**Designing Data Architecture for Business Intelligence**

*Name: Priyanka Milind Malpekar*

*NUID: 00130274*

**SQL Queries**

**Question1**

**Rentals by Customer Geography**

-- 1.1 Contribution of Countries & Cities (in hierarchy) by rental amount - Treemap (or Heatmap) & sql query

SELECT c.country,

       ci.city,

       sum(p.amount) AS 'Total Amount'

FROM country c

INNER JOIN city ci ON c.country\_id = ci.country\_id

INNER JOIN address a ON a.city\_id = ci.city\_id

INNER JOIN customer cust ON cust.address\_id = a.address\_id

INNER JOIN payment p ON p.customer\_id = cust.customer\_id

GROUP BY c.country,

         ci.city

ORDER BY sum(p.amount) DESC;

-- 1.2 Rental amounts by countries for PG & PG-13 rated films - bar chart & sql query

SELECT c.country,

       sum(p.amount) AS 'Total Amount for PG & PG-13'

FROM country c

INNER JOIN city ci ON c.country\_id = ci.country\_id

INNER JOIN address a ON a.city\_id = ci.city\_id

INNER JOIN customer cust ON cust.address\_id = a.address\_id

INNER JOIN payment p ON p.customer\_id = cust.customer\_id

WHERE p.rental\_id IN

    (SELECT rental\_id

     FROM rental

     WHERE inventory\_id IN

         (SELECT inventory\_id

          FROM inventory

          WHERE film\_id IN

              (SELECT film\_id

               FROM film

               WHERE rating IN ('PG','PG-13'))))

GROUP BY c.country

ORDER BY sum(p.amount) DESC;

-- 1.3. Top 20 cities by number of customers who rented - bar chart & sql query

SELECT ci.city,

       count(r.customer\_id) as 'Number of Customers'

FROM rental r

INNER JOIN customer cust ON r.customer\_id = cust.customer\_id

INNER JOIN address a ON a.address\_id = cust.address\_id

INNER JOIN city ci ON ci.city\_id = a.city\_id

WHERE cust.customer\_id in (select customer\_id from rental)

GROUP BY ci.city

ORDER BY count(r.customer\_id) DESC limit 20;

-- 1.4. Top 20 cities by number of films rented - bar chart & sql query

SELECT ci.city,

       count(r.inventory\_id) as 'Total Film Rentals'

FROM rental r

INNER JOIN customer cust ON r.customer\_id = cust.customer\_id

INNER JOIN address a ON a.address\_id = cust.address\_id

INNER JOIN city ci ON ci.city\_id = a.city\_id

INNER JOIN inventory i ON i.inventory\_id = r.inventory\_id

INNER JOIN film f ON f.film\_id = i.film\_id

GROUP BY ci.city

ORDER BY count(r.inventory\_id) DESC limit 20;

-- 1.5. Rank cities by average rental cost - bar chart & sql query

SELECT ci.city,

       AVG(p.amount) "Average Rental",

       RANK() OVER (ORDER BY AVG(p.amount) DESC) AS 'Rank'

FROM customer c

INNER JOIN rental r ON c.customer\_id = r.customer\_id

INNER JOIN payment p ON p.rental\_id = r.rental\_id

INNER JOIN address a ON a.address\_id = c.address\_id

INNER JOIN city ci ON ci.city\_id = a.city\_id

GROUP BY ci.city

ORDER BY AVG(p.amount) DESC;

**Question2**

#Film categories by rental amount (ranked) & rental quantity – tabular & sql query

SELECT c.name, sum(p.amount) as RentalAmount,

        count(r.rental\_id) as Rental\_Quantity,

        RANK() OVER (ORDER BY sum(p.amount) DESC) AS 'Rank'

FROM   category AS c

       JOIN film\_category AS f

         ON c.category\_id = f.category\_id

       JOIN inventory AS i

         ON i.film\_id = f.film\_id

  JOIN rental AS r

         ON r.inventory\_id = i.inventory\_id

  JOIN payment AS p

         ON p.rental\_id = r.rental\_id

GROUP  BY c.name

ORDER  BY sum(p.amount) desc;

#2-Film categories by rental amount (ranked) – bar chart & sql query

SELECT  RANK() OVER (ORDER BY sum(p.amount) DESC) AS 'Rank',c.name FilmCategory,sum(p.amount) as RentalAmount

