**Crowdfunding ETL – Project 2**

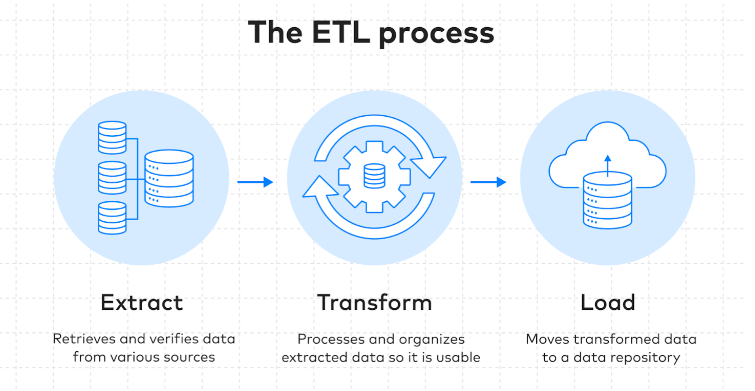
Nicholas Buse

Sam Hoemann

Marty Thompson

**Introduction**

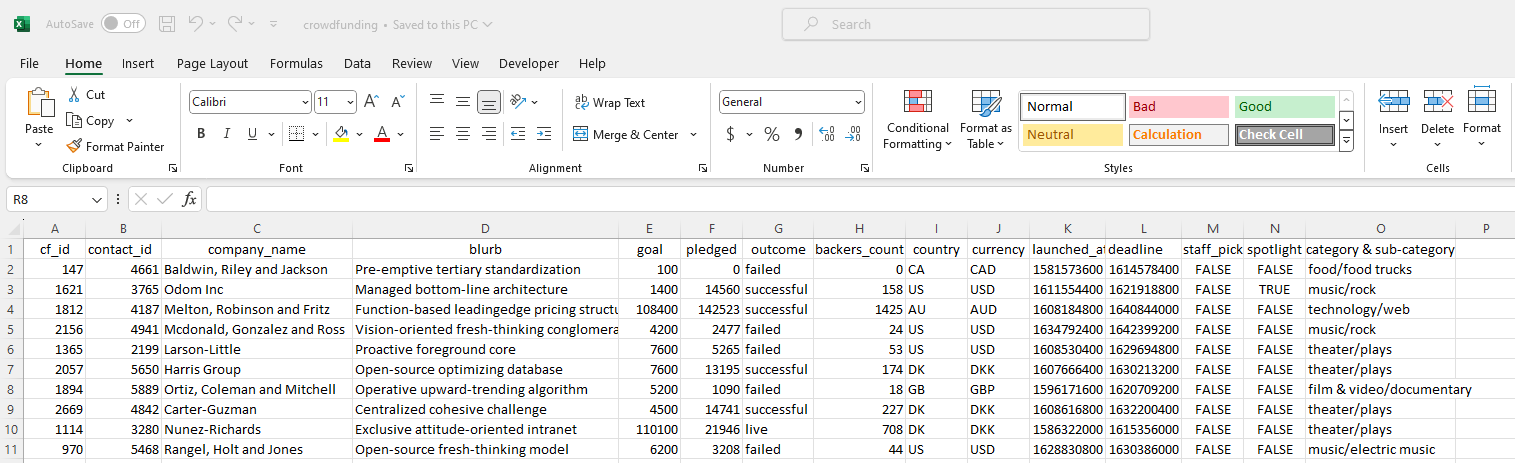
What is ETL? ETL is the process of extracting, transforming, and loading data from multiples sources into a single central data repository. This is the process of taking raw data, cleaning and organizing the information into a usable format, and then loading it into a data repository. We can then access the data to improve business intelligence, generating analytics and reporting to better steer company decisions. This project is a showcase of the ETL process, culminating in an analysis of the data using SQL queries and data visualizations.



