



DIGITS N DATA



YOUTUBE

SQL

ANALYSIS

Next Slide ➤



Project Overview – YouTube SQL Analysis

This project aims to explore and derive key insights from a dataset of popular YouTube channels using SQL. The analysis is driven by real-world questions related to performance metrics, engagement statistics, and revenue generation across different channels and video content.

The dataset consists of four interconnected tables:

- **youtube_channels:** Channel-level metadata including subscribers, views, NOX score.
- **channel_videos:** Video-level details such as upload dates, views, likes, and comments.
- **channel_engagement:** Engagement metrics per channel – likes, comments, and shares.
- **channel_revenue:** Monthly and yearly revenue statistics per channel.



youtube_channels Table

channel_id	channel_name	category	subscribers	avg_views	nox_score
1	MrBeast	Entertainment	370M	187.67M	0.5
2	T-Series	Music	288M	336.88K	48.2
3	Cocomelon - Nursery Rhymes	Education	190M	2.95M	0.8
4	SET India	Entertainment	182M	15.61K	4.3
5	Vlad and Niki	Entertainment	135M	3.44M	24.3
6	Kids Diana Show	Entertainment	131M	4.2M	8.5
7	Like Nastya	Entertainment	126M	15.04M	4.8
8	Stokes Twins	People & Blogs	116M	77.77M	5.4
9	Zee Music Company	Music	115M	1.37M	467.1
10	PewDiePie	Gaming	110M	2.98M	0.9

channel_engagement Table

engagement_id	channel_id	likes_per_video	comments_per_video	shares_per_video
1	1	450,000.50	12,000.75	30,000.25
2	2	300,000.20	10,000.00	25,000.10
3	3	200,000.15	8,000.25	20,000.50
4	4	150,000.75	5,000.60	18,000.80
5	5	180,000.50	4,000.90	15,000.60
6	6	175,000.35	3,500.75	14,000.40
7	7	190,000.45	4,500.85	17,000.20
8	8	210,000.30	7,000.25	20,000.35
9	9	50,000.00	2,000.75	5,000.80
10	10	120,000.90	6,000.35	9,000.50

channel_videos Table

video_id	channel_id	video_title	upload_date	views	likes	comments
1	1	Epic Challenge Video	2024-03-01	15,000,000	500,000	10,000
2	2	Best Bollywood Songs	2024-02-15	3,000,000	250,000	5,000
3	3	Kids Learning ABC	2024-01-10	5,000,000	200,000	8,000
4	4	Comedy Skit Compilation	2024-03-05	2,000,000	150,000	3,000
5	5	Vlad and Niki Adventure	2024-02-22	6,000,000	180,000	4,000
6	6	Diana's New Toys	2024-01-30	4,000,000	175,000	3,500
7	7	Fun Family Vlog	2024-02-18	3,500,000	190,000	4,500
8	8	Crazy Stunts Compilation	2024-03-01	8,000,000	210,000	7,000
9	9	Top Music Hits 2024	2024-02-10	1,000,000	50,000	2,000
10	10	Gaming Livestream Highlights	2024-01-20	2,500,000	120,000	6,000

