**## Bonus: Other Recommended Analyses**

**\* The following are optional challenge queries. These are highly recommended to attempt, but not required for the homework.**

**### Temperature Analysis I**

**\* Hawaii is reputed to enjoy mild weather all year. Is there a meaningful difference between the temperature in, for example, June and December?**

**\* You may either use SQLAlchemy or pandas's `read\_csv()` to perform this portion.**

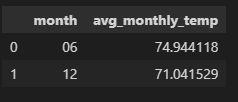
-----------------------------------------------------------------------------------------------------------------------------

*SELECT substr(date,6,2) as month, avg(tobs) as avg\_monthly\_temp*

*FROM measurement*

*where substr(date,6,2) in (‘12’,'06')*

*group by month*

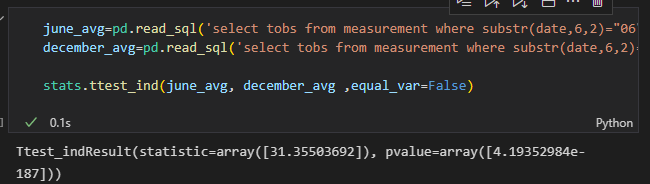
**

*That reputation is well deserved as can be seen by the above figure. Not much difference between December and June average temperatures*

\* Identify the average temperature in June at all stations across all available years in the dataset. Do the same for December temperature.

\* Use the t-test to determine whether the difference in the means, if any, is statistically significant. Will you use a paired t-test, or an unpaired t-test? Why?

*I will a paired T-test as the means came from the same sample.*

**

*The t value is very close to zero so that indicates that the 2 groups are similar*

-----------------------------------------------------------------------------------------------------------------------------

### Temperature Analysis II *(see Juypter notebook and python file for responses to these questions)*

\* The starter notebook contains a function called `calc\_temps` that will accept a start date and end date in the format `%Y-%m-%d`. The function will return the minimum, average, and maximum temperatures for that range of dates.

\* Use the `calc\_temps` function to calculate the min, avg, and max temperatures for your trip using the matching dates from the previous year (i.e., use "2017-01-01" if your trip start date was "2018-01-01").

\* Plot the min, avg, and max temperature from your previous query as a bar chart.

\* Use the average temperature as the bar height.

\* Use the peak-to-peak (TMAX-TMIN) value as the y error bar (YERR).

*The y error bar indicates there may be some variations in determining the average temperature.*

**Chart, box and whisker chart

Description automatically generated**

### Daily Rainfall Average

\* Calculate the rainfall per weather station using the previous year's matching dates.

\* Calculate the daily normals. Normals are the averages for the min, avg, and max temperatures.

\* You are provided with a function called `daily\_normals` that will calculate the daily normals for a specific date. This date string will be in the format `%m-%d`. Be sure to use all historic TOBS that match that date string.

\* Create a list of dates for your trip in the format `%m-%d`. Use the `daily\_normals` function to calculate the normals for each date string and append the results to a list.

\* Load the list of daily normals into a Pandas DataFrame and set the index equal to the date.

\* Use Pandas to plot an area plot (`stacked=False`) for the daily normals.

Chart, line chart

Description automatically generated

### Copyright