**Extraction**

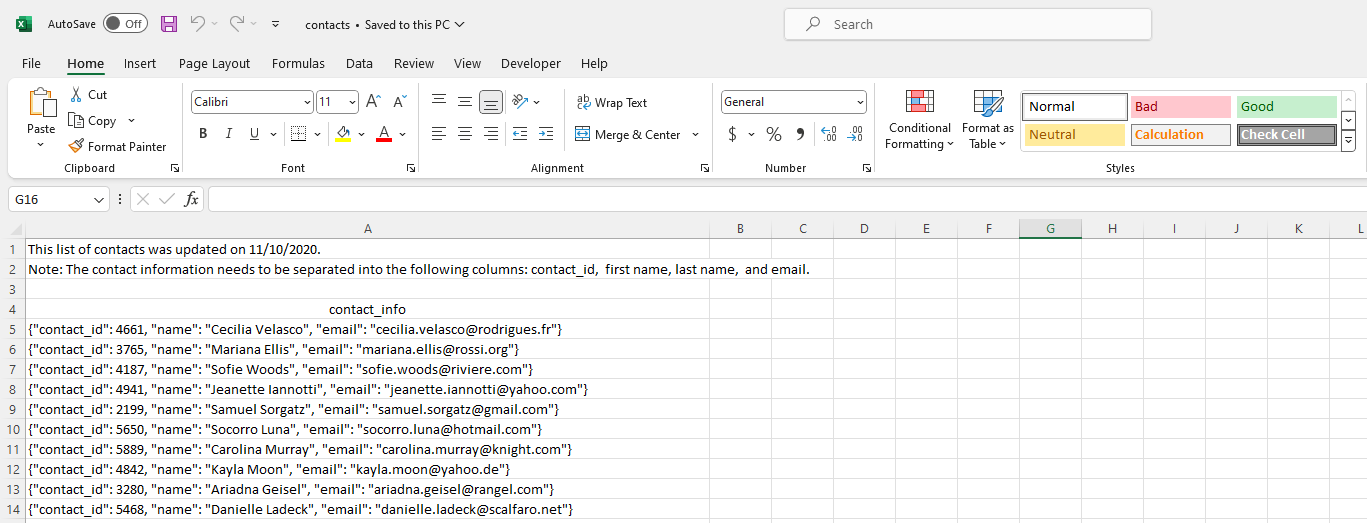
In the Extraction phase, we were given raw data in the form of two excel files found in the Resources folder, to simulate retrieval of raw data from multiple sources. Each was saved in a different format for us to extract data into usable files for uploading into our database.

The first, was “Resources/crowdfunding”:



This excel format had information clearly separated by columns. We were to use this file to separate information needed to generate our category, subcategory, and campaign tables. Our contacts table would require the second resource file, saved in JSON format.

The second, was “Resources/contacts”:



**Transformation**

After receiving our extracted data, we then transitioned into the Transformation phase. To begin this phase, we designed a table schema for our database using QuickDBD (quickdatabasediagram.com). As part of this creation, we generated an Entity Relationship Diagram to isolate necessary information for each required table and a SQL Source File for the scripting necessary to create tables within our database software, Postgres.

Entity Relationship Diagram, “QUICKDBD/QuickDBD-Project 2.png”