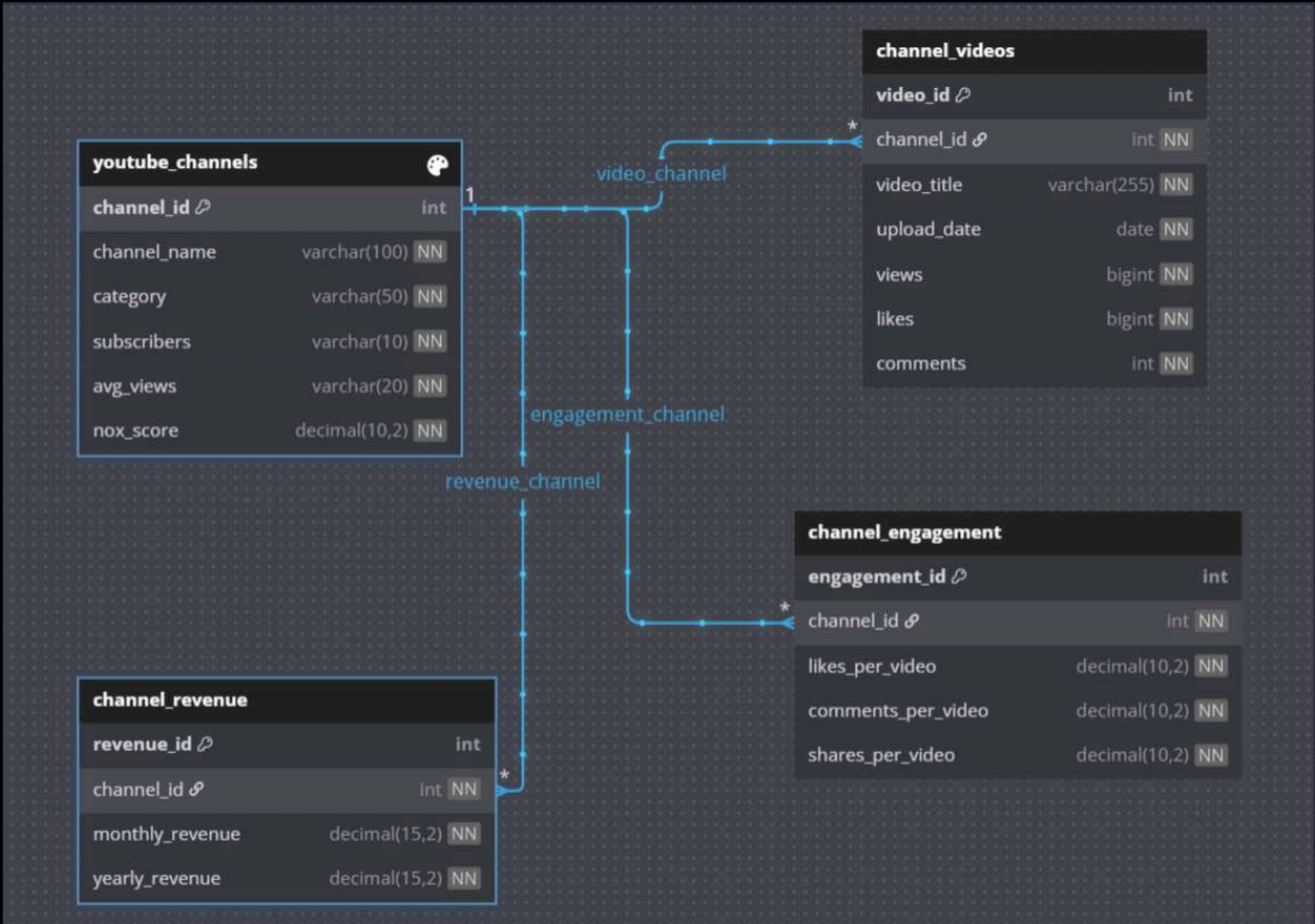
channel_revenue Table

revenue_id	channel_id	monthly_revenue	yearly_revenue
1	1	5,000,000.00	60,000,000.00
2	2	4,500,000.00	54,000,000.00
3	3	1,500,000.00	18,000,000.00
4	4	1,200,000.00	14,400,000.00
5	5	900,000.00	10,800,000.00
6	6	850,000.00	10,200,000.00
7	7	800,000.00	9,600,000.00
8	8	780,000.00	9,360,000.00
9	9	750,000.00	9,000,000.00
10	10	700,000.00	8,400,000.00

ENTITY-RELATIONSHIP DIAGRAM



SQL ANALYSIS








1. Get the top 5 most subscribed YouTube channels ?
2. Find the total yearly revenue of all channels ?
3. Find the video uploaded by 'MrBeast' ?
4. Get the average engagement (likes & comments) per video for each channel ?
5. List channels that have a NOX score higher than 10 ?
6. Find the most commented video ?
7. Count the total number of channels in each category ?
8. Search for channel names that start with "M" and end with "t" (using Like function) ?
9. Find videos uploaded between '2024-03-01' AND '2024-08-01' ?
10. Find the top 3 most liked videos on YouTube ?



```
-- 1 Get the top 5 most subscribed YouTube channels ?  
SELECT  
    channel_name AS Top_5_Youtube_Channels,  
    MAX(subscribers) AS Subscribers  
FROM youtube_channels  
GROUP BY channel_name  
ORDER BY Subscribers DESC  
LIMIT 5;
```

OUTPUT:-

Result Grid   Filter Rows: <input type="text"/> Export: 		
	Top_5_Youtube_Channels	Subscribers
▶	MrBeast	370M
	T-Series	288M
	Cocomelon - Nursery Rhymes	190M
	SET India	182M
	Vlad and Niki	135M



```
-- 2 Find the total yearly revenue of all channels ?
```

```
Select
```

```
    channel_name AS Channel_Name,
```

```
    SUM(yearly_revenue) as Yearly_Revenue
```




```
FROM youtube_channels
```

```
JOIN channel_revenue ON youtube_channels.channel_id = channel_revenue.channel_id
```

```
GROUP BY Channel_Name
```

```
ORDER BY Yearly_Revenue DESC;
```

