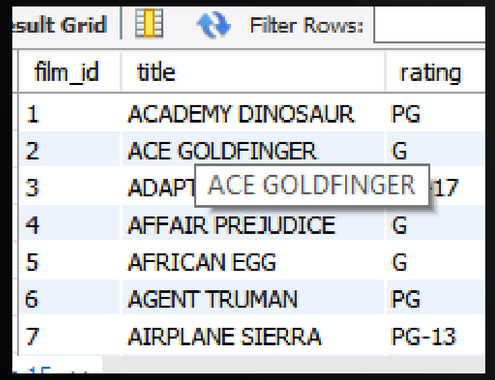
Task 4.2:Here i need to find the titlenot starts with rating ‘R’ by using where not function

Task 4.3: Here i need to find the title starts with rating ‘G’by using where function



film_id	title	rat
8	AIRPORT POLLOCK	R
17	ALONE TRIP	R
20	AMELIE HELLFIGHTERS	R
21	AMERICAN CIRCUS	R
23	ANACONDA CONFESSIONS	R
24	ANALYZE HOOSIERS	R
30	ANYTHING SAVANNAH	R

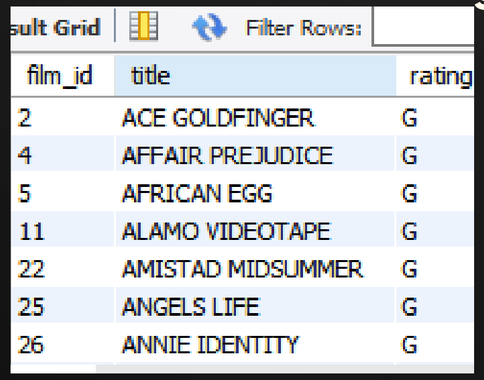
Task 4.1: 196 actors



film_id	title	rating
1	ACADEMY DINOSAUR	PG
2	ACE GOLDFINGER	G
3	ADAPT	17
4	AFFAIR PREJUDICE	G
5	AFRICAN EGG	G
6	AGENT TRUMAN	PG
7	AIRPLANE SIERRA	PG-13

ACE GOLDFINGER

Task 4.2: 806 rows



film_id	title	rating
2	ACE GOLDFINGER	G
4	AFFAIR PREJUDICE	G
5	AFRICAN EGG	G
11	ALAMO VIDEOTAPE	G
22	AMISTAD MIDSUMMER	G
25	ANGELS LIFE	G
26	ANNIE IDENTITY	G

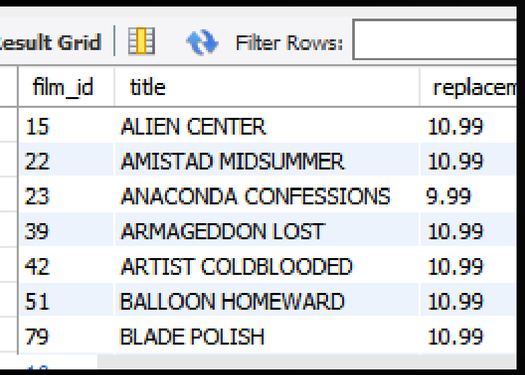
Task 4.3: 179 rows

```sql
-- Task 5i---
Select film_id, title, replacement_cost From film where replacement_cost<=11;

-- Task 5ii---
Select film_id, title, replacement_cost From film where replacement_cost between 11 and 20;

-- Task 5iii---
Select film_id, title, replacement_cost from film order by replacement_cost desc;
```

Task 5.1: Here i need to find the title with the replacement cost greater than 11 using where condition

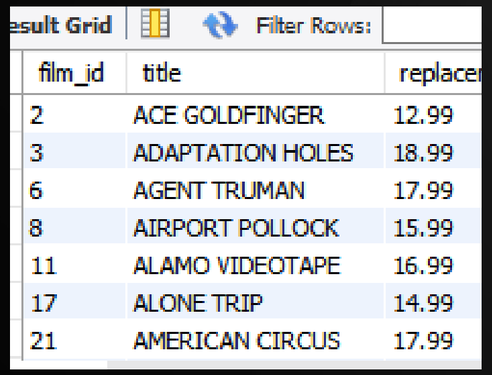
Task 5.2:Here i need to find the title with the replacement cost between 11 and 20 using where condition

Task 5.3: Here i need to find the title describing the replacement cost using order by condition



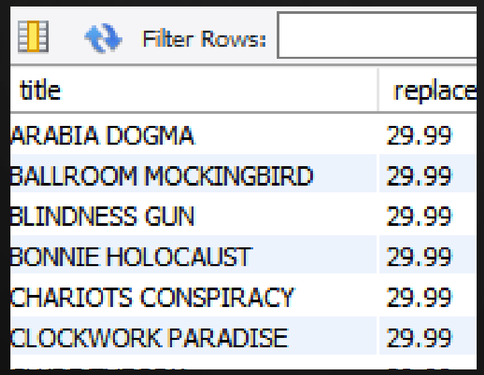
film_id	title	replacem
15	ALIEN CENTER	10.99
22	AMISTAD MIDSUMMER	10.99
23	ANACONDA CONFESSIONS	9.99
39	ARMAGEDDON LOST	10.99
42	ARTIST COLDBLOODED	10.99
51	BALLOON HOMEWARD	10.99
79	BLADE POLISH	10.99

Task 5.1: 191 actors



film_id	title	replacer
2	ACE GOLDFINGER	12.99
3	ADAPTATION HOLES	18.99
6	AGENT TRUMAN	17.99
8	AIRPORT POLLOCK	15.99
11	ALAMO VIDEOTAPE	16.99
17	ALONE TRIP	14.99
21	AMERICAN CIRCUS	17.99

