CS 329E Project 7, due Thursday, 03/21.

In this project, we complete the automation sprint by implementing referential integrity checks, creating the final target tables, and orchestrating the end-to-end pipeline.

Objectives

- Create Cloud Composer environment
- Develop an Airflow pipeline that creates the primary keys and foreign keys on the staging tables
- Develop an Airflow pipeline that validates the primary key and foreign key constraints on the staging tables
- Develop an Airflow pipeline that creates the final target tables in the consumption layer
- Develop an Airflow controller that calls each one of the sub controllers to automate the entire pipeline from ingest to consumption.
- Successfully execute the end-to-end pipeline. Reduce the level of task parallelism if necessary.
- Delete and re-create your Cloud Composer environment to reduce billing charges

Implementation Guidelines

Please follow these guidelines when implementing your solution:

- Create a custom Composer cluster with **3.25G** of RAM for the worker nodes (instead of the default 2G).
- Store the target tables in a new dataset in BigQuery. The name of the consumption dataset should follow our convention of **[domain]_*csp_af***.
- Ensure that the target tables generated through Airflow match the ones created from Colab in the consumption layer. Both the schema and contents should match.
- Use the 5 provided code samples as a starting point for your own DAGs: **p7-key-controller.py**, **p7-create-pk.py, p7-create-fk.py, p7-target-controller.py, and p7-master-controller.py.**
- If you encounter non-determinist failures when running your DAG, it is likely a resource issue. You can reduce the number of concurrent tasks using the `DAG.currency` parameter as shown here (line 29).
- Take a screenshot of your master controller execution run showing that all tasks completed successfully. Name the file **master-controller-run.png**.
- When not actively developing, delete your Composer instance to avoid burning through your GCP credits. Note: there is no way to stop and restart a Composer instance.
- Publish to your repo: **key-controller.py, create-pk.py, create-fk.py, target-controller.py, master-controller.ipy,** and **master-controller-run.png**.

CS 329E Project 7 Rubric
Due Date: 03/21/24

`key-controller.py` has all required info and correctly populates keys **-10** did not update global variables **-10** for each TriggerDagRunOperator object missing **-15** if upload .ipynb instead of .py **-40** missing file	40
`target-controller.py` creates dataset and populates all tables correctly **-5** did not update global variables **-5** for each table missing **-5** for each BigQueryInsertJobOperator object missing **-10** if upload .ipynb instead of .py **-25** missing file	25
`create-pk.py, create-fk.py, master-controller.py` have all required info **-5** missing _create_fk and _check_ref_integrity methods in create-fk.py **-5** missing _create_pk and _check_pk methods in ,create-pk.py **-5** missing TriggerDagRunOperator objects in master-controller.py for all controllers **-5** for each missing file	10
Google Cloud BigQuery bucket has properly loaded all keys **-5** for each missing key or table **-10** if tables not under “csp_af” dataset **-20** missing file	15
`master-controller-run.png` shows proper proof of Airflow controller execution **-10** missing file	10

`submission.json` submitted into Canvas. Your project **will not** be graded without this submission. The file should have the following schema:
 	
`{`
`  "commit-id": "your most recent commit ID from Github",`
`  "project-id": "your project ID from GCP"`
`}`
 	
Example:
 	
`{`
`  "commit-id": "dab96492ac7d906368ac9c7a17cb0dbd670923d9",`
`  "project-id": "some-project-id"`
`}`
	Required

Total Credit:	**100**