

## Data Engineer assignment for the RCA team

RCA stands for `Real-time Customer data and Activation`. Streaming data is in the core of everything that we do. As such, this assignment is meant for you to showcase your skills in this matter and consists of 2 parts; streaming data and aggregating tables. Our platform is entirely based on Google Cloud, so please showcase your knowledge about this cloud provider.

### Stream data

Please create a super simple python code that will pull data from a Cloud Pub/Sub subscription, transform the data into another schema and write that data into a Bigquery table.

**Please share:**

- The Python code
- The DDL statement, or terraform file(s), that you used to create the Bigquery table
- A diagram that represents the ideal setup in your view. For example, give a good indication of what should happen if a transformation throws an exception.

### Aggregated tables

Using the setup mentioned above, you created a “streaming table” that contains multiple events related to the same entity. For example, if the exercise above was around `orders`, you would have an `orders\_events` table with multiple status relative to the lifetime of a single order.

A stakeholder mentions that they are having issues because they do not know how to properly handle the granularity of the `order\_events` table and asks you to create a consolidated `orders` table where only the latest status is available. The stakeholder would like to have this table updated every hour. Assume that the table to be created must be partitioned by `DAY(timestamp)`, where the main timestamp is when the order was created, not when it was last modified.

Please propose at least two solutions for how to create and keep the new consolidated table updated. You can assume that there is no need for aggregated arrays (ARRAY\_AGG) in the construction of the table.

**Please share:**

- A diagram with all the components that you would use to schedule/orchestrate any change.
- The SQL or Python code that you propose to keep the, in the mentioned example, `orders` table updated.

### Conversion upload

Now, the business team has requested that whenever an order is completed, a corresponding conversion should be uploaded to Google Ads for campaign attribution.

**Please share:**

- Add to the diagram how the Google Ads upload happens.
- Which other tables will you read from, what other types of data will you need?
- List the required fields that must be provided to the Google Ads API for a successful conversion upload.