Task 5.2: 425 rows



title	replace
ARABIA DOGMA	29.99
BALLROOM MOCKINGBIRD	29.99
BLINDNESS GUN	29.99
BONNIE HOLOCAUST	29.99
CHARIOTS CONSPIRACY	29.99
CLOCKWORK PARADISE	29.99

Task 5.2: 1001 rows

```sql
-- Task 6---
SELECT film.title, COUNT(actor.actor_id) AS actor_count FROM film JOIN film_actor ON film.film_id = film_actor.film_id
JOIN actor ON film_actor.actor_id = actor.actor_id GROUP BY film.film_id ORDER BY actor_count DESC LIMIT 3;
```

Result Grid | Filter Rows:

title	actor_count
LAMBS CINCINATTI	15
CRAZY HOME	13
RANDOM GO	13

Here i need to display the top 3 with the greatest number of actors. so i used the the filters like- joint, group by and order by.

There are 3 titke found with the greatest number of actor_count

```
-- Task 7---
Select title from film where title like 'K%' or title li
```

Result Grid | Filter Ro

title
KANE EXORCIST
KARATE MOON
KENTUCKIAN GIANT
KICK SAVANNAH
KILL BROTHERHOOD
KILLER INNOCENT
KING EVOLUTION

Here i need to display the titles of movie starting from ‘K’ and ‘Q’.

There are 15 rows found ,the title starts ‘K’ and ‘Q’ using the where condition

```sql
-- Task 8---
SELECT actor.actor_id, actor.first_name, actor.last_name FROM actor
JOIN film_actor ON actor.actor_id = film_actor.actor_id
JOIN film  ON film_actor.film_id = film.film_id WHERE film.title = 'Agent Truman
```

Result Grid | Filter Rows:

actor_id	first_name	last_name
21	KIRSTEN	PALTROW
23	SANDRA	KILMER
62	JAYNE	NEESON
108	WARREN	NOLTE
137	MORGAN	WILLIAMS
169	KENNETH	HOFFMAN
197	REESE	WEST

Here i need to display the name of all actors who apperared in the film ‘Agent Truman’

There are 7 rows found that the name of all actors appeared in that film using the filters like- join , where.

```sql
-- Task 9---
Select film.title from film JOIN film_category ON film.film_id = film_category.film_id
JOIN category ON film_category.category_id = category.category_id where category.name = 'Family';
```

Result Grid | Filter Ro

title
AFRICAN EGG
APACHE DIVINE
ATLANTIS CAUSE
BAKED CLEOPATRA
BANG KWAI
BEDAZZLED MARRIED
BILKO ANONYMOUS

Here i need to display all the movies that is categorized as family films

There are 69 rows found that the movies categorized as family films with the filter using- join and where

```sql
-- Task 10i---
SELECT film.rating, MAX(rental_rate) AS max_rental_rate, MIN(rental_rate) AS min_rental_rate, AVG(rental_rate) AS avg_rental_rate
FROM film GROUP BY film.rating ORDER BY avg_rental_rate DESC;

-- Task 10ii---
SELECT film.film_id, title, COUNT(rental_id) AS rental_count FROM rental JOIN inventory ON rental.inventory_id = inventory.inventory_id
JOIN film ON inventory.film_id = film.film_id GROUP BY film.film_id, title ORDER BY rental_count DESC;
```

Result Grid | Filter Rows: | Export:

rating	max_rental_rate	min_rental_rate	avg_rental_rate
PG	4.99	0.99	3.051856
PG-13	4.99	0.99	3.034843
NC-17	4.99	0.99	2.970952
R	4.99	0.99	2.938718
G	4.99	0.99	2.888876

Task 10.1 : 7 rows

Task 10.1: Here i need to display the minimum, maximum and rental rate of movies based on their ratings. the output displayed as desecnding order using groupby and orderby filter with 7 rows.

Result Grid | Filter Rows:

film_id	title	rental_count
103	BUCKET BROTHERHOOD	34
738	ROCKETEER MOTHER	33
331	FORWARD TEMPLE	32
382	GRIT CLOCKWORK	32
489	JUGGLER HARDLY	32
730	RIDGEMONT SUBMARINE	32
767	SCALAWAG DUCK	32

Task 10.2 : 958 rows

Task 10.2: Here i need to display the movies in descending order of their rental frequencies. the output displayed as desecnding order using groupby and orderby filter with 958 rows.

```sql
-- Task 11---
SELECT category.name AS category_name, AVG(film.replacement_cost - film.rental_rate)
AS cost_rate_difference FROM film JOIN film_category ON film.film_id = film_category.film_id
JOIN category ON film_category.category_id = category.category_id
GROUP BY category_name HAVING AVG(film.replacement_cost - film.rental_rate) > 15;
```

Result Grid | Filter Rows:

category_name	cost_rate_difference
Action	18.265625
Animation	17.318182
Children	17.166667
Classics	18.263158
Comedy	15.862069
Documentary	16.955882
Drama	18.064516

Task 11: Here i need to display the list of all film categories with the corresponding average film replacement cost and average film rental rate.
There are 16 rows displayed by using join,groupby and having the condition of greater than 15

```sql
-- Task 12---
SELECT category.name AS category_name, COUNT(film.film_id) AS movie_count FROM category
JOIN film_category ON category.category_id = film_category.category_id JOIN film ON film_category.film_id = film.film_id
GROUP BY category_name HAVING movie_count > 70;
```

Result Grid | Filter Rows:

category_name	movie_count
Foreign	73
Sports	74

Task 12: Here i need to display the film categories in which the number of movies is greater than 70

There are 2 rows using the filter join, groupby and having condition of greater than 70

THANK YOU