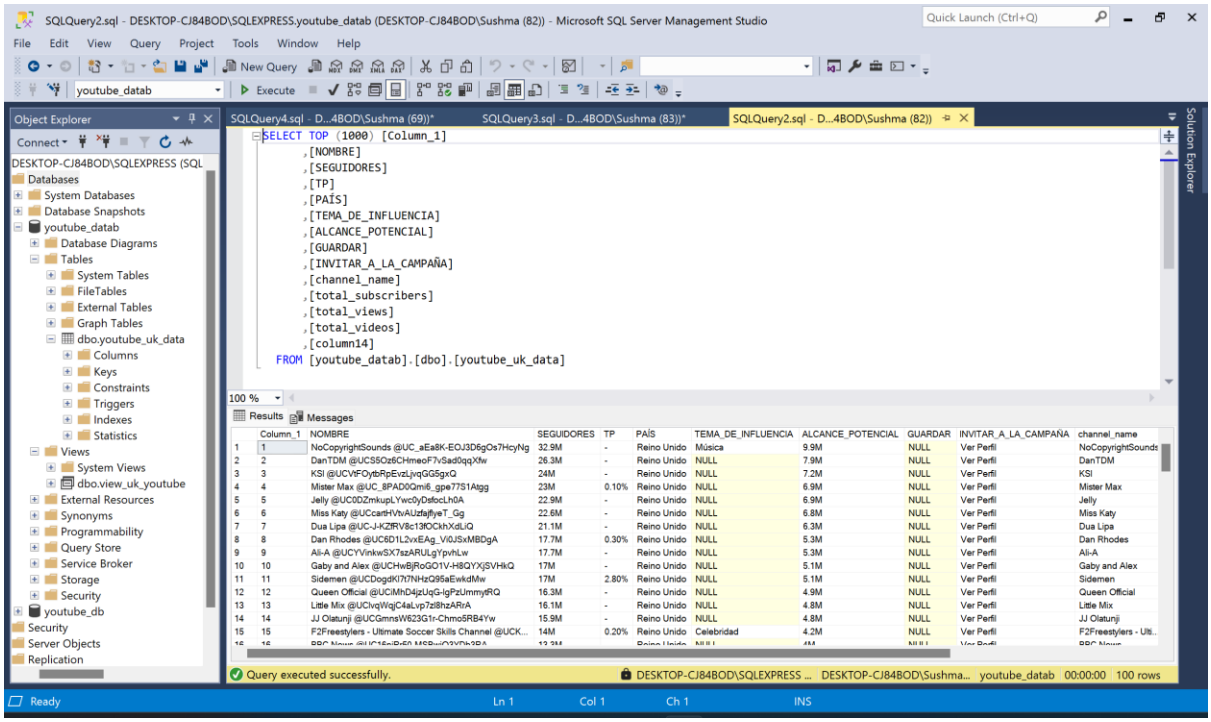
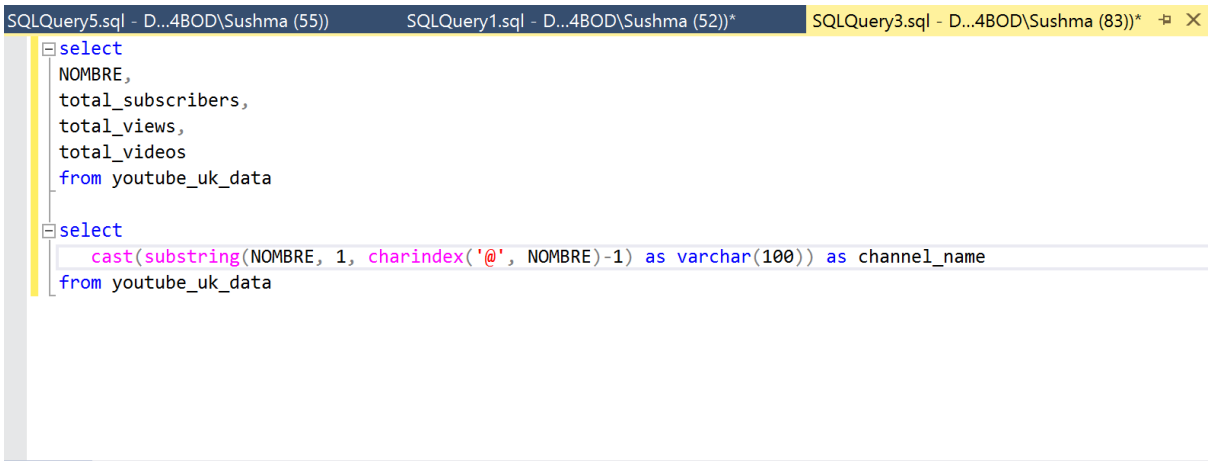
YOUTUBE ANALYSIS

DATA CLEANING AND CREATING A VIEW





Checking the data

SQLQuery5.sql - D...4BOD\Sushma (55)) SQLQuery1.sql - D...4BOD\Sushma (52))*

```
SELECT TOP (1000) [channel_name]
      ,[total_subscribers]
      ,[total_videos]
      ,[total_views]
  FROM [youtube_datab].[dbo].[view_uk_youtube]

--1. row count check

SELECT count(*) AS no_of_rows
FROM view_uk_youtube

--2.column count check

SELECT count(*) AS column_count
FROM
  INFORMATION_SCHEMA.COLUMNS
WHERE
   TABLE_NAME = 'view_uk_youtube'
```

```
--3.data type check

SELECT
    COLUMN_NAME,
    DATA_TYPE
FROM
     INFORMATION_SCHEMA.COLUMNS
WHERE
     TABLE_NAME = 'view_uk_youtube'

--4.Duplicates check

SELECT channel_name,
      COUNT(*) AS duplicate_count
FROM view_uk_youtube
GROUP BY channel_name
HAVING COUNT(*) >1
```

ROI ANALYSIS FOR THE CAMPAIGN :

Conversion rate is 2%

Production cost is $5 per unit

Cost for the campaign is $13000

MOST VIEWS

SQLQuery7.sql - D...4BOD\Sushma (64))* SQLQuery2.sql - D...4BOD\Sushma (82)) SQLQuery4.sql - D...4BOD\Sushma (69))*

```
---- MOST VIEWS

DECLARE @conversionrate FLOAT= 0.02;  --conversion rate is 2%
DECLARE @productcost MONEY = 5.0;   --Production cost is $5 per unit
DECLARE @campaigncost MONEY = 13000.0; --Cost of the campaign
WITH ChannelData AS (
    SELECT
        channel_name,
        total_views,
       total_videos,
        ROUND(CAST(total_views AS FLOAT) / total_videos, -4) AS avg_views_per_video
    FROM view_uk_youtube
    )
SELECT
    channel_name,
    avg_views_per_video,
    (avg_views_per_video * @conversionrate) AS potential_units_sold_per_video,
    (avg_views_per_video * @conversionrate * @productcost) AS potential_revenue_per_video,
    (avg_views_per_video * @conversionrate * @productcost) - @campaigncost AS net_profit
FROM
    ChannelData

WHERE
    channel_name IN ('Mister Max', 'DanTDM', 'Dan Rhodes')
ORDER BY
    net_profit DESC;
```

100 % Results Messages

	channel_name	avg_views_per_video	potential_units_sold_per_video	potential_revenue_per_video	net_profit
1	Mister Max	14060000	281200	1406000	1393000
2	Dan Rhodes	11150000	223000	1115000	1102000
3	DanTDM	5340000	106800	534000	521000

TOP SUBSCRIBERS

SQLQuery7.sql - D...4BOD\Sushma (64))* SQLQuery2.sql - D...4BOD\Sushma (82)) SQLQuery4.sql - D...4BOD\Sushma (69))*

```
    --top subscribers

DECLARE @conversionRate FLOAT = 0.02;        -- The conversion rate @ 2%
DECLARE @productCost FLOAT = 5.0;            -- The product cost @ $5
DECLARE @campaignCost FLOAT = 13000.0;       -- The campaign cost
WITH ChannelData AS (
    SELECT
        channel_name,
        total_views,
        total_videos,
        ROUND((CAST(total_views AS FLOAT) / total_videos), -4) AS rounded_avg_views_per_video
    FROM
       view_uk_youtube
)
SELECT
    channel_name,
    rounded_avg_views_per_video,
    (rounded_avg_views_per_video * @conversionRate) AS potential_units_sold_per_video,
    (rounded_avg_views_per_video * @conversionRate * @productCost) AS potential_revenue_per_video,
    ((rounded_avg_views_per_video * @conversionRate * @productCost) - @campaignCost) AS net_profit
FROM
    ChannelData
WHERE
    channel_name in ('NoCopyrightSounds', 'DanTDM', 'Dan Rhodes')
ORDER BY
    net_profit DESC
```

100 % Results Messages

	channel_name	rounded_avg_views_per_video	potential_units_sold_per_video	potential_revenue_per_video	net_profit
1	Dan Rhodes	11150000	223000	1115000	1102000
2	NoCopyrightSounds	6920000	138400	692000	679000
3	DanTDM	5340000	106800	534000	521000

