

-- Query 1

```
WITH CustomerSpending AS (
    SELECT
        Customer_ID,
        Customer_Name,
        Total_Revenue AS total_spent
    FROM CustomerInsightsDW
),
RankedCustomers AS (
    SELECT
        cs.*,
        NTILE(100) OVER (ORDER BY total_spent DESC) AS percentile_rank
    FROM CustomerSpending cs
)
SELECT
    rc.Customer_ID,
    rc.Customer_Name,
    rc.total_spent,
    CASE
        WHEN percentile_rank <= 10 THEN 'Platinum'
        WHEN percentile_rank <= 50 THEN 'Gold'
        ELSE 'Regular'
    END AS Customer_Tier
FROM RankedCustomers rc
ORDER BY rc.total_spent DESC;
```

-- Query 2

```
WITH LateReturns AS (
    SELECT
        Customer_ID,
        Customer_Name,
        Total_Rentals,
        Total_Late_Fees,
        ROUND(Total_Late_Fees / NULLIF(Total_Rentals, 0), 2) AS Avg_Late_Fee_Per_Rental,
        NTILE(100) OVER (ORDER BY Total_Late_Fees DESC) AS Percentile_Rank
    FROM CustomerInsightsDW
    WHERE Total_Late_Fees > 0
)
SELECT
    lr.Customer_ID,
    lr.Customer_Name,
    lr.Total_Rentals,
    lr.Total_Late_Fees,
    lr.Avg_Late_Fee_Per_Rental,
    CASE
        WHEN lr.Percentile_Rank <= 10 THEN 'High Risk'
        WHEN lr.Percentile_Rank <= 50 THEN 'Moderate Risk'
        ELSE 'Low Risk' -- Bottom 50%
    END AS Late_Fee_Risk_Category
FROM LateReturns lr
ORDER BY lr.Total_Late_Fees DESC
LIMIT 25;
```

-- Query 3

```
WITH CountryRevenue AS (
    SELECT
        Customer_Country,
        COUNT(Customer_ID) AS Total_Customers,
        SUM(Total_Revenue) AS Total_Revenue,
        ROUND(SUM(Total_Revenue) / COUNT(Customer_ID), 2) AS Avg_Revenue_Per_Customer
    FROM CustomerInsightsDW
    GROUP BY Customer_Country
)
SELECT
    cr.Customer_Country,
    cr.Total_Customers,
    cr.Total_Revenue,
    cr.Avg_Revenue_Per_Customer,
    RANK() OVER (ORDER BY cr.Total_Revenue DESC) AS Revenue_Rank
FROM CountryRevenue cr
ORDER BY cr.Total_Revenue DESC;
```

-- Query 4

```
WITH GenrePreferences AS (
    SELECT
        Customer_ID,
        Customer_Name,
```

```
Highest_Rented_Category AS Genre,
SUM(Total_Rentals) AS Total_Rentals,
SUM(Total_Revenue) AS Total_Genre_Revenue,
RANK() OVER (PARTITION BY Customer_ID ORDER BY SUM(Total_Rentals) DESC, SUM(Total_Revenue) DESC) AS Genre_Rank
FROM CustomerInsightsDW
GROUP BY Customer_ID, Customer_Name, Highest_Rented_Category
)
SELECT
    Customer_ID,
    Customer_Name,
    Genre,
    Total_Rentals,
    Total_Genre_Revenue
FROM GenrePreferences
WHERE Genre_Rank = 1
ORDER BY Total_Rentals DESC, Total_Genre_Revenue DESC;
```

-- Query 5

```
WITH GenrePreferences AS (
    SELECT
        Customer_ID,
        Highest_Rented_Category AS Genre
    FROM CustomerInsightsDW
    GROUP BY Customer_ID, Highest_Rented_Category
),
RankedGenres AS (
    SELECT
        gp.Customer_ID,
        ci.Customer_Name,
        gp.Genre,
        RANK() OVER (PARTITION BY gp.Customer_ID ORDER BY COUNT(*) DESC) AS Genre_Rank
    FROM GenrePreferences gp
    JOIN CustomerInsightsDW ci ON gp.Customer_ID = ci.Customer_ID
    GROUP BY gp.Customer_ID, ci.Customer_Name, gp.Genre
)
SELECT
    Customer_ID,
    Customer_Name,
    Genre
FROM RankedGenres
WHERE Genre_Rank = 1
ORDER BY Customer_Name;
```

-- Query 6

```
WITH FilmPreferences AS (
    SELECT
        Customer_ID,
        Highest_Rented_Film AS Film_Title,
        SUM(Total_Revenue) AS Total_Film_Revenue,
        SUM(Total_Rentals) AS Total_Film_Rentals,
        ROUND(SUM(Total_Revenue) / NULLIF(SUM(Total_Rentals), 0), 2) AS Revenue_Per_Rental
    FROM CustomerInsightsDW
    GROUP BY Customer_ID, Highest_Rented_Film
),
RankedFilms AS (
    SELECT
        fp.Customer_ID,
        ci.Customer_Name,
        fp.Film_Title,
        fp.Total_Film_Revenue,
        fp.Total_Film_Rentals,
        fp.Revenue_Per_Rental,
        RANK() OVER (PARTITION BY fp.Customer_ID ORDER BY fp.Revenue_Per_Rental DESC) AS Film_Rank
    FROM FilmPreferences fp
    JOIN CustomerInsightsDW ci ON fp.Customer_ID = ci.Customer_ID
)
SELECT
    Customer_ID,
    Customer_Name,
    Film_Title,
    Total_Film_Rentals,
    Total_Film_Revenue,
    Revenue_Per_Rental
FROM RankedFilms
WHERE Film_Rank = 1
ORDER BY Revenue_Per_Rental DESC, Total_Film_Revenue DESC;
```

-- Query 7

```
WITH RentalCounts AS (
    SELECT
```

```

        MONTHNAME(First_Rental_Date) AS rental_month,
        COUNT(*) AS rental_count
FROM CustomerInsightsDW
GROUP BY rental_month

UNION ALL

SELECT
    MONTHNAME>Last_Rental_Date) AS rental_month,
    COUNT(*) AS rental_count
FROM CustomerInsightsDW
GROUP BY rental_month
)
SELECT
    rental_month,
    SUM(rental_count) AS total_rentals
FROM RentalCounts
GROUP BY rental_month
ORDER BY total_rentals DESC
LIMIT 25;

```

-- Query 8

```

WITH GenreRevenue AS (
    SELECT
        Customer_Country AS country,
        Highest_Rented_Category AS genre,
        SUM(Total_Revenue) AS total_revenue
FROM CustomerInsightsDW
GROUP BY Customer_Country, Highest_Rented_Category
)
SELECT
    gr.country,
    gr.genre,
    gr.total_revenue
FROM GenreRevenue gr
ORDER BY gr.total_revenue DESC;

```