**Transformation Steps**

Winemag-130k-v2.csv

* The ‘Winemag-130k-v2.csv’ dataframe was called.
* The first column was initially unnamed but changed to ‘id’.
* Then, the 'id', 'description', 'province', 'region\_1', 'region\_2', 'taster\_name', 'taster\_twitter\_handle', and 'variety' columns were deleted.
  + This was so that only the ‘title’, ‘vintage’, ‘country’, ‘winery’, ‘designation’, ‘points’, ‘price’, and ‘source’ columns remained.
* Next, several lines of code were written to keep only the rows that had values in specified columns.
* Perform groupby with mean aggregation to look at average point and price for each title.
* Perform sort\_values to determine the Top 5 & Bottom 5 popular titles.

Wines.xlsx & Combining

* The ‘Winemag-130k-v2.csv’ data & cleanup code was taken with just a few changes.
  + Instead of columns to drop, columns to keep were specified.
  + The ‘id’ column was kept.
  + A copy of the first dataframe was made.
  + All NaN values were dropped rather than just those from certain columns.
  + A row count was performed.
  + Each step stored the dataframe in a new variable.
* The same steps were then used to cleanup ‘Wines.xlsx’.
* The ‘Wines.xlsx’ dataframe was appended onto the ‘Winemag-130k-v2.csv’ dataframe to make ‘wine\_df\_final’.
  + It was originally proposed to use a join statement, but there were issues with finding a unique primary key to join the data on (which is admittedly odd, since the ‘title’ contains a composite of ‘winery’, ‘vintage’, and ‘designation’).
  + Since the data came from separate sites, it was assumed that there wouldn’t be duplicates.
* The ‘groupby’ function was used in conjunction with the ‘mean’ function to return a dataframe with the mean ‘points’ and ‘price’ values by each ‘title’ as ‘wine\_df\_grouped’.
* The ‘sort\_values’ function was run on the resultant dataframe on ‘points’ to show the top 5 & bottom 5 popular wine titles.