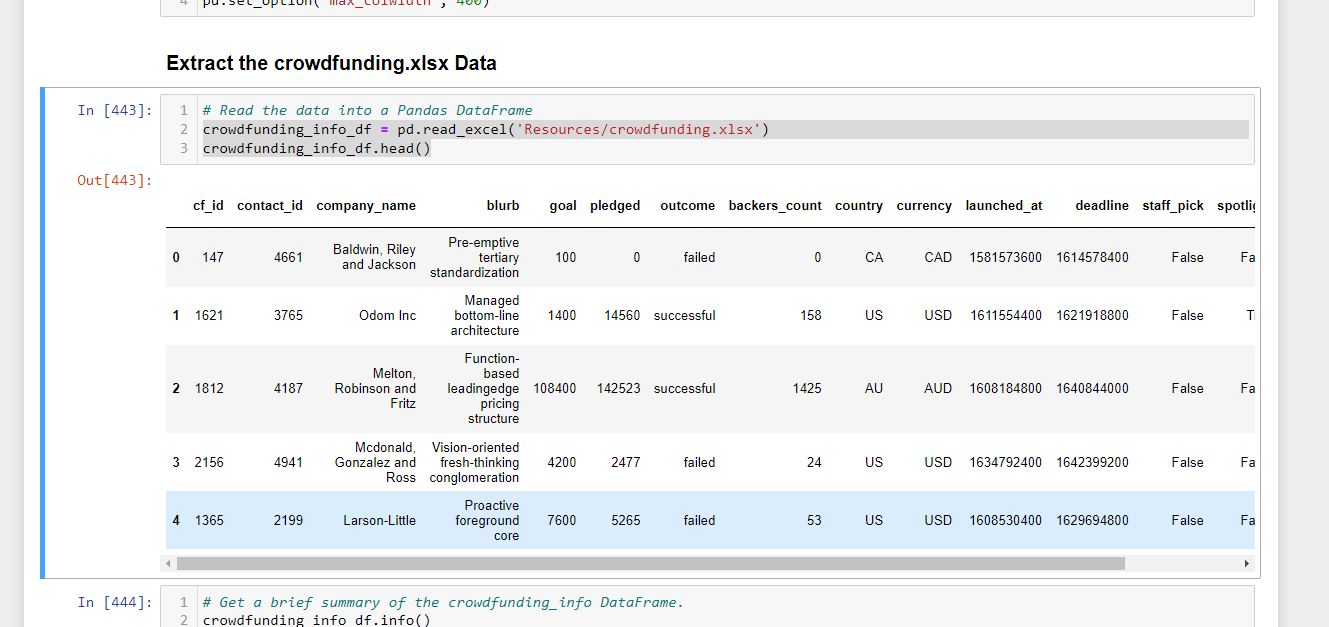
For the ETL mini project, we have worked with a partner to practice building an ETL pipeline using Python, Pandas, and either Python dictionary methods or regular expressions to extract and transform the data. After you transform the data, we have created four CSV files and used the CSV file data to create an ERD and a table schema. Finally, we have uploaded the CSV file data into a Postgres database.

Setting Up github:

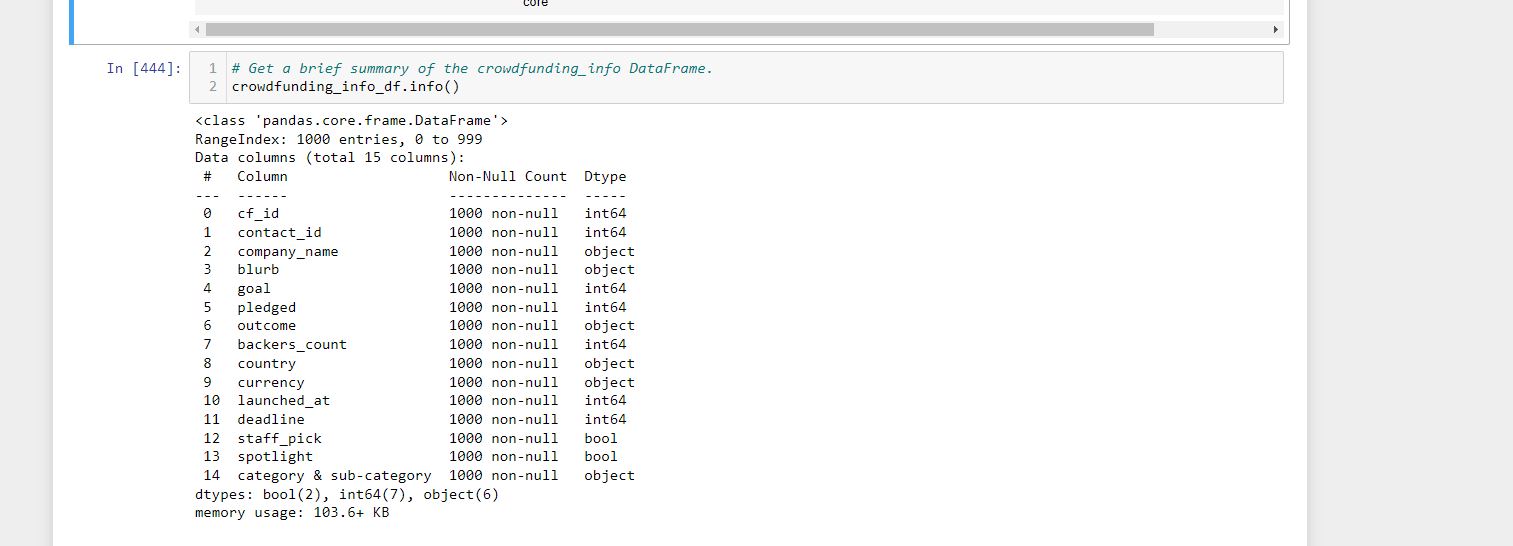
1. Created and renamed the ETL\_Mini\_Project\_starter\_code.ipynb as our names initial such as (MI and El) Mohammad Islam and Erjola Luka.
2. Cloned the repository to our local desktop
3. Added   this Jupyter notebook file and the Resources folder containing the crowdfunding.xlsx and the contacts.xlsx files to our repository.