OUTPUT:-

Result Grid   Filter Rows: <input type="text"/> Export: 		
	Channel_Name	Yearly_Revenue
▶	MrBeast	600000000.00
	T-Series	540000000.00
	Cocomelon - Nursery Rhymes	180000000.00
	SET India	144000000.00
	Vlad and Niki	108000000.00
	Kids Diana Show	102000000.00

Next Slide 



```
-- 3 Find the video uploaded by 'MrBeast' ?
```

```
SELECT
```

```
    yc.channel_name as Channel_Name,
```

```
    cv.video_title
```

```
FROM youtube_channels yc
```

```
LEFT JOIN channel_videos cv ON yc.channel_id = cv.channel_id
```

```
WHERE yc.channel_name = "MrBeast";
```

OUTPUT:-

Result Grid			Filter Rows:	
	Channel_Name	video_title		
▶	MrBeast	Epic Challenge Video		

Next Slide ➤



```
-- 4 Get the average engagement (likes & comments) per video for each channel ?
```

```
SELECT
    channel_name AS Channel_Name,
    Round(AVG(likes_per_video),0) AS Avg_Likes,
    Round(AVG(comments_per_video),0) AS Avg_Comments
FROM youtube_channels
LEFT JOIN channel_engagement ON youtube_channels.channel_id = channel_engagement.channel_id
GROUP BY channel_name
Order by Avg_Likes;
```

OUTPUT:-




Result Grid   Filter Rows: <input type="text"/> Export: 			
	Channel_Name	Avg_Likes	Avg_Comments
▶	Zee Music Company	50000	2001
	PewDiePie	120001	6000
	SET India	150001	5001
	Kids Diana Show	175000	3501
	Vlad and Niki	180001	4001
	Like Nastya	190000	4501
	Cocomelon - Nursery Rhymes	200000	8000

Next Slide 



```
-- 5 List channels that have a NOX score higher than 10 ?
SELECT
    channel_name AS Channel_Name,
    nox_score AS Nox_Score
FROM youtube_channels
WHERE nox_score > 10
ORDER BY Nox_Score DESC;
```

OUTPUT:-

Result Grid   Filter Rows: <input type="text"/> Export: 		
	Channel_Name	Nox_Score
▶	Zee Music Company	467.10
	T-Series	48.20
	Vlad and Niki	24.30



```
-- 6 Find the most commented video ?

SELECT
    CV.video_title AS Video_Title,
    SUM(CE.comments_per_video) AS High_Comments
FROM channel_videos CV
LEFT JOIN channel_engagement CE ON CV.channel_id = CE.channel_id
GROUP BY Video_Title
ORDER BY High_Comments DESC
LIMIT 1;
```

OUTPUT:-




Result Grid			Filter Rows:	Export:
	Video_Title	High_Comments		
▶	Epic Challenge Video	12000.75		

Next Slide ➤



```
-- 7 Count the total number of channels in each category ?
SELECT
    category as Category,
    COUNT(channel_name) AS Total_Channels
FROM youtube_channels
GROUP BY Category
Order by Total_Channels DESC;
```




OUTPUT:-

Result Grid   Filter Rows: <input type="text"/> Export: 		
	Category	Total_Channels
▶	Entertainment	5
	Music	2
	Education	1
	People & Blogs	1
	Gaming	1



```
-- 8 Search for channel names that start with "M" and end with "t" (using Like function) ?  
SELECT  
    channel_name AS Channel_Name  
FROM youtube_channels  
WHERE channel_name LIKE "M%" AND channel_name LIKE "%t";
```

OUTPUT:-

Result Grid			 Filter Rows:	
	Channel_Name			
	MrBeast			



```
-- 9 Find videos uploaded between '2024-03-01' AND '2024-08-01' ?
SELECT
    video_title AS Video_Title
FROM channel_videos
WHERE upload_date BETWEEN '2024-03-01' AND '2024-08-01';
```

OUTPUT:-

Result Grid		Filter Rows:	Export:
	Video_Title		
▶	Epic Challenge Video		
	Comedy Skit Compilation		
	Crazy Stunts Compilation		



```
-- 10 Find the top 3 most liked videos on YouTube ?

SELECT
    video_title AS Top_3_Videos,
    MAX(likes) AS Total_Likes
FROM channel_videos
GROUP BY Top_3_Videos
ORDER BY Total_Likes DESC
LIMIT 3;
```

OUTPUT:-

Result Grid			Filter Rows:	Export:
	Top_3_Videos	Total_Likes		
▶	Epic Challenge Video	500000		
	Best Bollywood Songs	250000		
	Crazy Stunts Compilation	210000		



Thank You!

SANJAY POLAMARASETTI

