The two data sources used were importing a flat file and web scraping:

a)For the flat file I downloaded a CSV file from *ourworldindata.org* with the number of international arrivals to 264 countries around the world. There is data from 1995-2016 but some years are missing for some countries.

b)I tried using the world bank API but was not successful. I ended finding the data I needed in a wikipedia page with a table. I used web scraping to grab the data I needed off the html table into a pandas dataframe.

**Extract and Transform:**

Source #1:

I used google to find data on international arrivals. I lead me to an interactive chart with the data I was looking for on *ourworldindata.org*. The chart has a DATA button that allows you to download a CSV file. I opened up the file on excel to view it and see how the data was organized. I then used the *read\_csv* method in pandas to load the CSV file into a pandas dataframe.There is a total of 264 countries in the dataset.

The first thing I did was renaming two of the columns into something more descriptive. Changed ‘Unnamed: 3’ into ‘Arrivals’ and ‘Entity’ into ‘Country’. I used pandas to drop every row that the year was not 2015. I chose 2015 because it was the most recent year that was available in the two datasets. I renamed the column again to ‘Arrivals\_2015’.

Source #2:

I found the data I was looking for on a wikipedia page. The data was in the form of an html chart. The population data is given in thousands, so I would have to adjust that before combining sources. I used the pandas method *read\_html* to grab the html into a pandas variable. I eliminated all the columns I didn't need using the drop method in pandas leaving only the Country and 2015 columns. I used the iLoc method to remove the first to rows because they were not individual countries. Renamed the remaining columns to something more descriptive.I chose to rename one ‘Country’ to match the first dataframe knowing I would want to merge dataframes using that column and renamed the other ‘Population\_2015’. Finally I multiplied that one column by 1000 to make sure both dataframes matched.

Merge:

I used the merge method in pandas to mergo on the country column. Any country that was available in only one of the two datasets was dropped. Ended up with 164 countries in the dataframe. I created a new column dividing the arrivals in one column by the population in another. I named the new column ‘Arrival\_Factor’. Dropped the year column because it was redundant.

**Load:**

I imported sqlalchemy to access mySQL. I created a connection engine using data for my local host using my password and existing database name. I then Used the *to\_sql* method in pandas to create a new table on the existing database. I verified the data was written into mySQL workbench. I finally used the *read\_sql* method to read the database back into pandas to verify it is there.

The data on both sources was similar in that they were both a rows of countries. I knew ahead of time what the final columns would be. Because I don't expect to need drastic changes I chose to use a SQL database.

I ended up with one database with one table.The table has 164 rows of countries each with population, international arrivals and the factor of arrivals and population in the year 2015.

This database could be used to look for countries with alot of arrivals compared to the population. The raw number of arrivals would naturally be bigger for bigger countries like the US or China. This way someone could see countries that are popular destinations for international travellers even if the country is mall. I think a database like this can be used for identifying new emerging travel destinations. The data can be extended to a multi year dataset. This dataset could then be used to look at past trends and forecasting the future influx of international arrivals.