Crete the Category and Subcategory Dataframes:

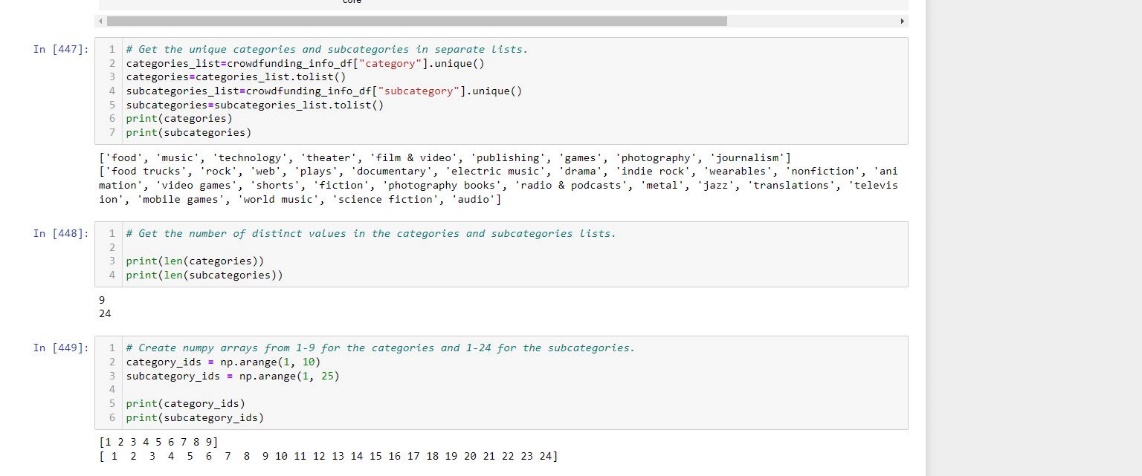
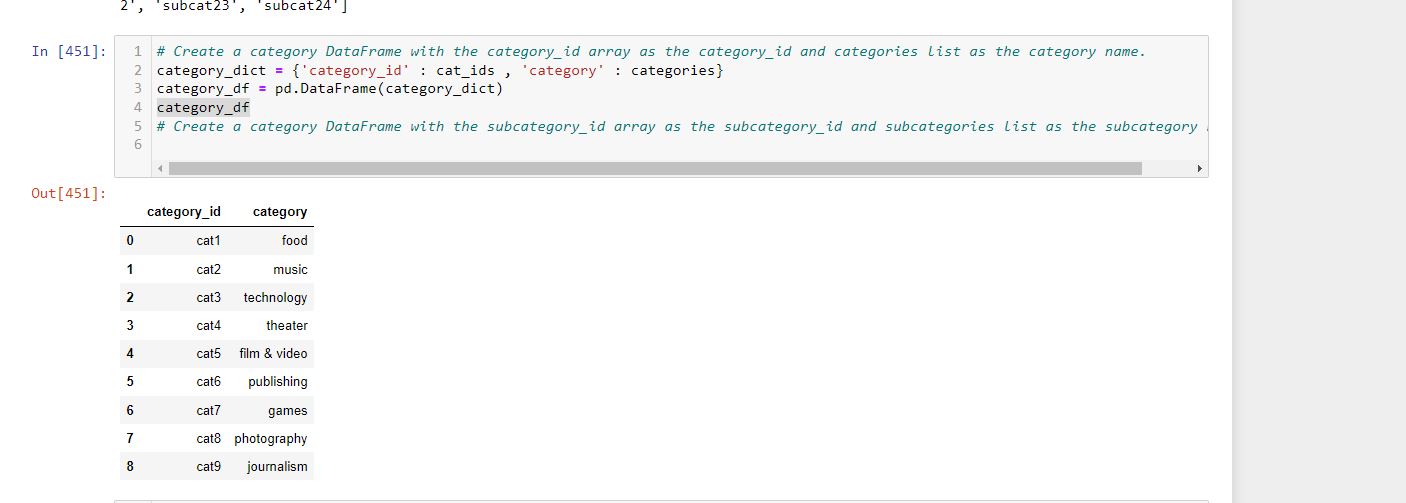
1. Read the excel file present in resources folder and created a df in pandas.