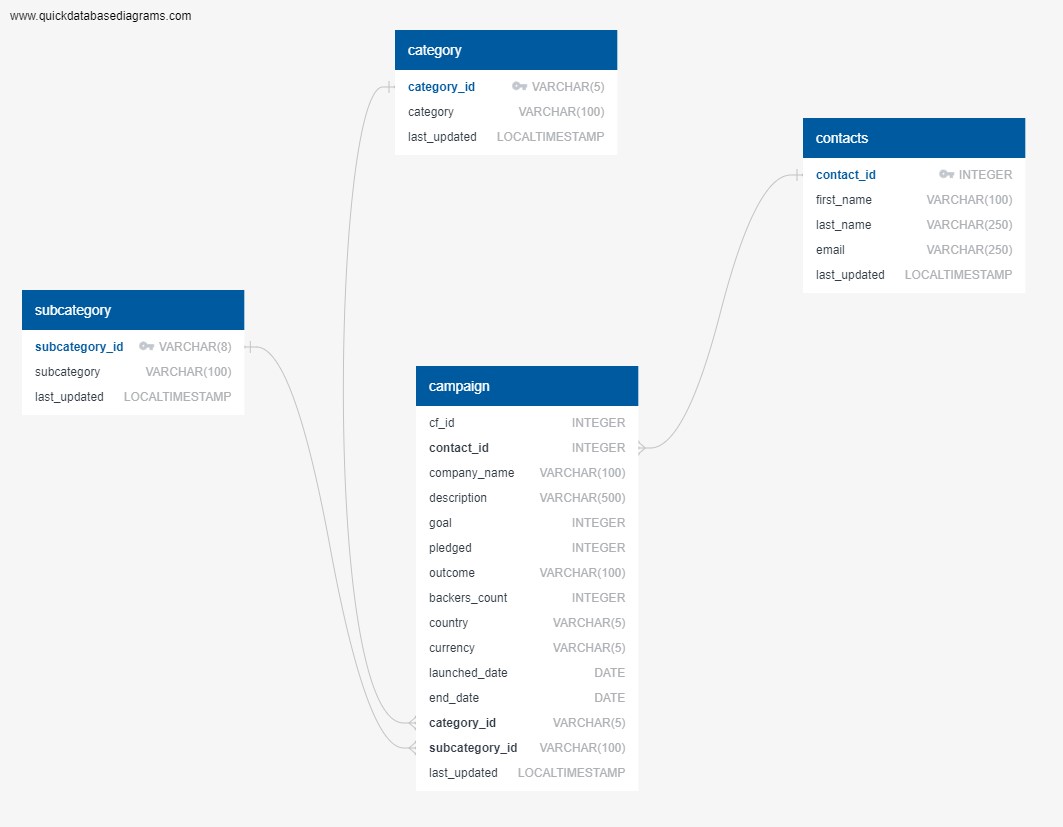
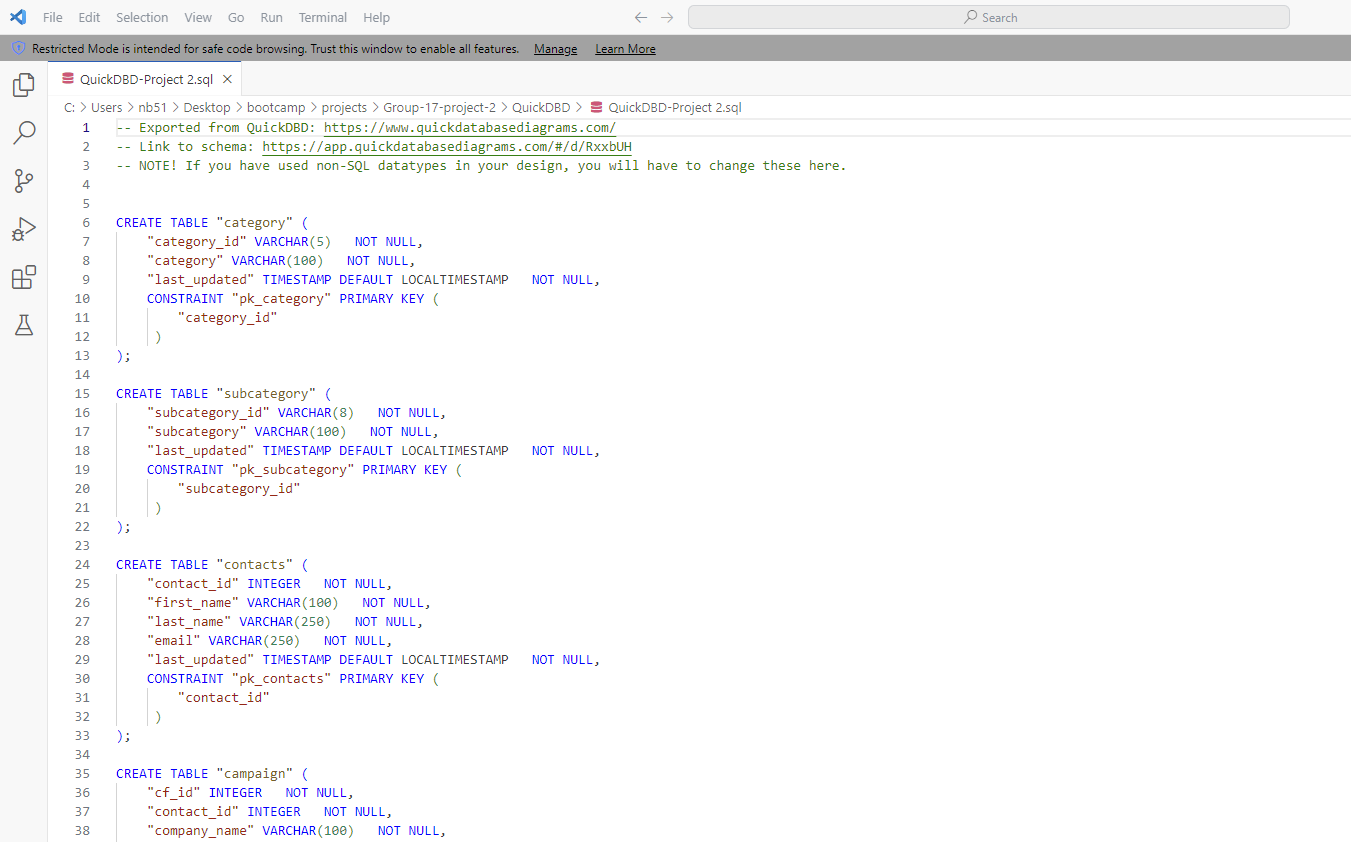
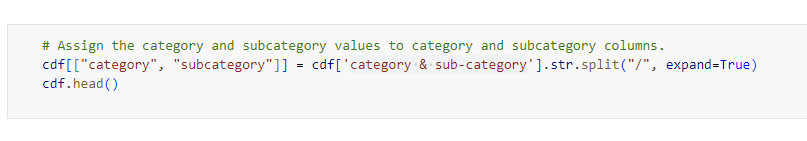


