Senior Data Developer - Home assignment

In our company, we collect and store game events in BigQuery DWH.
In this exercise, we want to process data that we've collected and serve it with API to the game backend.

1. Build a SQL-based pipeline to calculate a table that will summarise some life-time properties of players. We call these properties "User Attributes".

   The end product of this pipeline is an aggregated table called **user_panel**.
   The table is a life-time summary of each player, meaning that it has **1 row per user**.
   It contains 1 column for player id and N columns, one per attribute
   The data pipeline is batch, it runs once in four hours, and recalculates all of the data.

   Use the game events provided in the raw data files, to implement the following user attributes:

   - **country** - last country we see the player in
   - **avg_price_10** - avg. of last (maximum) 10 deposits prices
   - **last_weighted_daily_matches_count_10_played_days** - metric is a weighted average of the number of matches a player played on their last 10 active days. The weighting is assigned based on recency: the most recent day gets the highest weight of 10, and the oldest of these days gets the lowest weight of 1. If there are fewer than 10 active days available, the weights are adjusted to range from 1 to the number of days the player was active, ensuring that recent activity is given greater emphasis.
   - **active_days_since_last_purchase** - amount of active days (active day = day in which we see some data from the player, which means he logged in) since last purchase to current moment. If a player purchased today - it is 0.
   - **score_perc_50_last_5_days** - the median score the player reached in matches in the last 5 calendar days. If he has no matches in the last 5 days - return null

Senior Data Developer - Home assignment

The end product should look like the following:

player_id (unique)	**country**	**avg_price_10**	**last_weighted_daily_matches_count_10_played_days**	**active_days_since_last_purchase**	**score_perc_50_last_5_days**
64b7cb0ee03e432f626c5d94	US	120	7.59	1	8560

Further instructions:

1. The raw data is provided as parquet files in the assignment email.
2. Load the data into BigQuery, this is the data warehouse that we want to use in this assignment
3. You can use the free version of BigQuery or the 300$ credits you are given by GCP in a new project, that should be more than enough.
4. To implement the pipeline,you may use dbt (https://www.getdbt.com/), but other tools are welcome as well.
5. Bonus: Make the queries as cost efficient as possible. Use any BigQuery features you want to do that.

Senior Data Developer - Home assignment

2. You are required to design and implement a system that retrieves specific attributes for users that are available in the "user_panel". This system will serve as a critical component in our backend architecture, interfacing with other services.

   Function Specification:

   - **GetAttribute(player_id, attribute_name)**: This API endpoint must take two parameters: `player_id` (a unique identifier for the user) and `attriubute_name` (the name of the attribute to be retrieved). It should return the value of the attribute for the given player ID.

   System requirements:

   - The API has to have ~50ms latency on average
   - The service has to be scalable, design it for scale 100 calls/sec
   - Assume that the process that you implemented in exercise once every four hours
   - When the data is re-calculated, it has to available to the backend after at most 15 minutes
   - The system has to be as stable as possible!
   - Design the service to handle rapid growth of the user_panel table, assume that it might reach a size of tens of gigabytes.
   - Consider cost effectiveness of the implementation

   Use python for your implementation.
   You can use any framework/library/technology you want, and add any other components you want.

   If you have questions or need clarifications, you can reach out to:
   netanel@play-perfect.com