FROM   category AS c

       JOIN film\_category AS f

         ON c.category\_id = f.category\_id

       JOIN inventory AS i

         ON i.film\_id = f.film\_id

  JOIN rental AS r

         ON r.inventory\_id = i.inventory\_id

  JOIN payment AS p

         ON p.rental\_id = r.rental\_id

GROUP  BY c.name

ORDER  BY sum(p.amount) desc;

#3- Film categories by average rental amount (ranked) & – tabular & sql query

SELECT  RANK() OVER (ORDER BY avg(p.amount) DESC) AS 'Rank',c.name FilmCategory,avg(p.amount) as RentalAmount

FROM   category AS c

       JOIN film\_category AS f

         ON c.category\_id = f.category\_id

       JOIN inventory AS i

         ON i.film\_id = f.film\_id

  JOIN rental AS r

         ON r.inventory\_id = i.inventory\_id

  JOIN payment AS p

         ON p.rental\_id = r.rental\_id

GROUP  BY c.name;

#4-Contribution of Film Categories by number of customers - Treemap (or Heatmap) & sql query

SELECT  ct.name, count(c.customer\_id) as NumberOfCustomers

FROM   customer AS c

       JOIN rental AS r

         ON c.customer\_id = r.customer\_id

  JOIN inventory as i

          ON r.inventory\_id=i.inventory\_id

  JOIN film as fi

  ON fi.film\_id=i.film\_id

JOIN film\_category as f

          ON f.film\_id=i.film\_id

JOIN category as ct

           ON ct.category\_id=f.category\_id

GROUP  BY ct.name

ORDER  BY NumberOfCustomers ;

#5-Contribution of Film Categories by rental amount - Treemap (or Heatmap) & sql query

SELECT  ct.name,sum(p.amount) as RentalAmount

FROM   customer AS c

       JOIN rental AS r

         ON c.customer\_id = r.customer\_id

  JOIN inventory as i

          ON r.inventory\_id=i.inventory\_id

JOIN film\_category as f

          ON f.film\_id=i.film\_id

JOIN category as ct

           ON ct.category\_id=f.category\_id

JOIN payment as p

           ON p.rental\_id=r.rental\_id

GROUP  BY ct.name

ORDER  BY RentalAmount desc;

**Question3**

#3.1. List Films with rental amount, rental quantity, rating, rental rate, replacement cost and category name – tabular & sql query

select f.title as films, f.replacement\_cost, f.rental\_rate, f.rating, sum(p.amount) as Rental\_amount,count(r.rental\_id) as Rental\_Quantity, c.name as Category\_name, c.category\_id

from payment as p

join rental as r on p.rental\_id = r.rental\_id

join inventory as i on r.inventory\_id = i.inventory\_id

join film as f on i.film\_id = f.film\_id

join film\_category as fc on f.film\_id = fc.film\_id

join category as c on fc.category\_id = c.category\_id

group by f.title;

#3.2. List top 10 Films by rental amount (ranked) – bar chart & sql query

select RANK() OVER (ORDER BY sum(p.amount) DESC) AS 'Rank',f.title as films,sum(p.amount) as rental\_amount

from payment as p

join rental as r on p.rental\_id = r.rental\_id

join inventory as i on r.inventory\_id = i.inventory\_id

join film as f on i.film\_id = f.film\_id

group by f.film\_id

order by rental\_amount desc

limit 10;

#3.3. List top 20 Films by number of customers(ranked) – bar chart & sql query

select row\_number() OVER (ORDER BY count(r.customer\_id) DESC) AS 'Rank', f.title,count(r.customer\_id) as count

from rental as r

JOIN inventory as i ON r.inventory\_id = i.inventory\_id

JOIN film as f ON i.film\_id = f.film\_id

GROUP BY f.title

ORDER BY count(r.customer\_id) desc

limit 20;

#3.4. List Films with the word “punk” in title with rental amount and number of customers –

Tabular & sql query

select f.film\_id, f.title ,  p.amount as rental\_amount, count(r.customer\_id) as NumberOfCustomers

from payment as p

join rental as r on p.rental\_id = r.rental\_id

join inventory as i on r.inventory\_id = i.inventory\_id

join film as f on i.film\_id = f.film\_id

where f.title like '%punk%'

group by film\_id;

#3.5. Contribution by rental amount for films with a documentary category - Treemap (or Heatmap) & sql query

select f.title as Films, sum(p.amount) as rental\_amount

from payment as p

join rental as r on p.rental\_id = r.rental\_id

join inventory as i on i.inventory\_id = r.inventory\_id

join film as f on  i.film\_id = f.film\_id

join film\_category as g on i.film\_id = g.film\_id

join category as c on g.category\_id = c.category\_id

where name like '%doc%'

group by Films

order by rental\_amount desc;