Table Creation Scripting, “QUICKDBD/ QuickDBD-Project 2.sql”

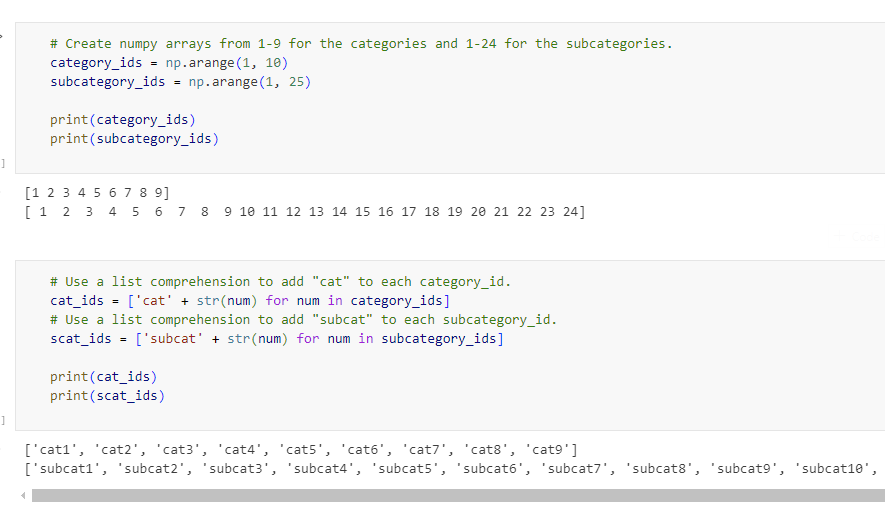


The next step in our Transformation phase was data cleaning and separation. To do this, we used Jupiter Notebook. All of our transformation scripting was done in “ELT\_Mini\_Project\_Starter\_Code.ipynb”. We used this notebook to transform the raw data into four CSV (comma-separated values) files that we would later use to load data into our Postgres tables.  
 For the category and subcategory tables, we split the “category & sub-category“ column into two columns, created an ID column, created DataFrames for each table, and exported the DataFrames as CSVs.

