#Question 1 : Create a query to get the total users who completed the order and total orders  per month (time frame Jan 2019 - April 2022)

--alternative query 1

        SELECT FORMAT\_DATE('%b %Y', created\_at) as Month,

               COUNT(DISTINCT user\_id) as Total\_User,

               COUNT(order\_id) as Total\_Orders,

          FROM `bigquery-public-data.thelook\_ecommerce.orders`

         WHERE status = 'Complete' AND created\_at BETWEEN '2019-01-01' and '2022-04-30'

      GROUP BY Month

      ORDER BY PARSE\_DATE('%b %Y', Month)

--alternative query 2

        SELECT DATE\_TRUNC(created\_at,Month) as Month,

               COUNT(DISTINCT user\_id) as Total\_User,

               COUNT(order\_id) as Total\_Orders,

          FROM `bigquery-public-data.thelook\_ecommerce.orders`

         WHERE status = 'Complete' AND created\_at BETWEEN '2019-01-01' and '2022-04-30'

      GROUP BY Month

      ORDER BY Month

#Question 2 : Create a query to get average order value and total number of unique users, grouped by month

--alternative query 1

        SELECT DATE\_TRUNC(DATE(I.created\_at),Month) as Month,

               COUNT(DISTINCT I.user\_id) as Distinct\_User,

               ROUND(SUM(sale\_price)/COUNT(DISTINCT order\_id),2) as AOV,

          FROM `bigquery-public-data.thelook\_ecommerce.order\_items` as I

          JOIN `bigquery-public-data.thelook\_ecommerce.products` as P

            ON I.product\_id = P.id

         WHERE I.created\_at BETWEEN '2019-01-01' and '2022-04-30' AND status = 'Complete'

      GROUP BY Month

      ORDER BY Month

--alternative query 2

        SELECT DATE\_TRUNC(DATE(a.created\_at),Month) as Month,

               COUNT(DISTINCT a.user\_id) as Distinct\_Users,

               SUM(b.sale\_price)/COUNT(a.order\_id) as Average\_Order\_Value

          FROM `bigquery-public-data.thelook\_ecommerce.orders` a

    INNER JOIN `bigquery-public-data.thelook\_ecommerce.order\_items` b

            ON a.order\_id = b.order\_id

         WHERE a.created\_at BETWEEN '2019-01-01' AND '2022-04-30'

           AND a.status = "Complete"

      GROUP BY Month

      ORDER BY Month DESC;

#Question 3 : Find the first and last name of users from the youngest and oldest age of each  gender

--alternative query 1

WITH Youngest\_table AS

(

        SELECT gender,

               'Youngest\_age' AS Remark,

               MIN(Age) as Age,

          FROM `bigquery-public-data.thelook\_ecommerce.users`

      GROUP BY 1

),

Oldest\_table AS

(

        SELECT gender,

               'Oldest\_age' AS Remark,

               MAX(Age) as Age,

          FROM `bigquery-public-data.thelook\_ecommerce.users`

      GROUP BY 1

)

        SELECT Y.gender,

               Y.remark,

               Y.age,

               U.first\_name,

               U.last\_name,

          FROM Youngest\_Table as Y

     LEFT JOIN `bigquery-public-data.thelook\_ecommerce.users` as U USING (gender)

         WHERE Y.age = U.age

     UNION ALL

        SELECT O.gender,

               O.remark,

               O.age,

               U.first\_name,

               U.last\_name,

          FROM Oldest\_Table as O

     LEFT JOIN `bigquery-public-data.thelook\_ecommerce.users` as U USING (gender)

         WHERE O.age = U.age

--alternative query 2

WITH Male as

(

        SELECT MAX(users.age) as theoldest,

               MIN(users.age) as theyoungest

          FROM `bigquery-public-data.thelook\_ecommerce.users` as users

         WHERE users.gender = 'M'

),

Female as

(

        SELECT MAX(users.age) as theoldest,

               MIN(users.age) as theyoungest

          FROM `bigquery-public-data.thelook\_ecommerce.users` as users

         WHERE users.gender = 'F'

)

        SELECT users.gender,

               'Oldest' as Age\_Status,

               users.first\_name,

               users.last\_name,

               users.age

          FROM `bigquery-public-data.thelook\_ecommerce.users` as users

    INNER JOIN Male as MM ON MM.theoldest = users.age

         WHERE users.gender = 'M'

