### **Project 2 – Crowdfunding ETL**

Team 1

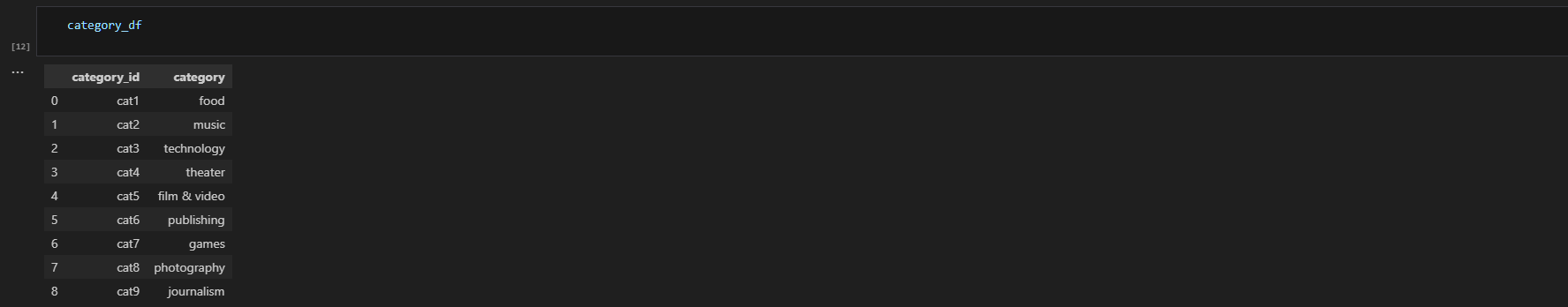
* Faisal Dernawi | [fdernawi11@gmail.com](mailto:fdernawi11@gmail.com)
* William Davis | [william.c.davis@nasa.gov](mailto:william.c.davis@nasa.gov)
* Samuel Kim | [sky4yoon@icloud.com](mailto:sky4yoon@icloud.com)

----

### **Requirements**

#### A Category DataFrame is Created (15 points)

* The DataFrame contains a "category\_id" column that has entries going sequentially from "cat1" to "catn", where n is the number of unique categories (5 points)
* The DataFrame has a "category" column that contains only the category titles (5 points)

