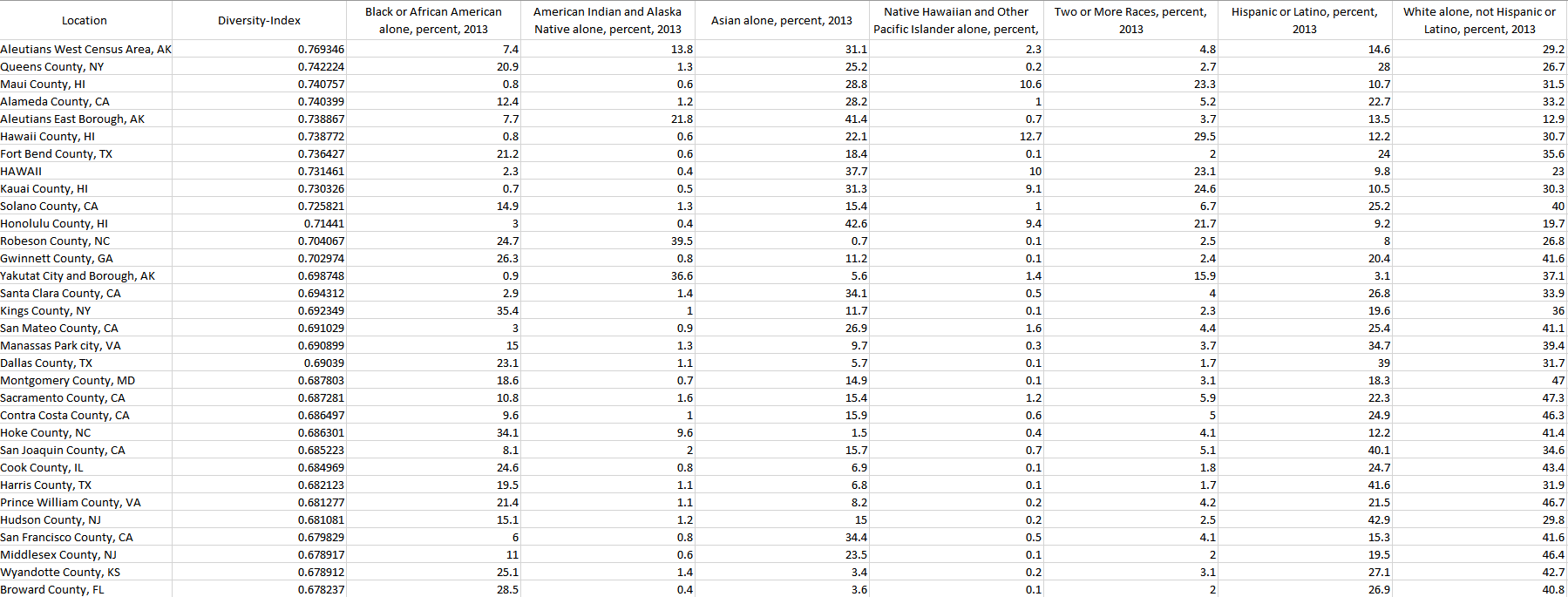
**Extract**

We have utilized two public data sets from Kaggle. Gathered data is based on county through all the States ranging over various years. Also, the available data has been provided in a CSV format which will require cleaning and reformatting which we have done during the transformation process in Python.

