**ETL Project Report**

Marylu, Pooja, Peter

* The sources of data that we extracted from:

|  |  |
| --- | --- |
| /var/folders/cd/f8rfwlfj52b1s20f3qwwvpt80000gn/T/com.microsoft.Word/Content.MSO/745B3319.tmp | Image result for happiness |
| **STARBUCKS DATASET**  <https://www.kaggle.com/starbucks/store-locations> | **HAPPINESS DATASET**  <https://www.kaggle.com/PromptCloudHQ/world-happiness-report-2019> |

**Extract:**

Starbucks Raw Data Set:

A screenshot of a cell phone

Description automatically generated

Happiness Raw Data Set:

A screenshot of a cell phone

Description automatically generated

**Transform:**

Steps to clean the DataFrame:

* Eliminated unnecessary columns from the DataFrame
* Searched for and removed duplicates

A screenshot of a cell phone

Description automatically generated

* Converted Country Code using PyCountry
* Normalized Country names, e.g. “Viet Nan to Vietnam”

A screenshot of a social media post

Description automatically generated

* Re-named the headers to match the Postgres tables

A screenshot of a cell phone

Description automatically generated

**Load:**

Created relational database in Postgres

Created tables: Starbucks and Happiness

A screenshot of a cell phone

Description automatically generated

Established an engine connection

Added security feature to prompt the user for UN/PW using GetPass

A screenshot of a cell phone

Description automatically generated

Checked for tables in Jupyter Notebook

Used Pandas to load csv converted DataFrame into database  
Appended data via Jupyter notebook into Postgres tables using SQL Alchemy

Confirmed data has been added by querying the tables in Jupyter Notebook

A screenshot of a cell phone

Description automatically generated

Created a join query to view Top 10 results displaying Country, Count of Starbucks Stores and Happiness Rank

A screenshot of a cell phone

Description automatically generated

Further areas for investigation include:

* The Starbucks location dataset was one year old. We used a dataset someone else had scrapped and uploaded onto Kaggle. We would have manually scrapped the Starbucks location webpage if the project had been analysis-focused.
* The most difficult aspect was converting and matching countries. Some of the country names derived from PyCountry didn’t match what was the Happiness dataset.
* The easy aspect was the Postgres querying was the easiest, since we all use SQL on a daily basis at work.