**Question4**

#1.List Customers (Last name, First Name) with rental amount, rental quantity, active status, country and city – tabular & sql query

SELECT c.last\_name,c.first\_name, sum(p.amount) as RentalAmount, count(r.rental\_id) as Rental\_Quantity, country.country, city.city,c.active

FROM   customer AS c

       JOIN rental AS r

         ON c.customer\_id = r.customer\_id

JOIN payment as p

           ON p.rental\_id=r.rental\_id

JOIN address as a

           ON a.address\_id=c.address\_id

JOIN city as city

           ON a.city\_id=city.city\_id

JOIN country as country

           ON country.country\_id=city.country\_id

WHERE c.active = 1

GROUP  BY c.last\_name,c.first\_name

ORDER  BY c.last\_name,c.first\_name desc;

#2-List top 10 Customers (Last name, First Name) by rental amount (ranked) for PG & PG-13 rated films – bar chart & sql query

SELECT RANK() OVER (ORDER BY sum(p.amount) DESC) AS 'Rank', c.last\_name,c.first\_name, sum(p.amount) as RentalAmount

FROM   customer AS c

       JOIN rental AS r

         ON c.customer\_id = r.customer\_id

  JOIN inventory as i

          ON r.inventory\_id=i.inventory\_id

JOIN film\_category as f

          ON f.film\_id=i.film\_id

JOIN film as fil

          ON fil.film\_id = f.film\_id

JOIN category as ct

           ON ct.category\_id=f.category\_id

JOIN payment as p

           ON p.rental\_id=r.rental\_id

WHERE fil.rating ='PG' or fil.rating = 'PG-13'

GROUP  BY c.last\_name,c.first\_name

ORDER  BY RentalAmount desc

limit 10;

#3-Contribution by rental amount for customers from France, Italy or Germany - Treemap (or Heatmap) & sql query

SELECT c.last\_name,c.first\_name, sum(p.amount) as RentalAmount

FROM   customer AS c

       JOIN rental AS r

         ON c.customer\_id = r.customer\_id

  JOIN payment as p

           ON p.rental\_id=r.rental\_id

JOIN address as a

           ON a.address\_id=c.address\_id

JOIN city as city

           ON a.city\_id=city.city\_id

JOIN country as country

           ON country.country\_id=city.country\_id

WHERE country.country ='France' or country.country ='Italy' or country.country ='Germany'

GROUP  BY c.customer\_id

ORDER  BY RentalAmount desc;

#4- List top 20 Customers (Last name, First Name) by rental amount (ranked) for comedy films – bar chart & sql query

SELECT row\_number() OVER (ORDER BY sum(p.amount) DESC) AS 'Rank', c.last\_name,c.first\_name, sum(p.amount) as RentalAmount

FROM   customer AS c

       JOIN rental AS r

         ON c.customer\_id = r.customer\_id

  JOIN inventory as i

          ON r.inventory\_id=i.inventory\_id

JOIN film\_category as f

          ON f.film\_id=i.film\_id

JOIN category as ct

           ON ct.category\_id=f.category\_id

JOIN payment as p

           ON p.rental\_id=r.rental\_id

WHERE ct.name = 'Comedy'

GROUP  BY c.customer\_id

limit 20;

#5-List top 10 Customers (Last name, First Name) from China by rental amount (ranked) for films that have

#replacement costs greater than $24 – bar chart & sql query

SELECT row\_number() OVER (ORDER BY sum(p.amount) DESC) AS 'Rank',c.last\_name,c.first\_name, sum(p.amount) as RentalAmount

FROM   customer AS c

       JOIN rental AS r

         ON c.customer\_id = r.customer\_id

  JOIN inventory as i

          ON r.inventory\_id=i.inventory\_id

JOIN film\_category as f

          ON f.film\_id=i.film\_id

JOIN film as fi

          ON fi.film\_id=f.film\_id

JOIN category as ct

           ON ct.category\_id=f.category\_id

JOIN payment as p

           ON p.rental\_id=r.rental\_id

JOIN address as a

           ON a.address\_id=c.address\_id

JOIN city as city

           ON a.city\_id=city.city\_id

JOIN country as country

           ON country.country\_id=city.country\_id

WHERE country.country ='China' AND fi.replacement\_cost>24

GROUP  BY c.customer\_id

limit 10;