**Fresh Prints GCP Take Home Task**

Table of Contents

[Google Sheet 1](#_Toc86125731)

[BQ External Table 2](#_Toc86125732)

[Postgres Instance 4](#_Toc86125733)

[Cloud Functions 5](#_Toc86125734)

[Cloud Composer 5](#_Toc86125735)

[QA Test Cases 7](#_Toc86125736)

This document is created to share the execution details of the task. All the source code is present in the public github repo “https://github.com/chandrasekharbattula/freshprints\_test”

## Google Sheet

Here’s the link to google sheet with some sample data used to execute this task.

Google Sheet : <https://docs.google.com/spreadsheets/d/1iv05TAQHjJy0AA-GPH4rgVtb1rd5PPSSyrFBLkyGwWg/edit?usp=sharing>

Few columns are created and few columns are derived from the existing columns.

Here’s a screen shot of the google sheet that show sample employee data.

Graphical user interface, application, table, Excel

Description automatically generated

Here’s a screen shot that shows the formulars used to derive the employee age.

Graphical user interface, application, table

Description automatically generated

Similarly couple of other fields that are derived from the existing columns are Employee\_Experince\_IN\_Months

Years\_Left\_For\_Retierment

Employee\_Annual\_Salary

Employee\_Full\_Name

## BQ External Table

To access the data from Google Sheet, I have created an external BQ table so that it is easy to query data and process in the ETL pipelines.

The following screen shots shows the details of the external BQ data.

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Table

Description automatically generated with medium confidence

Text

Description automatically generated

## Postgres Instance

A PostgresSQL instance is created through Cloud SQL in GCP. Here’s the screen shot that shows the instance details and the employee data base where the destination employee\_details is created.

Graphical user interface, application

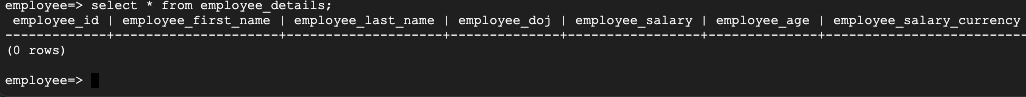
Description automatically generated

Graphical user interface, application

Description automatically generated

Text

Description automatically generated



A picture containing text

Description automatically generated

## Cloud Functions

Finally each individual the ETL task is executed through cloud function. The below screen shots shows the cloud functions that are created and deployed on to GCP.

A custom service account is created with the required permissions to execute the function.

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, application

Description automatically generated

## Cloud Composer

Finally a cloud composer instance is created to schedule and trigger the cloud functions and form a ETL pipeline.

Graphical user interface, application

Description automatically generated

Graphical user interface, text, application, chat or text message

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Final Result

Graphical user interface, text

Description automatically generated

## QA Test Cases

Below are the test case that needs to be executed to verify the data quality between source and destination.

1. Check for duplicate data - Make sure there is no duplicate data in the destination table.
2. Check for Count of rows – The count of rows should In the source and destination should match after the pipeline is executed after the invalid records are excluded from source.
3. Check the values – Ensure that the data of each columns and rows matches with the source after the transformations.
4. Check to ensure no employees that have retired are loaded to the destination table
5. Check to ensure no null values or spaces are present in the destination table.