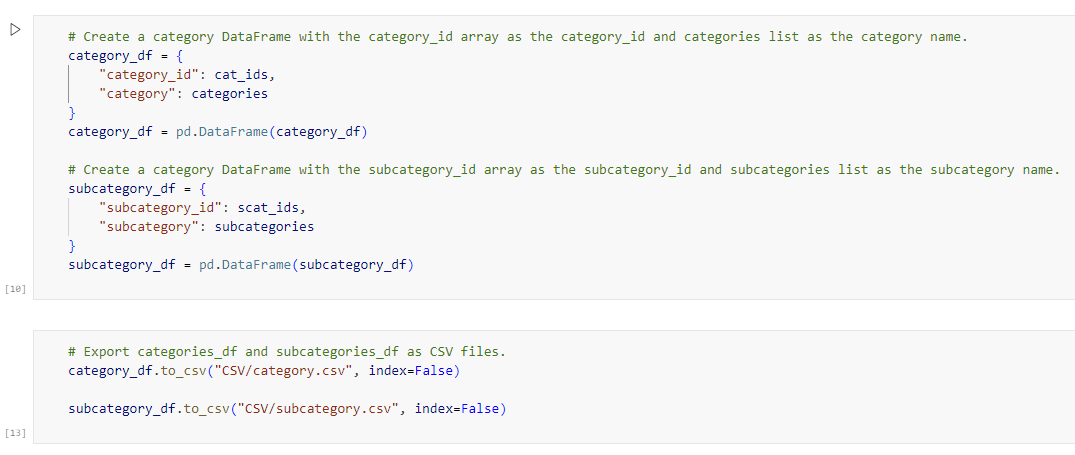
Split Columns:



Created an ID column:



Created DataFrames for each table and saved them as CSVs:

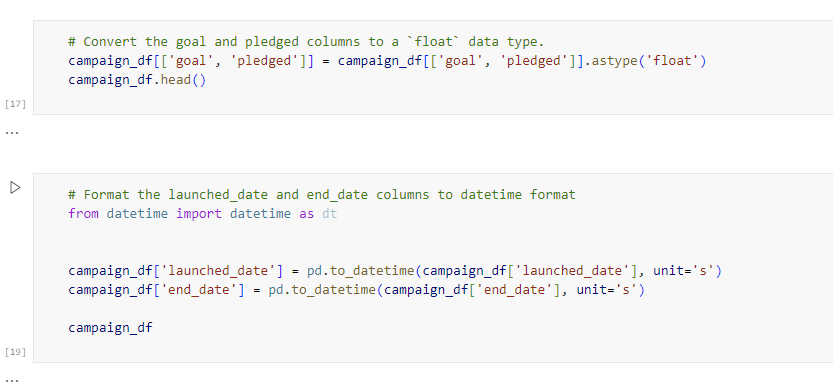


For the campaign table, we renamed columns to more readable titles, changed datatypes, merged with the category and subcategory tables to bring in category\_id and subcategory\_id, dropped unwanted columns, and exported to a CSV.