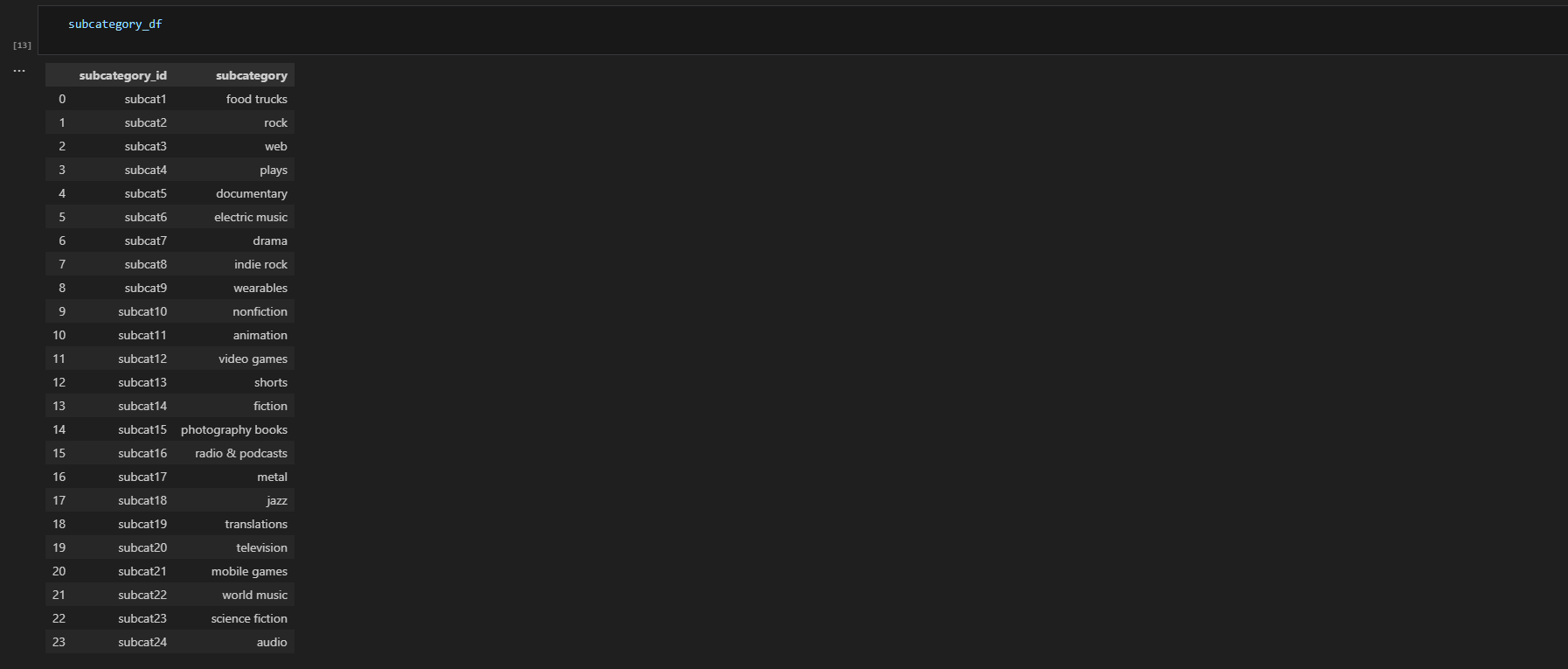


* The category DataFrame is exported as category.csv (5 points)

See file in repo: Resources/category.csv

#### A Subcategory DataFrame is Created (15 points)

* The DataFrame contains a "subcategory\_id" column that has entries going sequentially from "subcat1" to "subcatn", where n is the number of unique subcategories (5 points)
* The DataFrame contains a "subcategory" column that contains only the subcategory titles (5 points)



* The subcategory DataFrame is exported as subcategory.csv (5 points)

See file in repo: Resources/subcategory.csv

#### A Campaign DataFrame is Created (30 points)

* The DataFrame has the following columns: (25 points)
  + A "cf\_id" column
  + A "contact\_id" column
  + A "company\_name" column
  + A "description" column
  + A "goal" column that is a float data type
  + A "pledged" column that is a float data type
  + An "outcome" column
  + A "backers\_count" column
  + A "country" column
  + A "currency" column
  + A "launch\_date" with the UTC times converted to the datetime format
  + An "end\_date" with the UTC times converted to the datetime format
  + A "category\_id" column that contains the unique identification numbers matching those in the "category\_id" column of the category DataFrame
  + A "subcategory\_id" column that contains the unique identification numbers matching those in the "subcategory\_id" column of the subcategory DataFrame

A screenshot of a computer

Description automatically generated

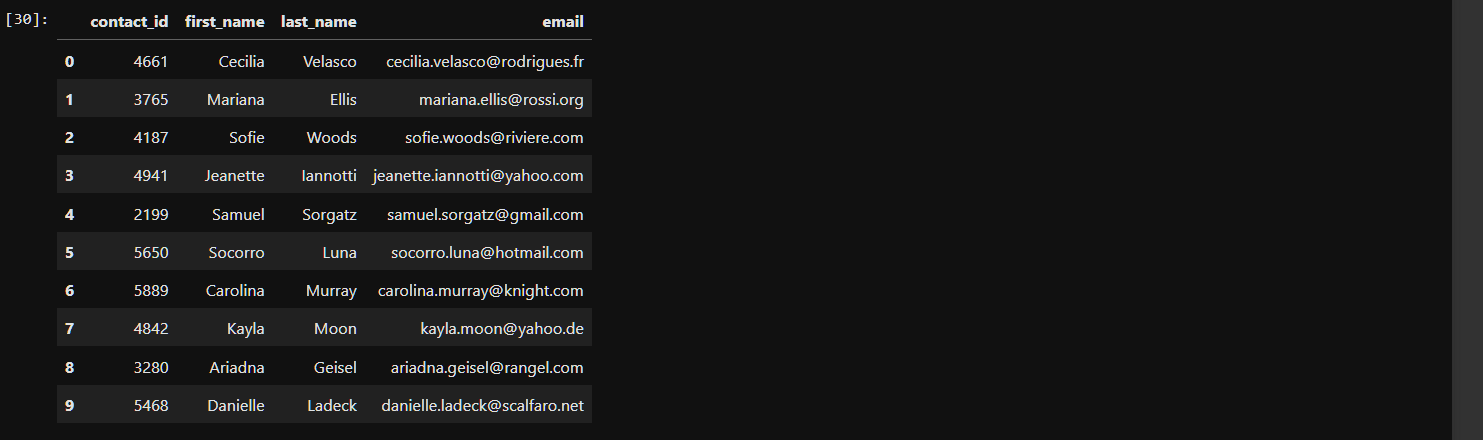
* The campaign DataFrame is exported as campaign.csv (5 points)

