TimeSearcher

Benjamin Anderson

Running the code

To run, just run a local server with Python from the root directory: `python -m http.server`. Or, just visit https://andersonbcdefg.github.io/timesearcher/ (https://andersonbcdefg.github.io/timesearcher/).

Important Files

- **index.html:** Skeletal webpage into which JavaScript code is injected.
- **main.js:** Main routine to run the application; mostly calls functions from `lib.mjs`.
- **lib.js:** Module containing important functions to build the application, including constructing the plot and adding event handlers.
- **main.css:** Styling for the plot and container page.

Commentary

I finished on Wednesday, October 14, and started the previous Friday. Probably spent a couple of hours each of those days on average, so I would estimate the whole process took around 15 hours. As expected, drawing the lines was not the hard part, and the interaction was--especially the logic to draw and resize rectangles. I had to think about it for a while and incrementally add features (add a rectangle of fixed size; then let the user draw it; etc.). Deleting was pretty easy, and resizing was hard. I ended up doing it in by designating one corner as the one that can be dragged, which is intuitive enough and works pretty well.

I anticipated that filtering would be really hard, but once I figured out *conceptually* how to do it, the coding wasn't so bad. Since D3 selections already have a `filter` method I just had to write the code to determine if a series passes through the box. One edge case is a box that is so narrow that it doesn't actually contain any data (i.e. it is drawn *between* months). To handle this case, I wrote code to interpolate between the two nearest months. Finally, doing the extra credit only took a few minutes! Once all the filtering logic was there for time boxes, adding a filter for names was straightforward.

In completing this project, I found a couple of helpful code snippets that I should officially cite here. This block (https://bl.ocks.org/larsenmtl/e3b8b7c2ca4787f77d78f58d41c3da91) was a good example for drawing a line chart, though I also referred to Dae Hyun's Observable notebook. And this block (https://bl.ocks.org/michaelwooley/b095fa7ce0e11d771dcb3f035fda1f07) was my starting point for drawing rectangles with the mouse, though of course I had to provide much more functionality than this basic example.