**ETL Project Report: “Far Out”**

**Team: Group#1 |** Elizabeth Jaye | Dinielle Nelson | Dolly Vickers

**Datasets**:

We had interest in datasets relating to NASA. We found a CSV listing of all US astronauts through Kaggle [1]. Wanting variety in the datasets, we also looked for tables on webpages or for available APIs. We found two more related tables on webpages, which had different formatting. One table was from Wikipedia [2] which is well known, and one was from windows2universe.org from the National Earth Science Teacher Association [3] which we found credible.

1. <https://www.kaggle.com/nasa/astronaut-yearbook>
2. <https://www.windows2universe.org/space_missions/manned_table.html>
3. <https://en.wikipedia.org/wiki/Comparison_of_crewed_space_vehicles>

**Choosing relational or non-relational databases:**

Looking at the datasets of Astronauts, Missions and Spacecraft, we drew out the relations to see if it would more useful to build a relational or non-relational database.

1. Astronauts: Each crew/astronaut has multiple missions and multiple spacecraft associated with them.
2. Missions: Each mission has multiple astronauts but only one Spacecraft
3. Spacecraft: Each spacecraft has multiple astronauts and multiple missions

Using this acknowledgement, we decided that a relational database would be best at describing this complex relationship. We knew we’d have to use SQLite to build the tables in a remote way that could be uploaded to a repository.

**Data Cleanup:**

1. Astronauts:
   1. Limit the dataset to anything older than 1991 since that was the max date of the Mission dataset
   2. Change Years and Group numbers from floats to integers
   3. Modify the Name field since the Mission dataset only identify crew by last name
      1. Remove “Jr.” from the last name
      2. Split first and middle names from last name and make separate column
      3. Rename Name to Full Name
   4. Reorder columns to read for easily
   5. Remove astronauts that do not have mission
   6. Add unique ID to each astronaut
   7. Remove the field that has multiple relationships, that being Missions
2. Missions:
   1. Move header to column names and remove first row containing header names
   2. Reset index
   3. Remove rows that contain “USSR” in the mission name to remove Russian missions
   4. Delete country label “(US)” from Mission name
   5. Rename Craft to Given Craft Name to better reflect that this is a name given by crew, and not the model name of the ship
   6. Add column as result of splitting Mission name which is the Craft name to be joined later
   7. Clean up crew columns from formatting issues
   8. Modify names to match those found in the Astronauts table
   9. Save dataset as a second dataframe so that one can be used for the future Join table
   10. Remove the field that has multiple relationships, that being Crew
3. Spacecraft:
   1. Move header to column names and remove first row containing header names for both tables
   2. Add orbital type to differentiate the two tables when merged
   3. Merge two tables
   4. Extract US spacecraft only
   5. Rename columns
   6. Reset index
   7. Remove notes in () and []

**Creating the relational database:**

The relational database consists of three tables with unique IDs and one table with the relations mapped out.

1. Crew
   1. Add unique ID
2. Missions
   1. Add unique ID
3. Spacecraft:
   1. Add unique ID
4. Join table (Astronauts and Missions):
   1. Based on the original Missions table, separate the missions from each crew and creating new row per crew for each mission
   2. Rename Last Name field to field name of Join table, that being Crew
   3. Extract crew ID and crew name from Astronaut table
   4. Merge Join table and extracted Astronaut table based on last name of astronauts
5. Join table (Mission/Joined and Spacecraft):
   1. Upon joining Spacecraft and Craft of the Spacecraft and Mission/Join tables, respectively, no data was returned. After reviewing data types and whitespace complications we discovered that our Spacecraft table listed Manufacturer and Model names but not the actual individual space craft. Even more complicated, mission names often took on the same names as the spacecraft models, using numbers to establish one mission form another.
   2. This caused us to split Mission name (as in step 2f of cleanup) to get a craft name to join upon
   3. Extract spacecraft ID and spacecraft name from Spacecraft table
   4. Merge Join table and extracted Spacecraft table based on craft name

**Output:**

SQLite file, NASA.db containing three tables, ASTRONAUTS, MISSIONS, SPACESHIPS and MASTER.