precipitation-viz project overview

Irene Hu

INFO 474

3/11/23

Feasible User Tasks

There are a number of possible tasks that can be done with this visualization, including:

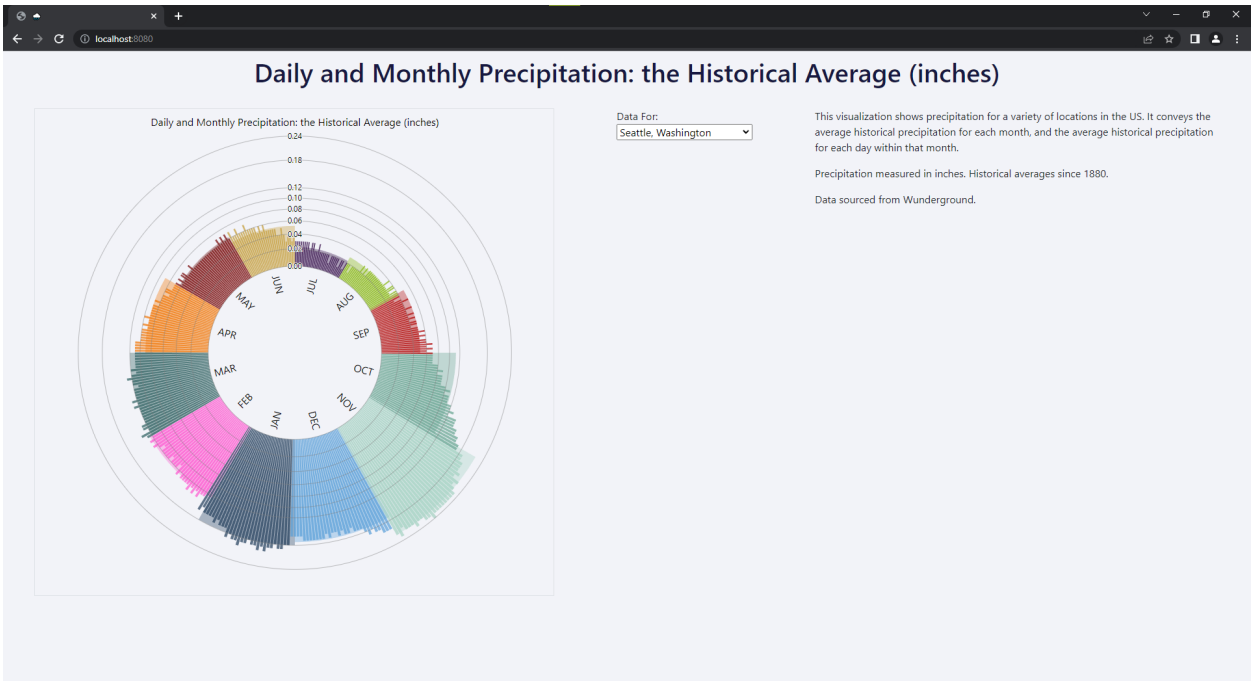
- Identifying the month with the highest average precipitation and the lowest average precipitation within a specific location.
- Identifying if precipitation within a specific location is consistent throughout the year, or seasonal
- Comparing the general precipitation trends in early June between Seattle, Washington and Phoenix, Arizona.
- Comparing the average precipitation for the months March and May within Seattle, Washington.