The data sources are as following

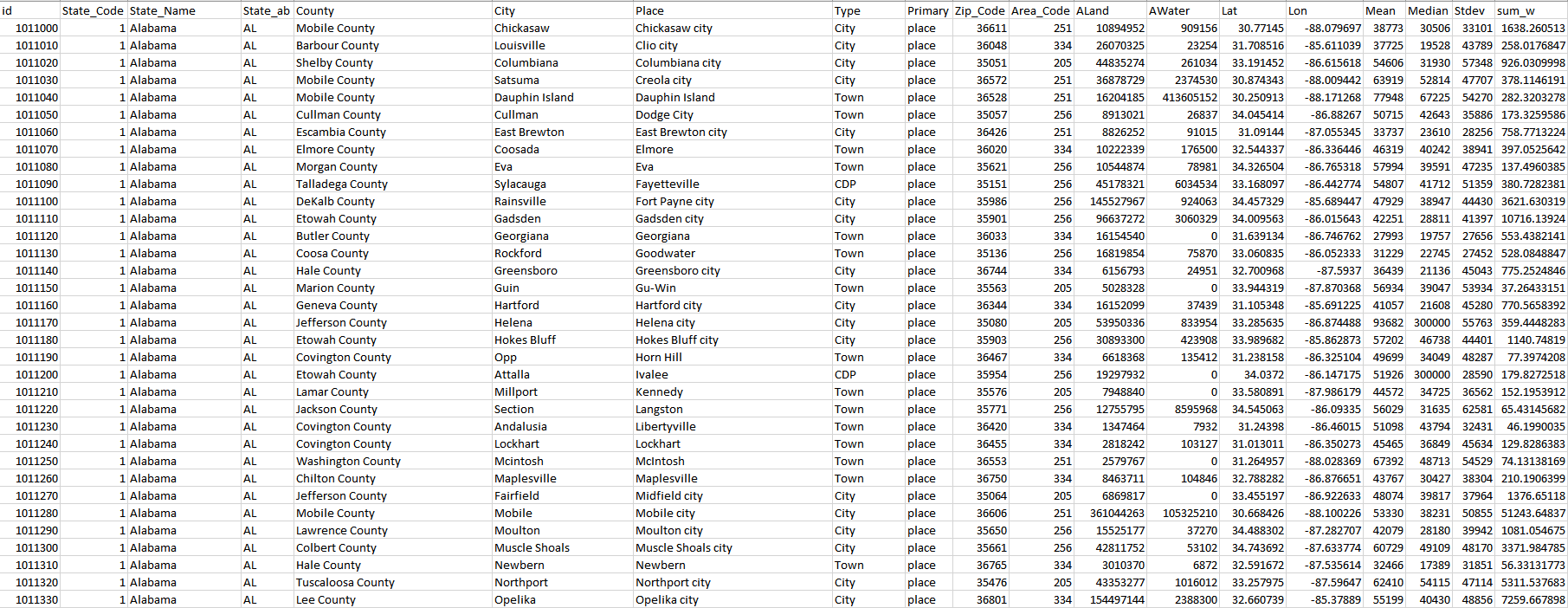
* Diversity Index from Kaggle

<https://www.kaggle.com/mikejohnsonjr/us-counties-diversity-index>



* Median Income by county Kaggle

<https://www.kaggle.com/mikejohnsonjr/us-counties-diversity-index>

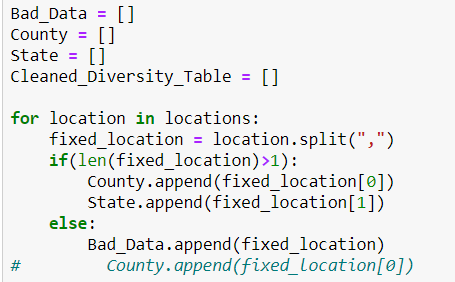


**Transformation**

1. We imported both files to python with the right encoding to ensure data is being transferred.

**Encoding**

1. The second step towards our transformation process is to clean up the datasets and indicate which variables were not relevant to analyze of both data sets. For Diversity Index Dataset, we dropped the irrelevant columns and removed any spaces in our “State” cells.
2. Clean the **Diversity Index** dataset to separate the county and state as PK.

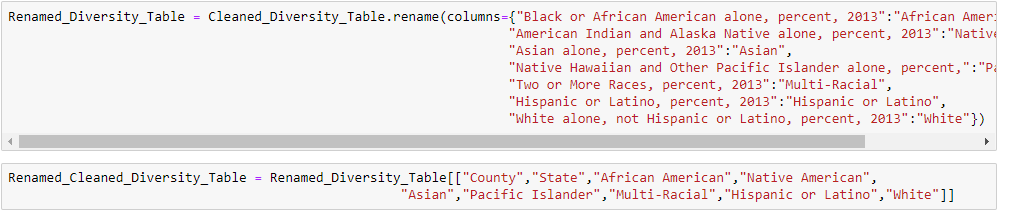


* **Dropped the non-relevant columns**

During this step, we removed any columns of data that can be completely irrelevant to our analysis.

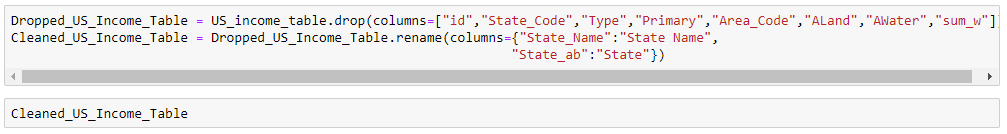


* **Rename the columns to a more compatible name to PgAdmin**



1. Clean the **Median Income** dataset to separate the county and state as PK

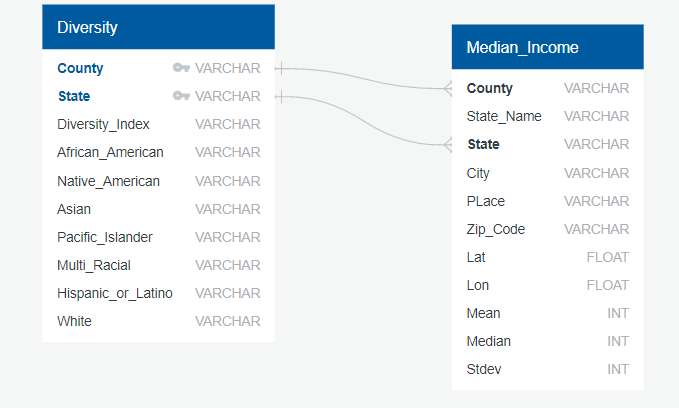
* **Dropped the non-relevant columns**



* **We did not need much renaming for this data set since it was much cleaner, and the column names were pretty straight forward**

1. Defining our Schema based on the available data from both data sets

* Defined County and State as our Primary keys.



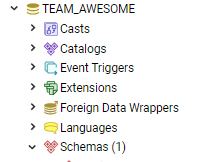
1. The final step of our transformation process will be to merge both data set on their primary keys which will be “County” and “State” in order to get the average income by each diversity category.



**Load**

The last step is to define our database on PgAdmin which we have as “TEAM\_AWESOME” and transfer our final output into our defined database. We created a database and respective tables to match the columns from the final Panda’s Data Frame using pgAdmin and then connected to the database using SQL-Alchemy and loaded the result.