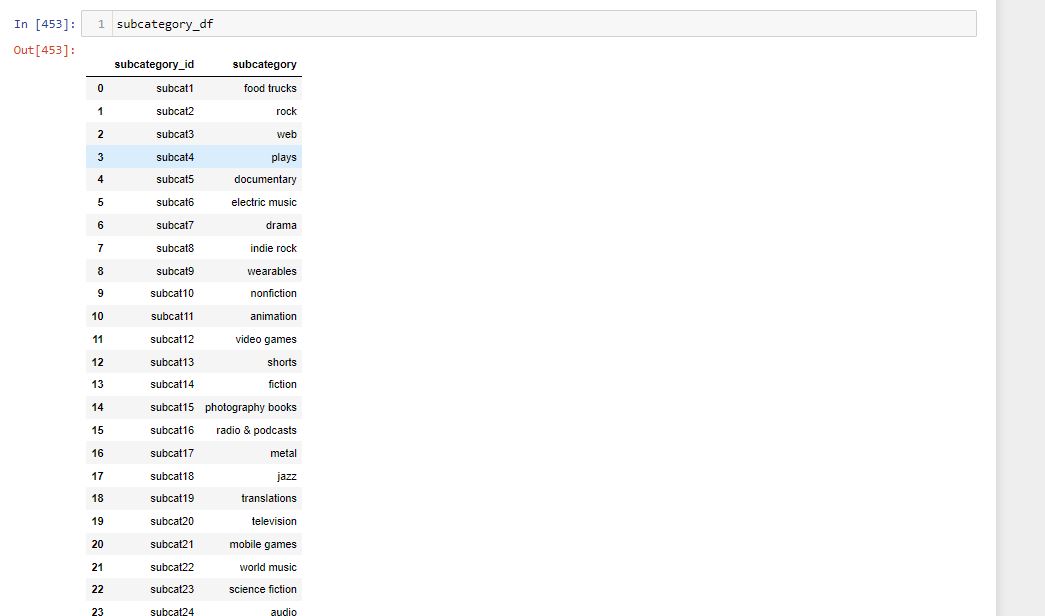
1. Read the datatypes of the data frame created



1. A "category\_id" column that has entries going sequentially from "cat1" to "catn", where n is the number of unique categories
2. A "category" column that contains only the category titles
3. The following image shows this category DataFrame



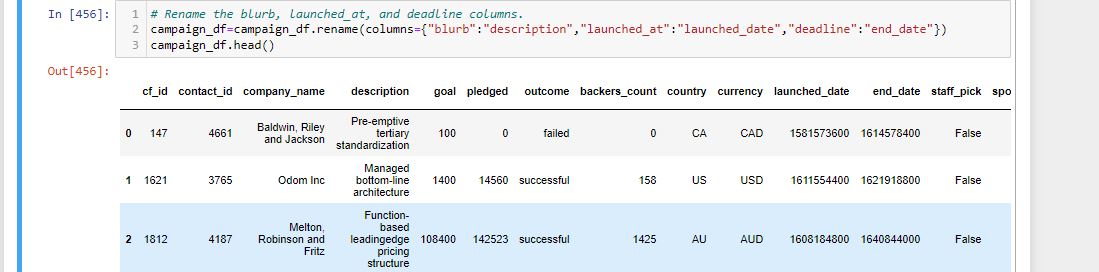
1. Exported the category DataFrame as category.csv and save it to your GitHub repository.
2. A "subcategory\_id" column that has entries going sequentially from "subcat1" to "subcat*n*", where *n* is the number of unique subcategories
3. A "subcategory" column that contains only the subcategory titles
4. The following image shows this subcategory DataFrame:



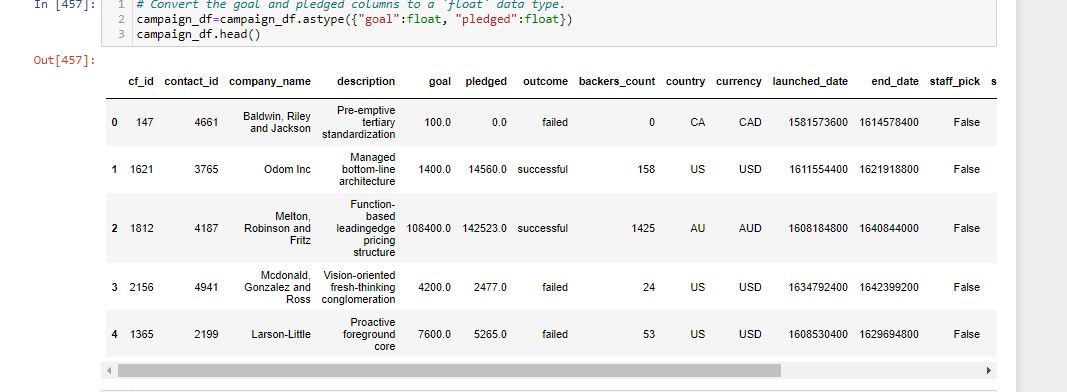
10)Exported the subcategory DataFrame as subcategory.csv and save it to your GitHub repository.

