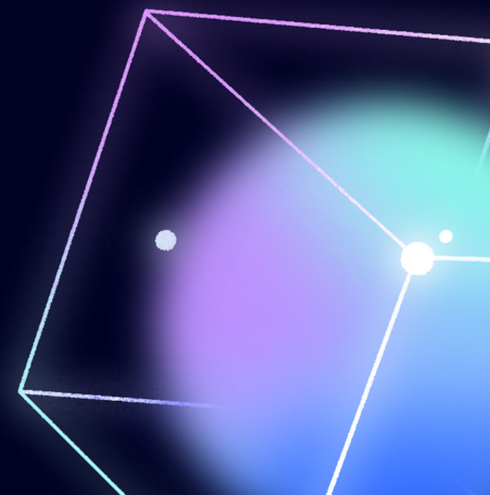
# GAME ANALYSIS WITH SQL

K.NIKHITHA


# INTRODUCTION TO SQL

SQL stands for Structured Query Language. It's a language used to interact with databases, allowing users to perform tasks like retrieving data, updating records, and managing database structure.



GAME ANALYSIS REFERS TO THE PROCESS OF EXAMINING VARIOUS ASPECTS OF A GAME, SUCH AS PLAYER BEHAVIOR, LEVEL DESIGN, AND PERFORMANCE METRICS, TO GAIN INSIGHTS INTO HOW THE GAME IS PLAYED AND HOW IT CAN BE IMPROVED.

START

In this project, we will be working with a dataset related to a game. The dataset includes two tables: Player Details and Level Details.
There are 15 questions for which you have to find the answers by writing SQL queries.

```
-- Q1) Extract P_ID,Dev_ID,PName and Difficulty_level of all players
-- at level 0
-- Q2) Find Level1_code wise Avg_Kill_Count where lives_earned is 2 and atleast
--    3 stages are crossed
-- Q3) Find the total number of stages crossed at each diffuculty level
-- where for Level2 with players use zm_series devices. Arrange the result
-- in decsreasing order of total number of stages crossed.
-- Q4) Extract P_ID and the total number of unique dates for those players
-- who have played games on multiple days.
-- Q5) Find P_ID and level wise sum of kill_counts where kill_count
-- is greater than avg kill count for the Medium difficulty.
-- Q6)  Find Level and its corresponding Level code wise sum of lives earned
-- excluding level 0. Arrange in asecending order of level.
-- Q7) Find Top 3 score based on each dev_id and Rank them in increasing order
-- using Row_Number. Display difficulty as well.
-- Q8) Find first_login datetime for each device id
-- Q9) Find Top 5 score based on each difficulty level and Rank them in
-- increasing order using Rank. Display dev_id as well.
-- Q10) Find the device ID that is first logged in(based on start_datetime)
-- for each player(p_id). Output should contain player id, device id and
-- first login datetime.
-- Q11) For each player and date, how many kill_count played so far by the player. That is, the total number of games played -- by the player until that date.
-- a) window function
-- b) without window function
-- Q12) Find the cumulative sum of stages crossed over a start_datetime
-- Q13) Find the cumulative sum of an stages crossed over a start_datetime
-- for each player id but exclude the most recent start_datetime
```