Design Overview

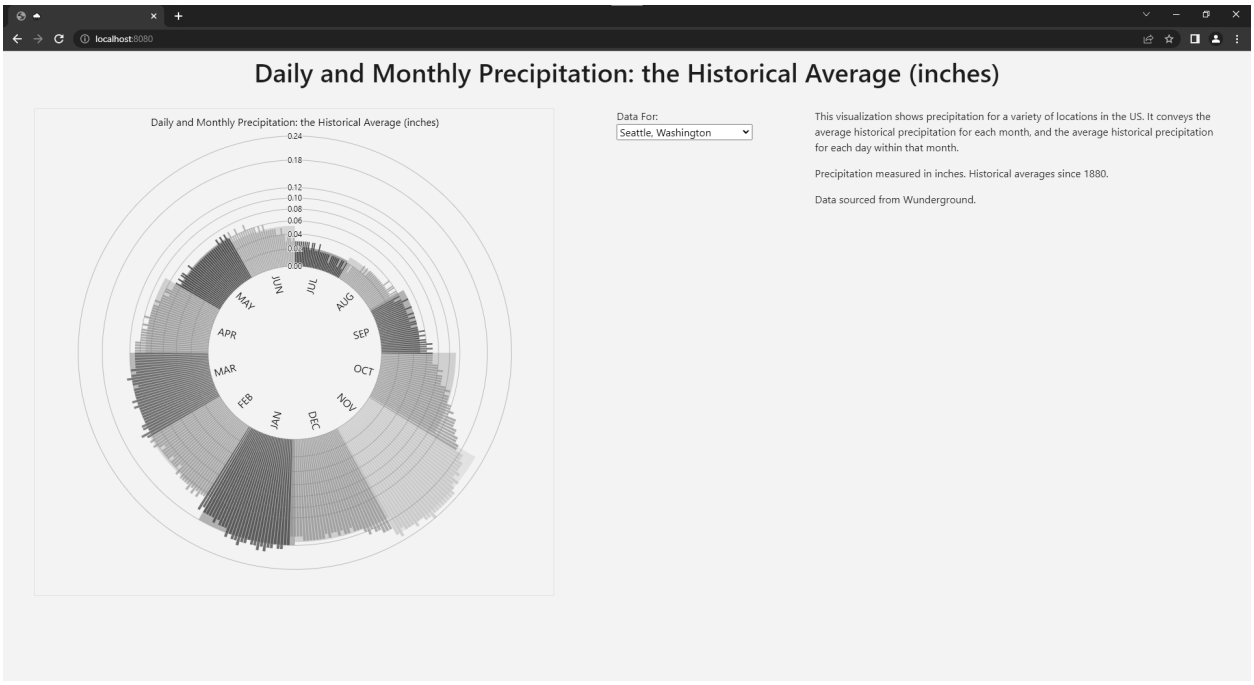
The design consists of a radial graph that shows the the historical average of each day within the year, as well as the calculated historical average of each month within the year. Averages were chosen in order to show the overall, general seasonal trends for a single location - for digestibility. The radial axis is incremented by 0.02 inches for values under 0.1 inch, and values divisible by 6 after - this choice was made because the original version that included all increments was too visually crowded and became overwhelming. Originally, the colors were planned to be all blue colors, but the choice to include a variety of colors was included to reinforce differences between each months. The colors were chosen to be differentiable from each other with and without greyscale for accessibility. Each month was also labeled within the graph, starting from the bottom and moving clockwise.

- averages : wanted to show what to expect! - general, seasonal trends for a single location. digestible information.
- slider - filter the amount of precipitation for each location.