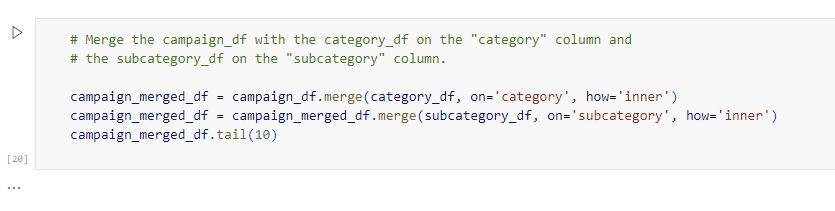
Renaming columns:



Changed Datatypes:



Merged category and subcategory dataframes into campaign dataframe:

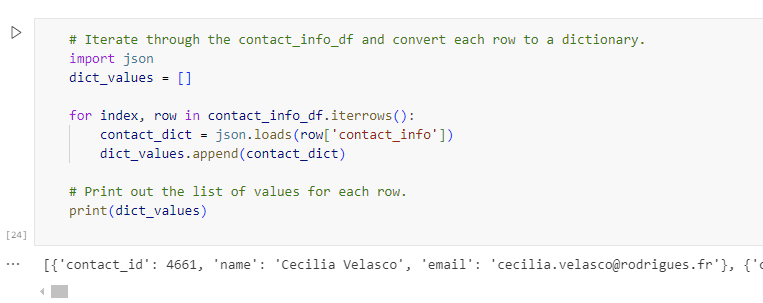


Dropped unwanted columns and Exported to a CSV:

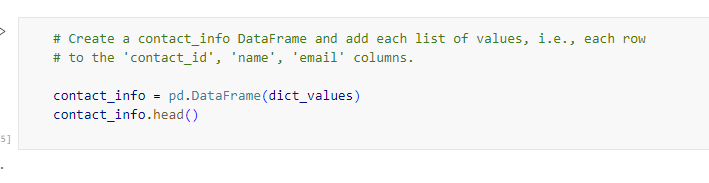


For the contacts table, appended each row to a dictionary, created a DataFrame, split the name column into first and last name, reordered columns, an exported to a CSV.

Appended rows to a dictionary:



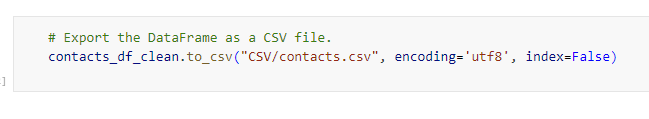
Created a DataFrame:



Manipulated columns:



Exported to a CSV:

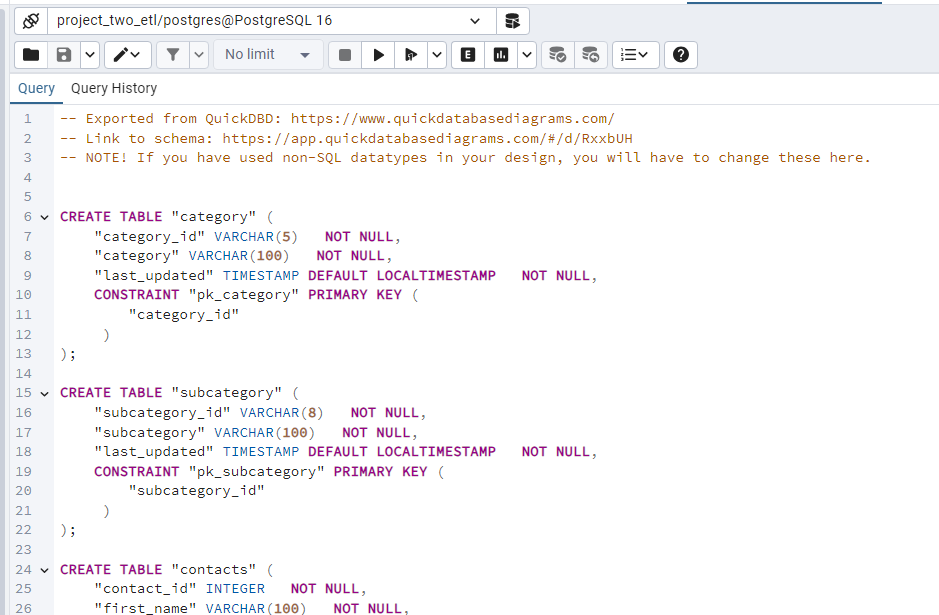


Created CSVs stored in CSV folder: “CSV/category.csv” “CSV/subcategory.csv” “CSV/campaign.csv” and “CSV/contatcts.csv”