**Campaign DataFrame**

1. Extracted and transformed the crowdfunding.xlsx Excel data to create a campaign DataFrame has the following columns:
   * The "cf\_id" column
   * The "contact\_id" column
   * The "company\_name" column
   * The "blurb" column, renamed to "description"



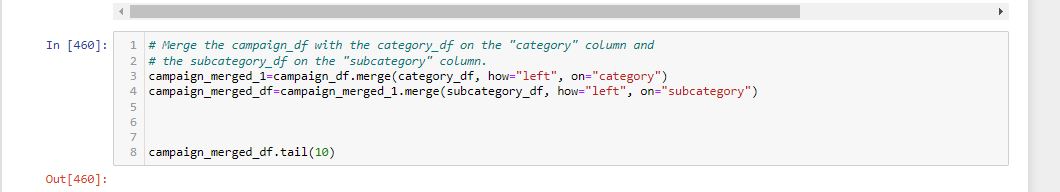
* + The "goal" column, converted to the float data type
  + The "pledged" column, converted to the float data type



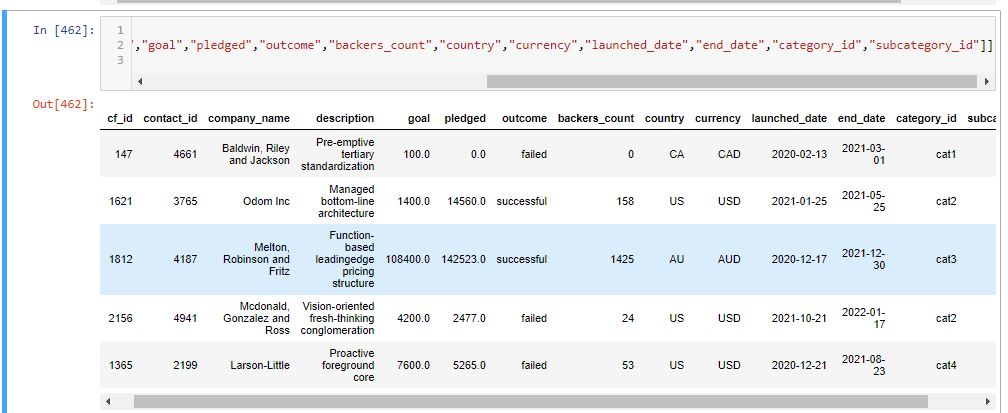
* + The "outcome" column
  + The "backers\_count" column
  + The "country" column
  + The "currency" column
  + The "launched\_at" column, renamed to "launch\_date" and with the UTC times converted to the datetime format
  + The "deadline" column, renamed to "end\_date" and with the UTC times converted to the datetime format



* + The "category\_id" column, with unique identification numbers matching those in the "category\_id" column of the category DataFrame
  + The "subcategory\_id" column, with the unique identification numbers matching those in the "subcategory\_id" column of the subcategory DataFrame



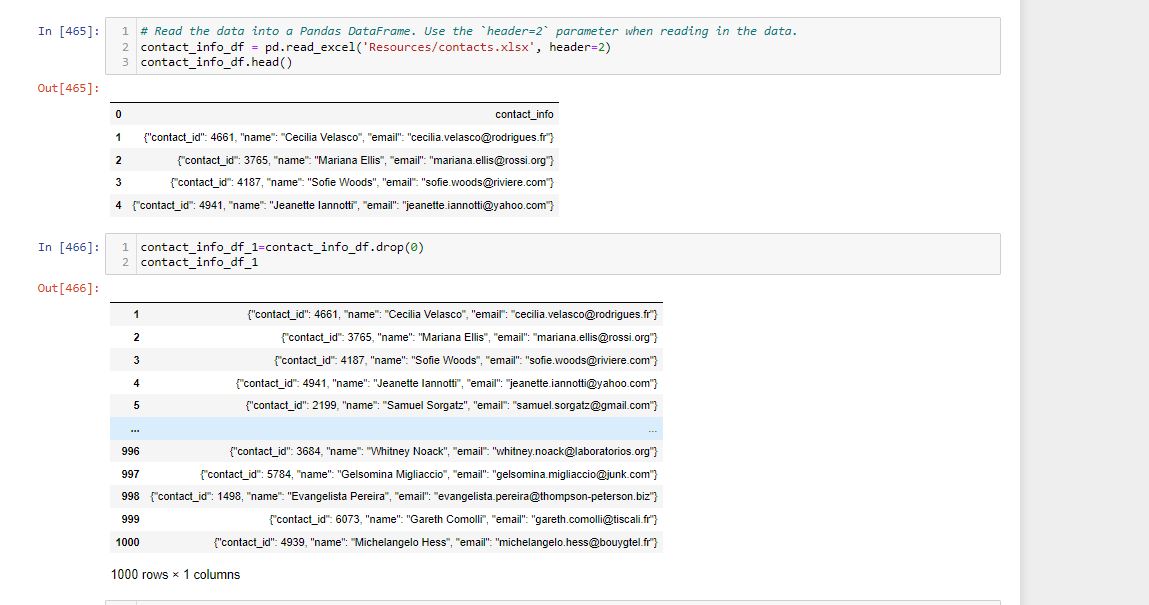
* + The following image shows this campaign DataFrame:



Exported the campaign DataFrame as campaign.csv and save it to our GitHub repository.