# DATASET

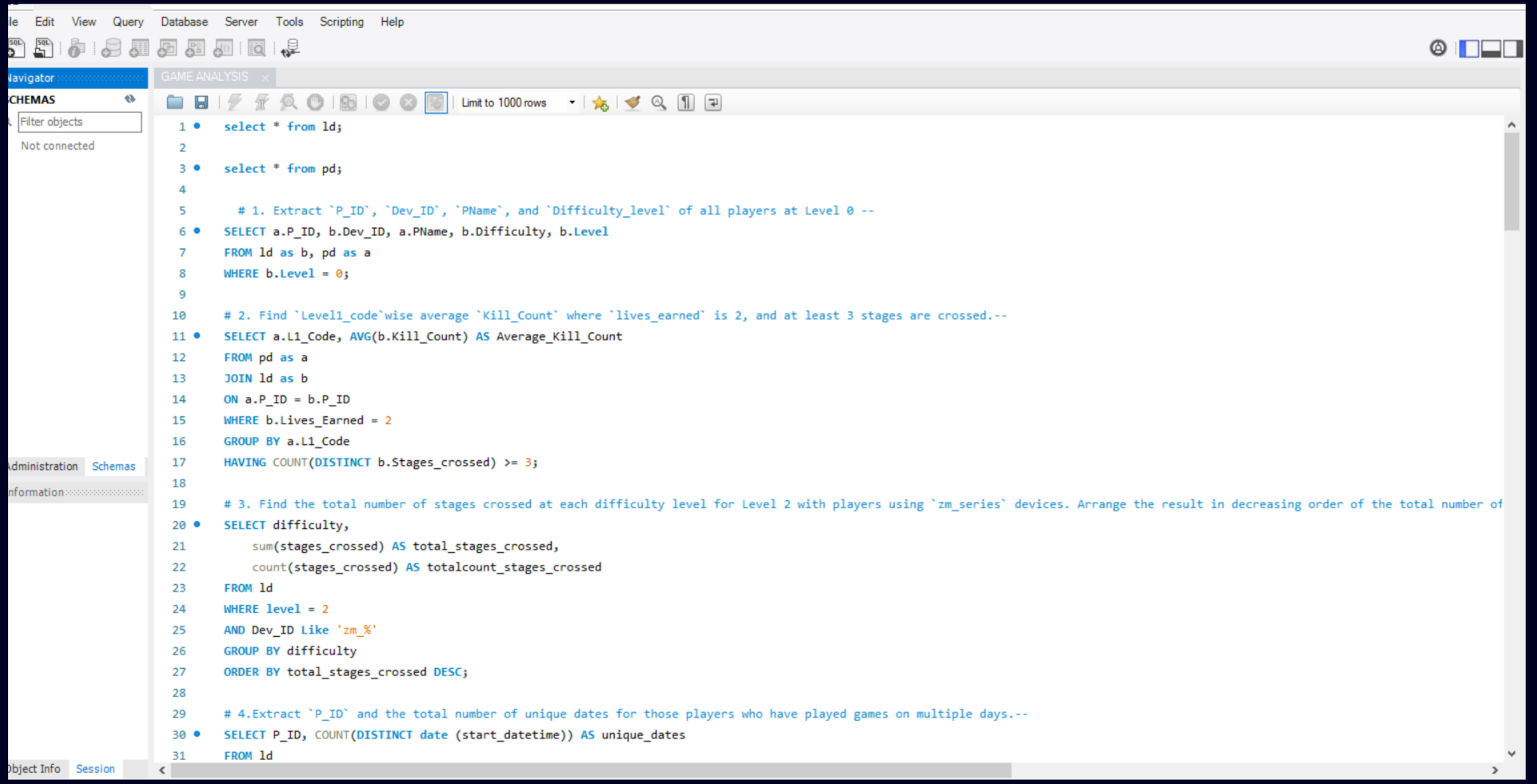
## PLAYER DETAILS

- P_ID: Player ID
- PName: Player Name
- L1_status: Level 1 Status
- L2_status: Level 2 Status
- L1_code: Systemgenerated Level 1 Code
- L2_code: Systemgenerated Level 2 Code

## LEVEL DETAILS

- P_ID: Player ID
- Dev_ID: Device ID
- start_time: Start Time
- stages_crossed: Stages Crossed
- level: Game Level
- difficulty: Difficulty Level
- kill_count: Kill Count
- headshots_count: Headshots Count
- score: Player Score
- lives_earned: Extra Lives Earned

Following are the 2,3 answers examples of SQL queries.



# CONCLUSION

"INSIGHTS FROM THIS ANALYSIS CAN INFORM GAME DESIGN DECISIONS AND IMPROVE PLAYER ENGAGEMENT."

# THANK YOU!