See file in repo: Resources/campaign.csv

#### A Contacts DataFrame is Created (15 points)

#### Option 1: Use Python dictionary methods

* The DataFrame has the following columns: (10 points)
  + A "contact\_id" column
  + A "first\_name" column
  + A "last\_name" column
  + An "email" column

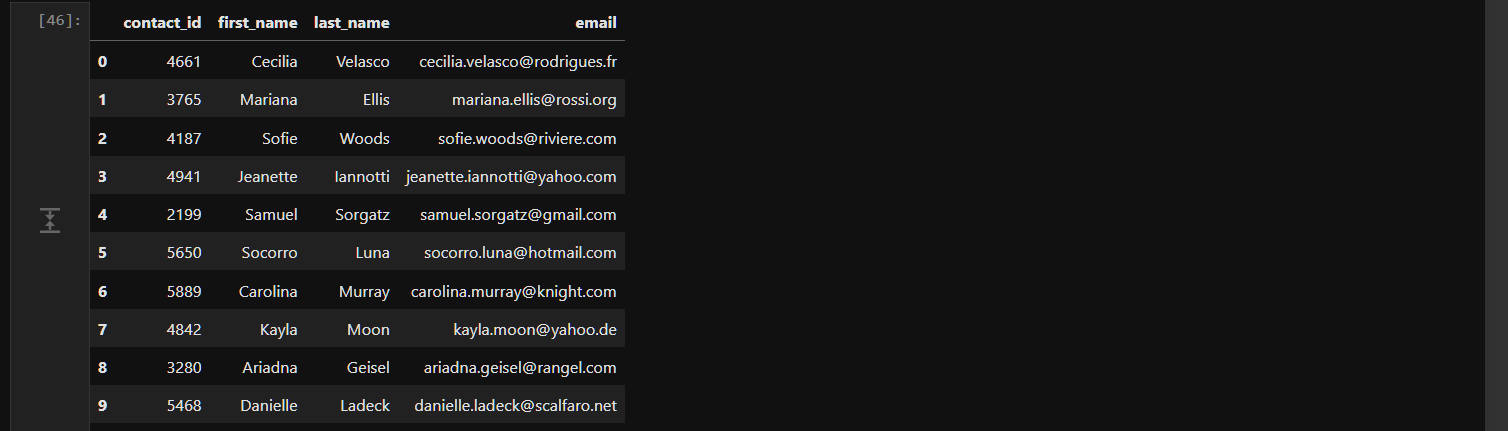


* The contacts DataFrame is exported as contacts.csv (5 points)

See file in repo: Resources/contacts.csv

#### Option 2: Use regular expressions

* The DataFrame has the following columns: (10 points)
  + A "contact\_id" column
  + A "first\_name" column
  + A "last\_name" column
  + An "email" column



* The contacts DataFrame is exported as contacts.csv (5 points)

See file in repo: Resources/campaign.csv

#### A Crowdfunding Database is Created (25 points)

* A database schema labeled, crowdfunding\_db\_schema.sql is created (5 points)

A close-up of a computer screen

Description automatically generated

See file in repo: Resources/ SQL\_Schema(s)/ crowdfunding\_db\_schema.sql

* A crowdfunding\_db is created using the crowdfunding\_db\_schema.sql file (5 points)

A computer screen with a white background

Description automatically generated with medium confidence

See file in repo: Resources/ crowdfunding\_db/ crowdfunding\_db.sql

* The database has the appropriate primary and foreign keys and relationships (5 points)

A screenshot of a computer

Description automatically generated

See file in repo: Resources/ SQL\_Schema(s)/ crowdfunding\_db\_schema.sql

* Each CSV file is imported into the appropriate table without errors (5 points)

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

* The data from each table is displayed using a SELECT \* statement (5 points)

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated