### **ETL Project: Thao Leatherman, Amelia Aberumen, Lyana Roslan**

### **Objective:** We wanted to determine which US domestic airlines has the most delays & cancellations, what were the major causes. Does holidays play a part in delays & cancellations? (E.g Thanksgiving, Christmas, etc.)

### **Data sources:** KaggleWorld &Bureau of Transportation Statistics

### Timeframe: From Jan 2016 to Dec 2016

### 2016 US flights from Kaggle World

* 2016 Bureau of Transportation Statistic data

### **Extraction:**

* Our data came in a raw CSV data format and our original data sources came from Kaggle World and Bureau of Transportation Statistics. Both of our data is limited to the year 2016 but includes all major airports in the country. The first Kaggle World data includes month, airport, arrival delay, arrival time, etc.
* We took a raw CSV data file and we used Pandas to place the csv into a dataframe and read the data from pandas. We also wanted to implement dataframes in Python for easier data processing. We then wanted to display only the statistical overview of the dataframe by calculating the (average) mean and standard deviation of delays by flight.

### **Transformation:**

* In terms of transformation, we needed to clean the data by ensuring that if there were any missing data by airline or by airport, we would take it out. We also dropped rows and columns where at least one element is missing by using ‘dropna’ query in the data frame.
* We approached the project with multistage data transformation where extracted data is moved to a staging area where transformations occur prior to loading the data into the warehouse.
* We also tried to join the data but failed -- we wanted to try because once we joined the 2 data sources, we are able to drop duplicated data/missing data (same airline, same airports, same dates) together. We then just transformed the data separately. We would also would’ve loved to index the data and see “everything”.
* We also planned to join the data because we wanted to clean the data, rename the columns, etc.

### **Loading:**

* We chose to load the final database into SQLdatabase and we first did that by establishing a connection to ensure interaction with the database. SQLAlchemy provides a nice “Pythonic” way of interacting with databases and can be used to automatically load tables from a database (reflection).
* We created flight\_db database and we added a table that included carrier name -- within the table we identified primary key as ARR (arranged flight). We also separated the carrier and airport name.
* The second table is carrier delay & we also used the same primary key. We picked the ARR because the ID number is different for each airline/carrier. There was no duplicated number. In that table, we separated weather delay from the carrier delay to compare.
* We then joined the first table (carrier name) with the (carrier delay) to compare & answer our question.
* At the end, we just chose to go with weather delay because they had the most cancelled flights, about 321 delays total in the year.
* We chose to choose SQL because relational databases store data sets as “relations”. We were able to compare tables with rows and columns where all information is stored to differentiate value. If we use html, we would have to use Mongo.