Most videos

```sql
    --youtubers with most video

DECLARE @conversionRate FLOAT = 0.02;           -- The conversion rate @ 2%
DECLARE @productCost FLOAT = 5.0;               -- The product cost @ $5
DECLARE @campaignCostPerVideo FLOAT = 13000.0;   -- The campaign cost per video
DECLARE @numberOfVideos INT = 11;               -- The number of videos (11)
WITH ChannelData AS (
    SELECT
        channel_name,
        total_views,
        total_videos,
        ROUND((CAST(total_views AS FLOAT) / total_videos), -4) AS rounded_avg_views_per_video
    FROM
        view_uk_youtube)
SELECT
    channel_name,
    rounded_avg_views_per_video,
    (rounded_avg_views_per_video * @conversionRate) AS potential_units_sold_per_video,
    (rounded_avg_views_per_video * @conversionRate * @productCost) AS potential_revenue_per_video,
    ((rounded_avg_views_per_video * @conversionRate * @productCost) - (@campaignCostPerVideo * @numberOfVideos)) AS net_profit
FROM
    ChannelData
WHERE channel_name IN ('GRM Daily', 'Man City', 'YOGSCAST Lewis & Simon ')
ORDER BY
 net_profit DESC;
```

Results Messages

	channel_name	rounded_avg_views_per_video	potential_units_sold_per_video	potential_revenue_per_video	net_profit
1	YOGSCAST Lewis & Simon	710000	14200	71000	-72000
2	GRM Daily	510000	10200	51000	-92000
3	Man City	240000	4800	24000	-119000

Youtubers with most views

```sql
    --youtubers with most views

DECLARE @conversionRate FLOAT = 0.02;        -- The conversion rate @ 2%
DECLARE @productCost MONEY = 5.0;            -- The product cost @ $5
DECLARE @campaignCost MONEY = 13000.0;      -- The campaign cost @ $13000

WITH ChannelData AS (
    SELECT
       channel_name,
        total_views,
        total_videos,
        ROUND(CAST(total_views AS FLOAT) / total_videos, -4) AS avg_views_per_video
    FROM
        view_uk_youtube)
SELECT
    channel_name,
    avg_views_per_video,
    (avg_views_per_video * @conversionRate) AS potential_units_sold_per_video,
    (avg_views_per_video * @conversionRate * @productCost) AS potential_revenue_per_video,
   (avg_views_per_video * @conversionRate * @productCost) - @campaignCost AS net_profit
FROM
    ChannelData
WHERE
    channel_name IN ('Mister Max', 'DanTDM', 'Dan Rhodes')
ORDER BY
    net_profit DESC;
```

Results Messages

	channel_name	avg_views_per_video	potential_units_sold_per_video	potential_revenue_per_video	net_profit
1	Mister Max	14060000	281200	1406000	1393000
2	Dan Rhodes	11150000	223000	1115000	1102000
3	DanTDM	5340000	106800	534000	521000

POWERBI MEASURES

1.

```
avg views per video =
VAR sumoftotalviews = SUM(view_uk_youtube[total_views])
VAR sumoftotalvideos = SUM(view_uk_youtube[total_videos])
VAR avgviewspervideo = DIVIDE(sumoftotalviews, sumoftotalvideos, BLANK())
VAR finalavgviewsperVideo = divide(avgviewspervideo, 1000000, BLANK())

Return finalavgviewsperVideo
```

2.

```
Subscriber Engagement rate =
VAR sumoftotalsubscriber = SUM(view_uk_youtube[total_subscribers])
VAR sumoftotalvideos = SUM(view_uk_youtube[total_videos])
VAR subscriberengrate = DIVIDE(sumoftotalsubscriber, sumoftotalvideos, BLANK())

RETURN subscriberengrate
```

3.

```
Total Subscriber (M) =
var million = 1000000
var SumofSubscriber = sum(view_uk_youtube[total_subscribers])
var totalsubscriber = divide(SumofSubscriber,million)

return totalsubscriber
```

4.

```
total videos =
VAR totalVideos = SUM(view_uk_youtube[total_videos])

RETURN totalVideos
```

5.

```
Total Views(B) =
var billion = 1000000000
var sumoftotalviews = SUM(view_uk_youtube[total_views])
var totalviews = divide(sumoftotalviews, billion)

return totalviews
```

6.

```
views per subscriber =
VAR sumoftotalviews = SUM(view_uk_youtube[total_views])
VAR sumoftotalsubscribers = SUM(view_uk_youtube[total_subscribers])
VAR viewspersubscriber = DIVIDE(sumoftotalsubscribers, sumoftotalsubscribers, BLANK())

RETURN viewspersubscriber
```