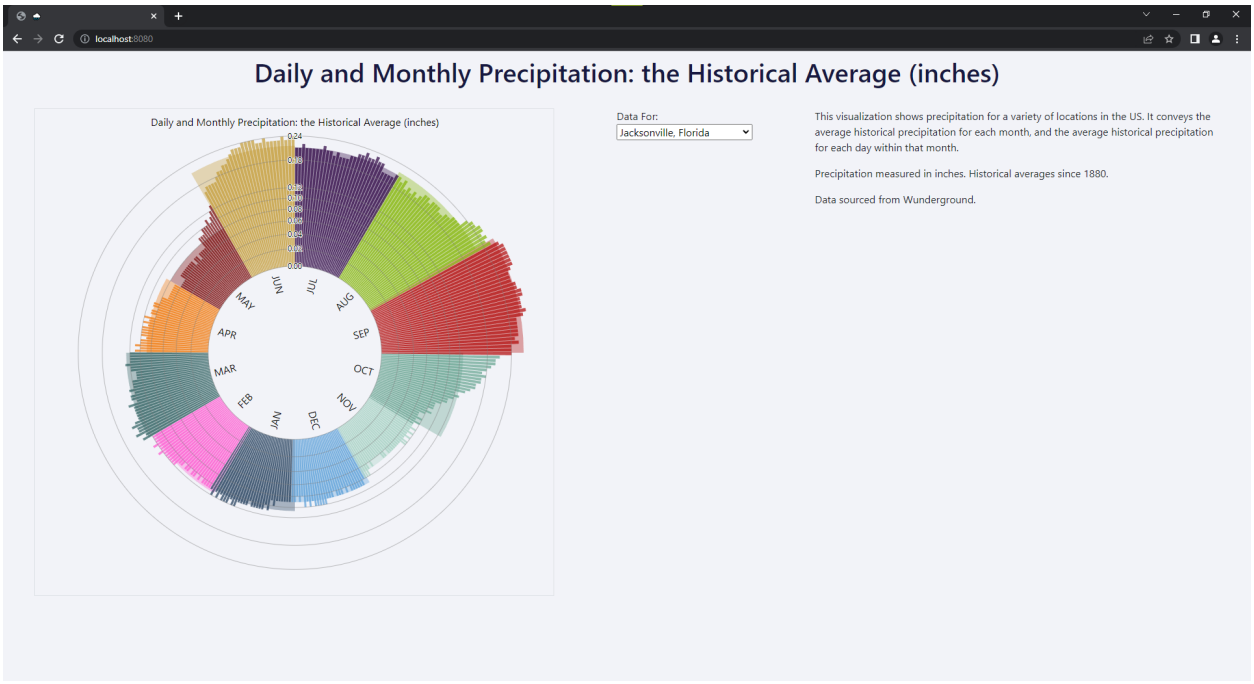
Screenshots



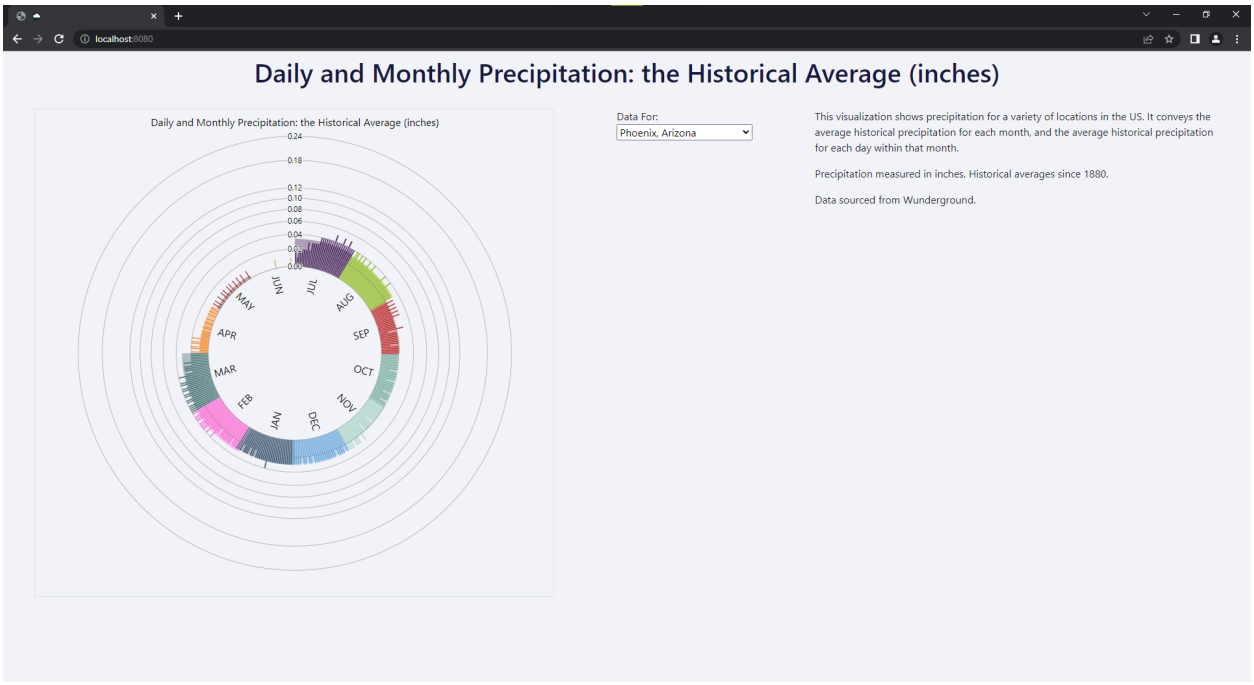
trends for Seattle



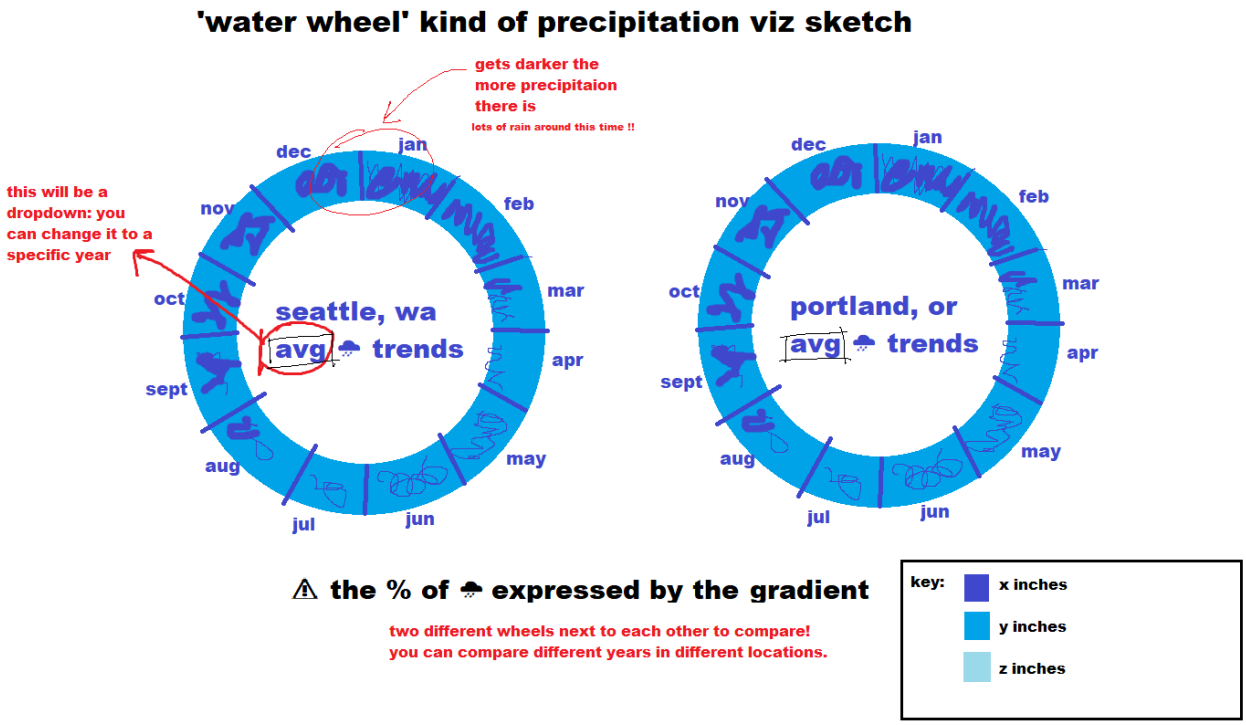
trends for Seattle, grayscaled



trends for Jacksonville, Florida



trends for Phoenix, Arizona



initial sketch concept for ‘water wheel’ visualization

Additional Notes

- to start server, navigate to folder in cmd and use: `python -m http.server 8080`
- most difficult part was working with radial graphs - working outside of x and y axis common in bar and line graphs gave a lot of headache and i ran into a lot of roadblocks
- if i had more time, i work out how to implement the double view - as we learned in class, comparing things side-by-side is much easier than back-and-forth.