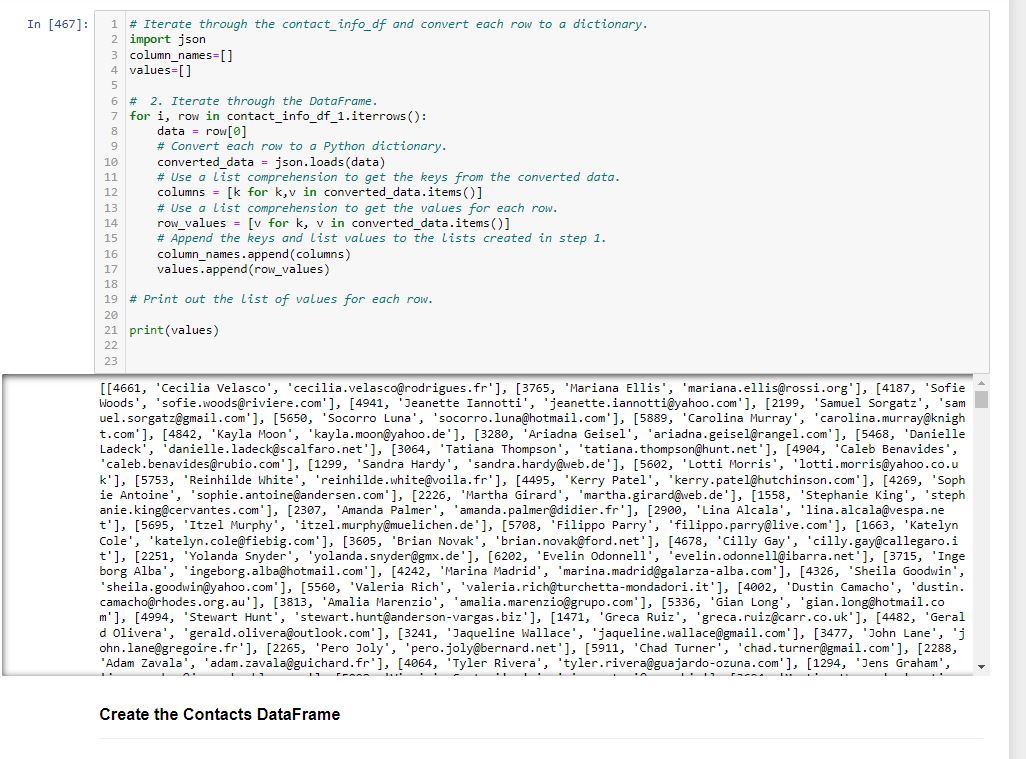
**Create the Contacts DataFrame**

1. Used both two options for extracting and transforming the data from the contacts.xlsx Excel data:



* + **Option 1:**  Python dictionary methods.
  + **Option 2:** Regular expressions.

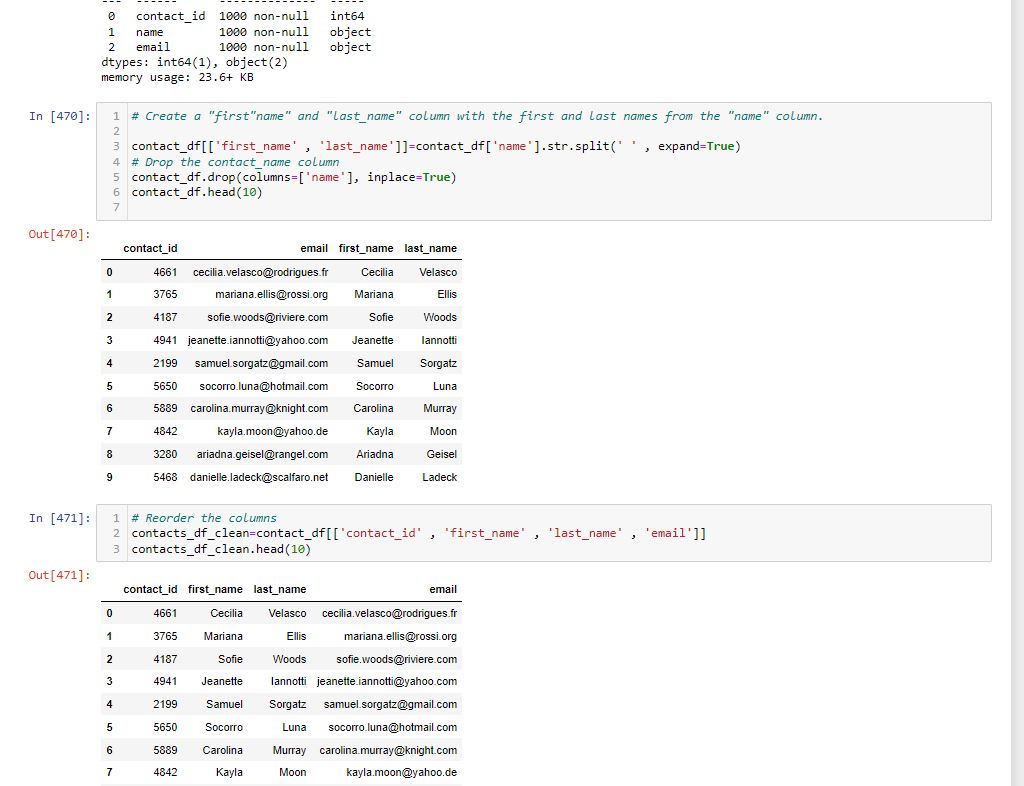
1. For Option 1, completed the following steps:
   * Imported the contacts.xlsx file into a DataFrame.
   * Iterated through the DataFrame, converting each row to a dictionary.