     UNION ALL

        SELECT users.gender,

               'Youngest' as Age\_Status,

               users.first\_name,

               users.last\_name,

               users.age

          FROM `bigquery-public-data.thelook\_ecommerce.users` as users

    INNER JOIN Male as NN ON NN.theyoungest = users.age

         WHERE users.gender = 'M'

     UNION ALL

        SELECT users.gender,

               'Oldest' as Age\_Status,

               users.first\_name,

               users.last\_name,

               users.age

          FROM `bigquery-public-data.thelook\_ecommerce.users` as users

    INNER JOIN Female as FF ON FF.theoldest = users.age

         WHERE users.gender = 'F'

     UNION ALL

        SELECT users.gender,

               'Youngest' as Age\_Status,

               users.first\_name,

               users.last\_name,

               users.age

          FROM `bigquery-public-data.thelook\_ecommerce.users` as users

    INNER JOIN Female as GG ON GG.theyoungest = users.age

         WHERE users.gender = 'F'

      ORDER BY gender asc, age asc

#Question 4 : Get the top 5 most profitable product and its profit detail breakdown by month

WITH Base AS

(

        SELECT DATE\_TRUNC(DATE(Ord.created\_at),Month) as Month,

               Ord.product\_id,

               I.product\_name,

               sale\_price as Sales,

               I.cost as Cost,

               sale\_price-I.cost as Profit,

          FROM `bigquery-public-data.thelook\_ecommerce.order\_items` as Ord

     LEFT JOIN `bigquery-public-data.thelook\_ecommerce.inventory\_items` as I

            ON Ord.inventory\_item\_id = I.id

         WHERE Ord.Status = 'Complete' AND ord.created\_at BETWEEN '2019-01-01' AND '2022-04-30'

      ORDER BY 1

),

Base2 AS

(

        SELECT Month,

               Product\_id,

               product\_name,

               SUM(Sales) AS Sales,

               SUM(Cost) AS Cost,

               SUM(Profit) AS Profit,

          FROM Base

      GROUP BY 1,2,3

),

MAIN AS

(

        SELECT \*, RANK() OVER(PARTITION BY Month ORDER BY Profit DESC) As Rank

          FROM Base2

)

        SELECT \*

          FROM MAIN

         WHERE RANK <=5

      ORDER BY Month

#Question 5 : Create a query to get Month to Date of total revenue in each product categories of past 3 months (current date 15 April 2022), breakdown by date. Also calculate daily revenue each product

WITH Apr\_Daily AS

(

        SELECT DATE (created\_at) as Month,

               category,

               SUM(sale\_price) as Revenue,

          FROM `bigquery-public-data.thelook\_ecommerce.order\_items` AS I

          JOIN `bigquery-public-data.thelook\_ecommerce.products` AS P ON product\_id = P.id

         WHERE status = 'Complete' AND created\_at BETWEEN '2022-04-01' AND '2022-04-16'

      GROUP BY 2,1

      ORDER BY 1

),

March\_Daily AS

(

        SELECT DATE (created\_at) as Month,

               category,

               SUM(sale\_price) as Revenue,

          FROM `bigquery-public-data.thelook\_ecommerce.order\_items` AS I

          JOIN `bigquery-public-data.thelook\_ecommerce.products` AS P ON product\_id = P.id

         WHERE status = 'Complete' AND created\_at BETWEEN '2022-03-01' AND '2022-03-16'

      GROUP BY 2,1

      ORDER BY 1

),

Feb\_Daily AS

(

        SELECT DATE (created\_at) as Month,

               category,

               SUM(sale\_price) as Revenue,

          FROM `bigquery-public-data.thelook\_ecommerce.order\_items` AS I

          JOIN `bigquery-public-data.thelook\_ecommerce.products` AS P ON product\_id = P.id

         WHERE status = 'Complete' AND created\_at BETWEEN '2022-02-01' AND '2022-02-16'

      GROUP BY 2,1

      ORDER BY 1

)

        SELECT Month,

               category AS Product\_category,

               SUM(Revenue)

               OVER (PARTITION BY category, DATE\_TRUNC(Month, Month) ORDER BY Month) as Revenue

          FROM Apr\_Daily

     UNION ALL

        SELECT Month,

               category AS Product\_category,

               SUM(Revenue)

               OVER(PARTITION BY category, DATE\_TRUNC(Month, Month) ORDER BY Month) as Revenue

          FROM March\_Daily

     UNION ALL

        SELECT Month,

               category AS Product\_category,

               SUM(Revenue)

               OVER(PARTITION BY category, DATE\_TRUNC(Month, Month) ORDER BY Month) as Revenue

          FROM Feb\_Daily

      ORDER BY 2,1 DESC