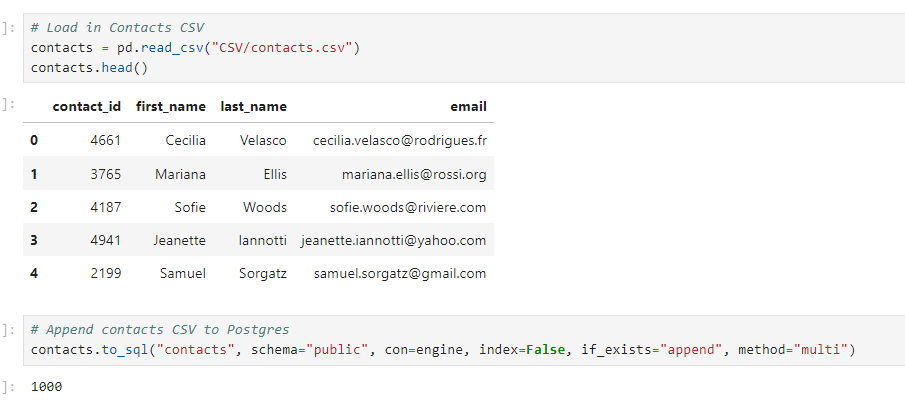
**Load**

The final phase of the ETL process was the Load phase. The first step in our Load phase was to create a database, labelled project\_two\_etl, for the project within Postgres and use the script created by QUICKDBD to create our four tables within that database. After the database and tables were generated, we created a second Jupyter Notebook, “ETL\_Load CSVs.ipynb”, to house all of the script necessary to upload the CSVs to the Postgres tables.

Creating tables within Postgres



Reading CSV and Appending CSV to SQL database



**Data Analysis**

As a ship cannot steer without a rudder, companies cannot function effectively without data analytics. Our ETL pipeline supplied us with information about crowdfunding projects. Stopping there would be a disservice to the information we worked so hard to gather. Our group decided to review this information based on a West vs. East approach, using the United States (US) and Canada (CA) as the West and identifying the East as China (CH) and Australia (AU).

