**Csl310 Assignment 2 Report**

In this assignment i have used the following given queries,their outputs and i have mentioned the assumptions made for each of the queries given :-

Q1)

1- **Query:-**

SElect language.name,

case

when AVG(payment.amount) is not null then AVG(payment.amount)

else 0

end as Average

from film inner join inventory on inventory.film\_id=film.film\_id inner join rental on rental.inventory\_id=inventory.inventory\_id inner JOIN payment on payment.rental\_id=rental.rental\_id

right join language on film.language\_id=language.language\_id

group by language.language\_id

**Explanation:-** in query i have joined tables film,inventory,rental,language,payment and then grouped by lanuage\_id and shown average of all the languages even if it’s zero.

2- **Query:-**

SElect category.name, case

when AVG(payment.amount) is not null then AVG(payment.amount)

else 0

end as Average

from film inner join film\_category on film.film\_id=film\_category.film\_id inner join category on category.category\_id=film\_category.category\_id inner join inventory on inventory.film\_id=film.film\_id inner join rental on rental.inventory\_id=inventory.inventory\_id inner JOIN payment on payment.rental\_id=rental.rental\_id

group by category.category\_id

**Explanation:-** in query i have joined tables film,inventory,rental,payment,category and film\_category and then grouped by category\_id and shown average of all the categories even if it’s zero.

3- **Query:-**

SElect film.rating,

case

when AVG(payment.amount) is not null then AVG(payment.amount)

else 0

end as Average

from film inner join inventory on inventory.film\_id=film.film\_id inner join rental on rental.inventory\_id=inventory.inventory\_id inner JOIN payment on payment.rental\_id=rental.rental\_id

group by film.rating

**Explanation:-** in query i have joined tables film,inventory,rental,payment and then grouped by film.rating and shown average of all the ratings.

Q2)

**Query:-**

Select inventory.store\_id, sum(payment.amount- film.rental\_rate )as Late\_Fee

from film inner join inventory on inventory.film\_id=film.film\_id

inner join rental on rental.inventory\_id=inventory.inventory\_id

inner JOIN payment on payment.rental\_id=rental.rental\_id

where rental.return\_date is not null and rental.return\_date > DATE\_ADD(rental.rental\_date, INTERVAL film.rental\_duration DAY)

GROUP by inventory.store\_id

I HAVE ASSUMED THAT STORE WHOSE INVENTORY IS RENTED GETS EARNINGS THUS STORE ID FROM INVENTORY

**Explanation:-** in query i have joined tables film,inventory,rental,payment and then grouped by inventory.store\_id and shown storewise earnings.In this i have assumed that the staff\_id is taken for films from the inventory as they are having the films.I have found the late date if the rental date + film duration is greater than return date

Q3)

**Query:-**

SELECT CONCAT(actor.first\_name," ",actor.last\_name) as Actor,

case

when SUM(payment.amount) is not null then SUM(payment.amount)

else 0

end as Total\_Earnings

from film inner join film\_actor on film.film\_id=film\_actor.film\_id

inner join actor on actor.actor\_id=film\_actor.actor\_id

inner join inventory on inventory.film\_id=film.film\_id

inner join rental on rental.inventory\_id=inventory.inventory\_id

inner JOIN payment on payment.rental\_id=rental.rental\_id

where (MONTHNAME(payment.payment\_date)) LIKE "January"

group by actor.actor\_id

ORDER BY Total\_Earnings DESC

limit 5

**Explanation:-** in query i have joined tables film,inventory,rental,payment,actor,film\_actor and then grouped by actor\_id where month is same as input given and shown top 5 actors.In this i have assumed that the earning is taken for films when the payment is made not when they are rented.I have assumed user gives the input in command where the month can be given by its complete name with capital letter in start(as shown).

Q4)

**Query:-**

SELECT category.name as Category,

case

when SUM(rental.rental\_id) is not null then SUM(rental.rental\_id)

else 0

end as Total\_Earnings

from rental inner join inventory on inventory.inventory\_id=rental.inventory\_id

inner join film on film.film\_id=inventory.film\_id

inner JOIN film\_category on film\_category.film\_id=film.film\_id

inner join category on category.category\_id=film\_category.category\_id

where Upper( (MONTHNAME(rental.rental\_date)) ) LIKE "JAN%"

group by category.category\_id

ORDER BY Total\_Earnings DESC

limit 1

I HAVE ASSUMED THAT RENTAL OF MOVIE IN A MONTH IS MADE IF MONTH IS IN RENTAL DATE NOT IN THE DURATION OF RENTAL.

**Explanation:-** in query i have joined tables film,inventory,rental,payment,category,film\_category and then grouped by category\_id where month is same as input given as mentioned in THREE Letters and shown top category.In this i have assumed that the rental is taken for films when the when they are rented i.e. rental date not in the duration given.I have assumed user gives the input in command where the month can be given by its THREE Letters with all capital letter (as shown).

Q5)

**Query:-**

SELECT CONCAT(staff.first\_name," ",staff.last\_name) as Staff,

case

when SUM(payment.amount) is not null then SUM(payment.amount)

else 0

end as Total\_Earnings

from rental inner join staff on rental.staff\_id=staff.staff\_id inner join payment on rental.rental\_id=payment.rental\_id

where (MONTHNAME(rental.rental\_date)) LIKE "January"

group by staff.staff\_id

ORDER BY Total\_Earnings DESC

limit 1

I HAVE ASSUMED SALES ARE MADE WHEN MOVIE ARE RENTED NOT PAYMENT.

**Explanation:-** in query i have joined tables rental,payment,staff and then grouped by staff\_id where month is same as input given and shown top staff.In this i have assumed that the salel is taken for films when the when they are rented.I have assumed user gives the input in command where the month can be given by its complete name with capital letter in start(as shown).