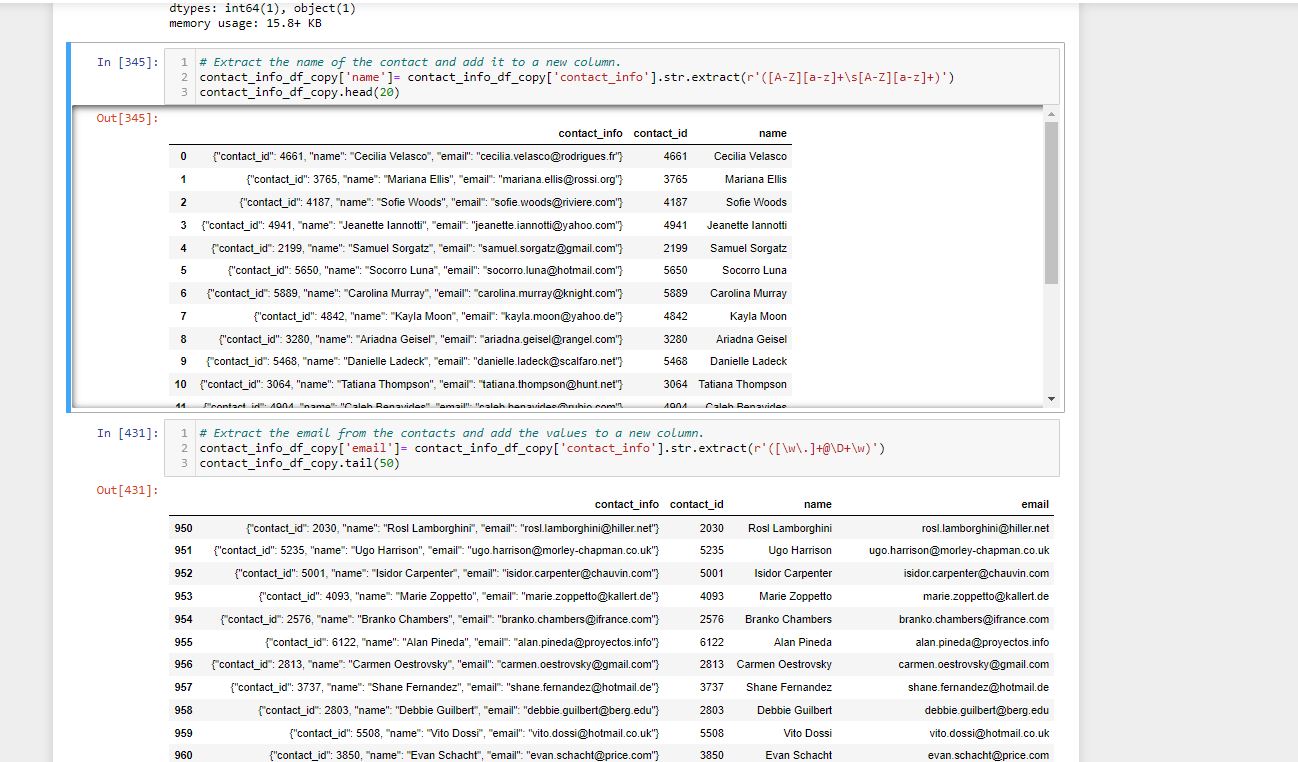
* + Iterated through each dictionary, doing the following:
    - Extracted the dictionary values from the keys by using a Python list comprehension.
    - Added the values for each row to a new list.
  + Created a new DataFrame that contains the extracted data.