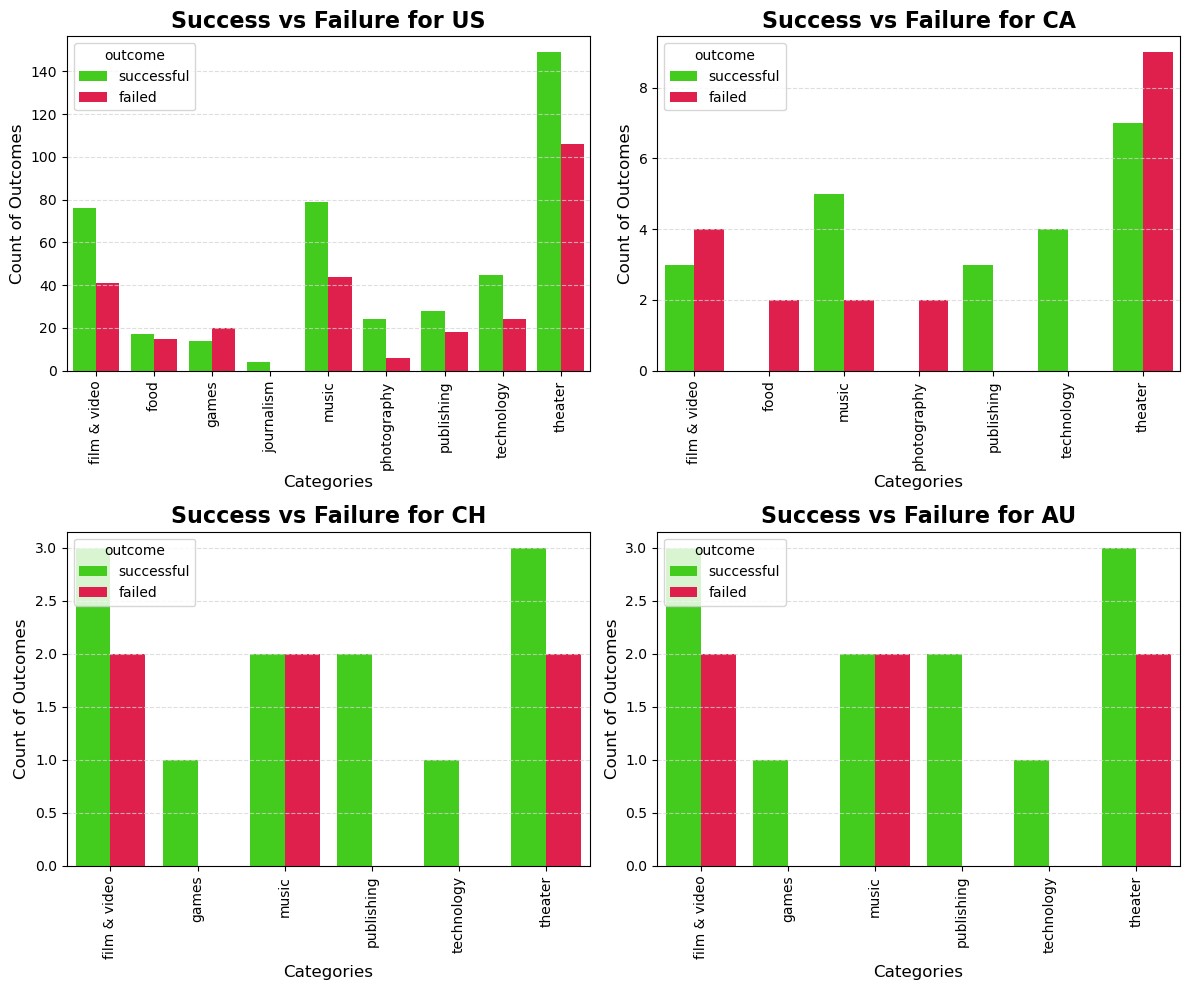
We separated our analysis into three questions:

1. When grouped by category, how does the overall count of successes vs. failures differ for each of our four selected countries?
2. How do our selected countries compare when looking at set goals, pledged finances, and number of backers?
3. How does the length of a campaign compare to the success/failure of our selected countries?

To connect our Jupyter Notebook to our Postgres database. We set up an engine to communicate with our database and then inspected the engine to ensure communication.



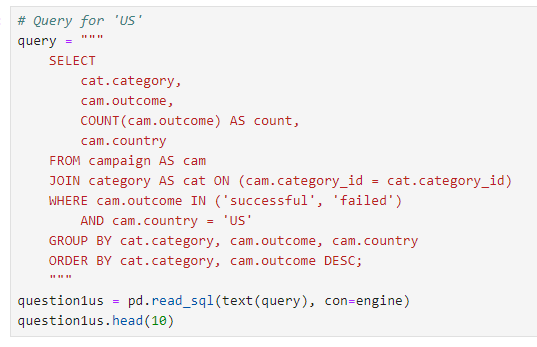
**Question #1 - Four countries, grouped by category, success vs. failure comparison**



This dataset had a strong representation of US crowdfunding projects and a smaller number of overall projects for each of our other selected countries. We are able to see some interesting trends however. Three categories, film & video, music, and theater, had a strong representation in all four countries, showing a popularity for the arts in crowdfunding projects seeking support. It was also interesting to see four categories in China had more failures than successes, a trend that wasn’t reflected in the other three countries, leading us to believe projects in the west had a greater chance of success than in the east.

Individual query and charting work done in “Ind Nick Work/ETL\_Data\_Viz”. All required charting saved in “Charts” folder.

SQL query used to retrieve data from our database, changing WHERE clause for each country:



Python code to generate four charts as subplots:



**Question #2 - Four countries, comparing set goals, pledged finances, and number of backers**