Code Glossary

Module 3: Advanced Functions on Single Tables in SQL

1. **To find the percentage change in weekday and weekend revenue over the months from the dataset:**

```
SELECT *,
round((( total_revenue-previous_rev)/previous_rev)*100) as
Percentage_Change
                    FROM
                    (
                      SELECT *,
                      LAG(total_revenue) OVER (partition by Day_of_week) as
previous_rev
                      FROM
                    (
                    SELECT
                     CASE
                       WHEN DAYOFWEEK(order_date) BETWEEN 2 AND 6 THEN
'Weekday'
                       WHEN DAYOFWEEK(order_date) IN (1, 7) THEN 'Weekend'
                     END AS Day_of_week,
                     Month(order_date) AS Month,
                     Round(SUM(final_price),0) AS total_revenue
                    FROM orders
                    GROUP BY Day_of_week, Month
                    ORDER BY Day_of_week)
                    t1
               )
          t2;
```

2. **To find the average delivery time over the months from the dataset:**

```
SELECT Month(order_date) AS Month,
AVG(TIMESTAMPDIFF(MINUTE, order_time, delivered_time)) AS
average_delivery_time
FROM orders
GROUP BY Month;
```

3. **To rank the the drivers and find the drivers with the least average delivery time over the months from the dataset:**

```
select * from
(
SELECT month, driver_id, avg_time, RANK() OVER (PARTITION BY month
ORDER BY avg_time) AS driver_rank
FROM (
SELECT MONTH(order_date) AS month, driver_id,
AVG(minute(TIMEDIFF(delivered_time, order_time))) AS avg_time
        FROM orders
        GROUP BY month, driver_id
) AS query_1
) query_2
where driver_rank between 1 and 5;
```