* **Define Database**



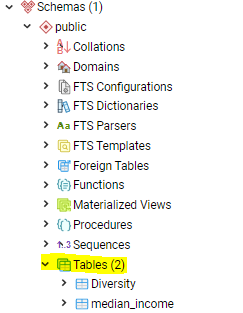
* **Loading Diversity Table**



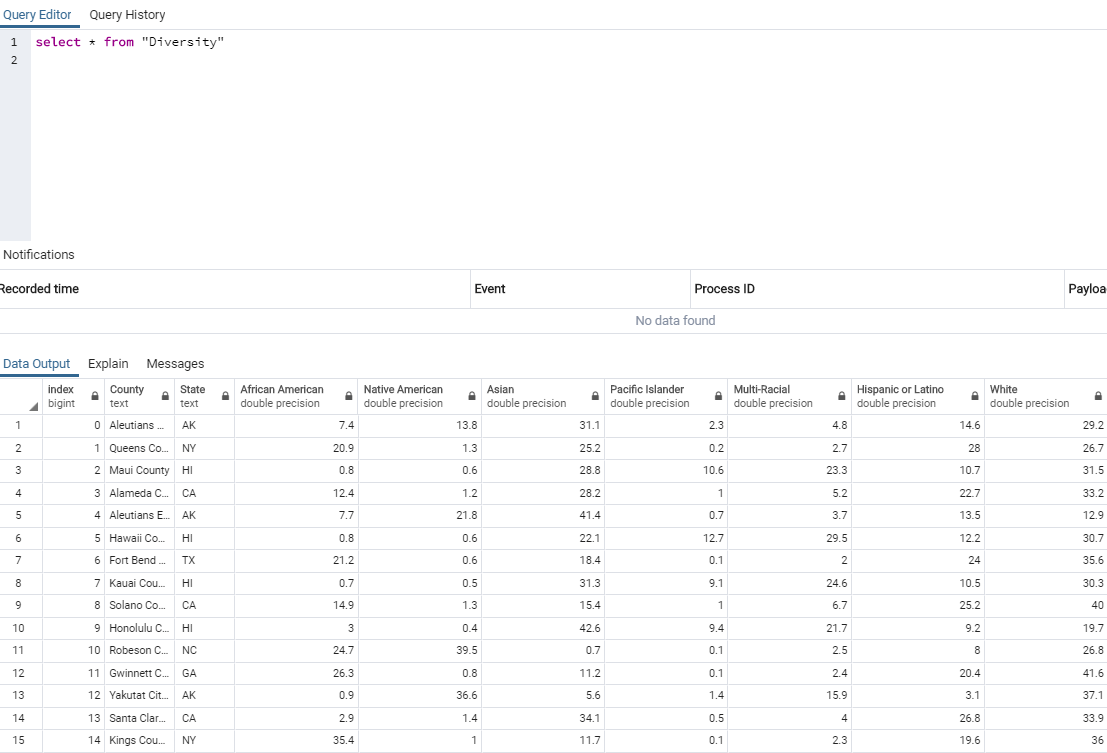
* **Loading Median Income Table**



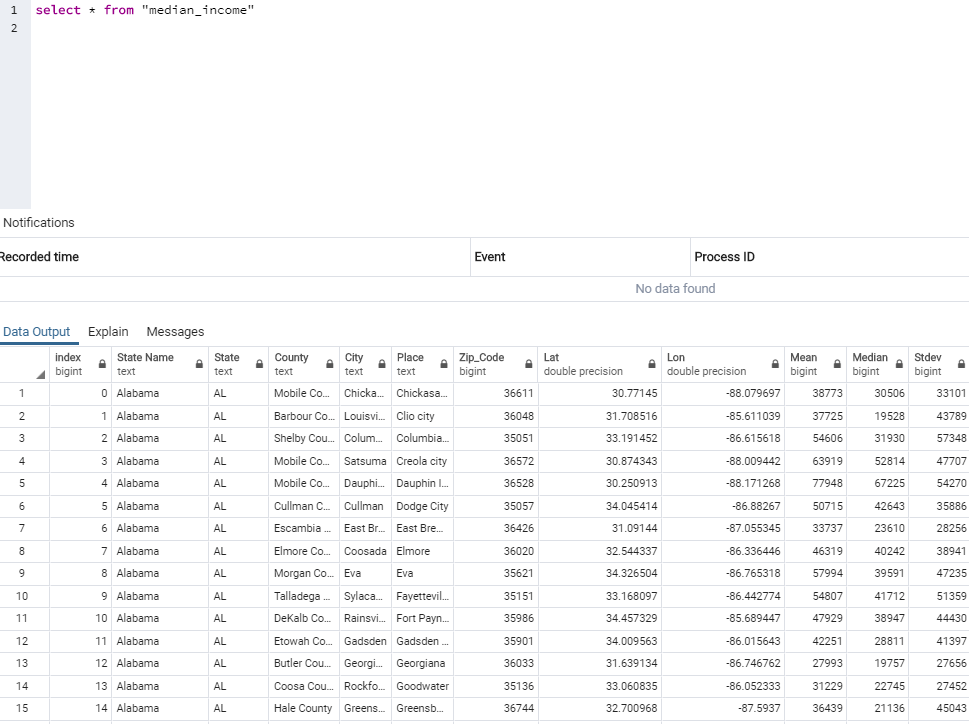
* **Tables have been created based on our Schema**



**Sample Query to view all the data in Diversity Index table**



**Sample Query to view all the data in Median Income Index table**



**Summary**

We used these datasets to identify the diversity ratio and median income per county for each state. The final output will help us to recognize which county, state that has the following.

* Median Household Income by diversity
* Diversity Index
* Race Index

These indices can be used to identify if any development aid/work are required for areas with high/low median income or with high population.

If the specific ratio population is high and if the schools/colleges/educational institutions are less, these indices can be used to build more educational institutions, which in turn create more employment opportunities. It is a cascading effect of more the population, more educational institutions, more employment, more household income.