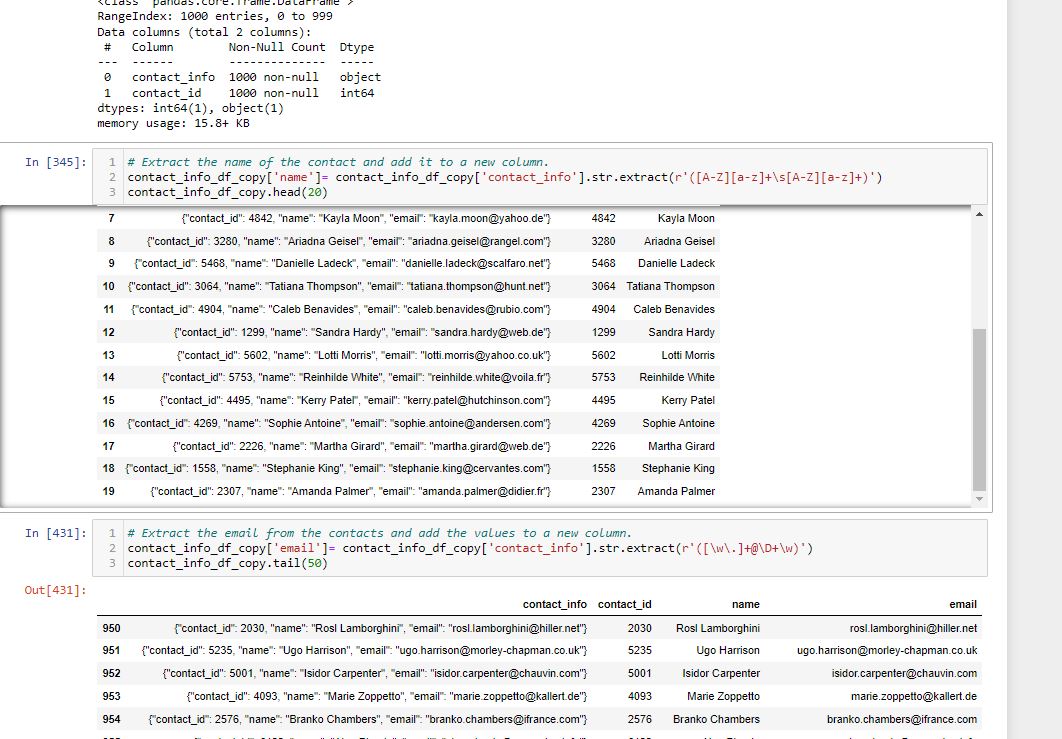
* + Split each "name" column value into a first and last name, and place each in a new column.
  + Cleaned and export the DataFrame as contacts.csv and save it to GitHub repository.

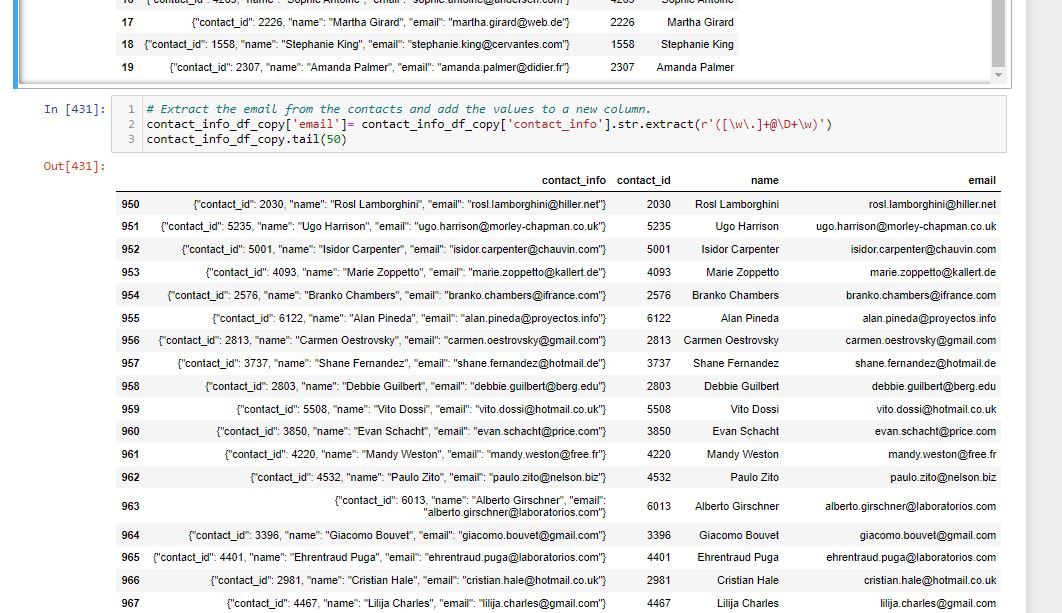


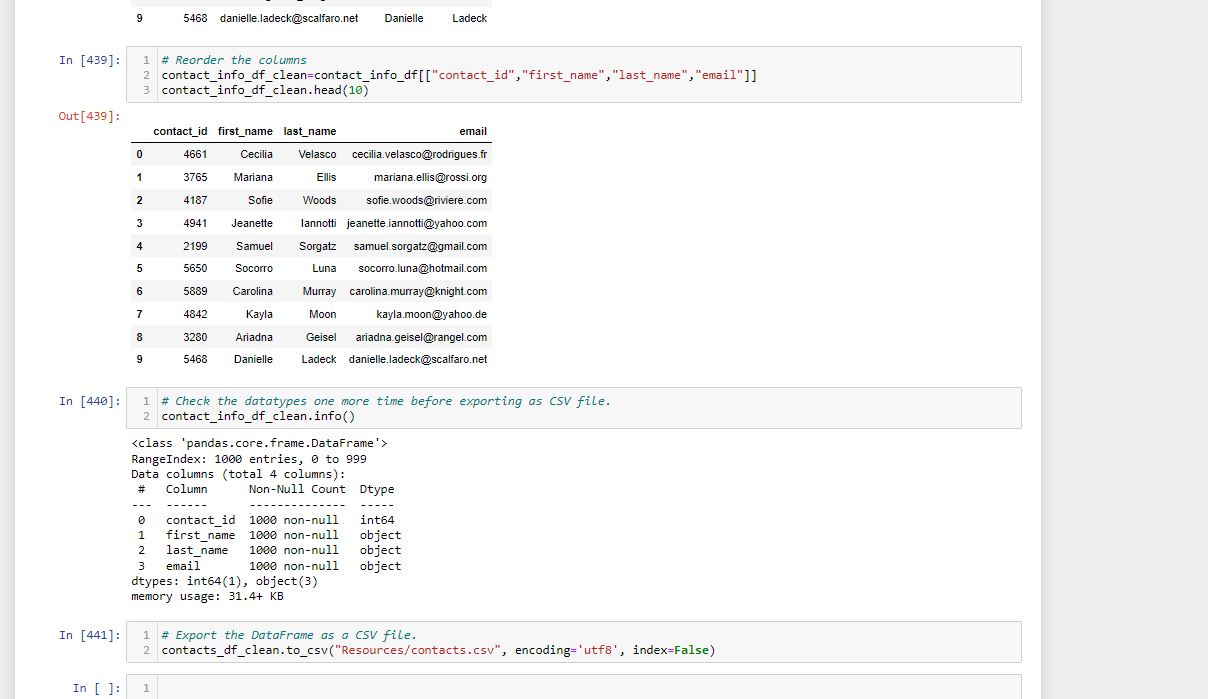
1. For Option 2, complete the following steps:
   * Imported the contacts.xlsx file into a Data Frame.



* + Extracted the "contact\_id", "name", and "email" columns by using regular expressions.
  + 
  + Created a new DataFrame with the extracted data.
  + Converted the "contact\_id" column to the integer type.
  + Split each "name" column value into a first and a last name, and place each in a new column.
  + Cleaned and then export the DataFrame as contacts.csv and save it to GitHub repository.

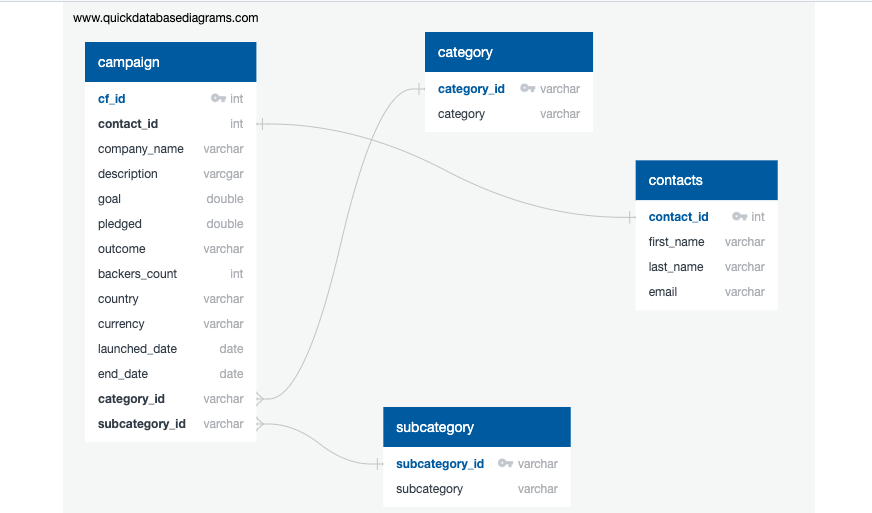






Creating Database:

1. Inspected the four CSV files, and then sketched an ERD of the tables.



1. Used the information from the ERD to create a table schema for each CSV file.
2. Saved the database schema as a Postgres file named crowdfunding\_db\_schema.sql, and save it to GitHub repository.
3. Create a new Postgres database, named crowdfunding\_db.
4. Using the database schema, created the tables in the correct order to handle the foreign keys.
5. Verified the table creation by running a SELECT statement for each table.
6. Imported each CSV file into its corresponding SQL table.
7. Verified that each table has the correct data by running a SELECT statement for each.

