Scooby Doo Visualization

From the start, I knew I wanted to work with dataset that was fun for me to wrangle and visualize. One of my go-to places to find interesting datasets is the tidytuesday Github repository. I went through all the datasets from 2021 and found the Scooby Doo dataset. This was a manually collected dataset from Kaggle and came with over 15 variables and nearly 600 rows. It was perfect for my purposes.

Before I detail the process, here is the entire source code and the live visualization.

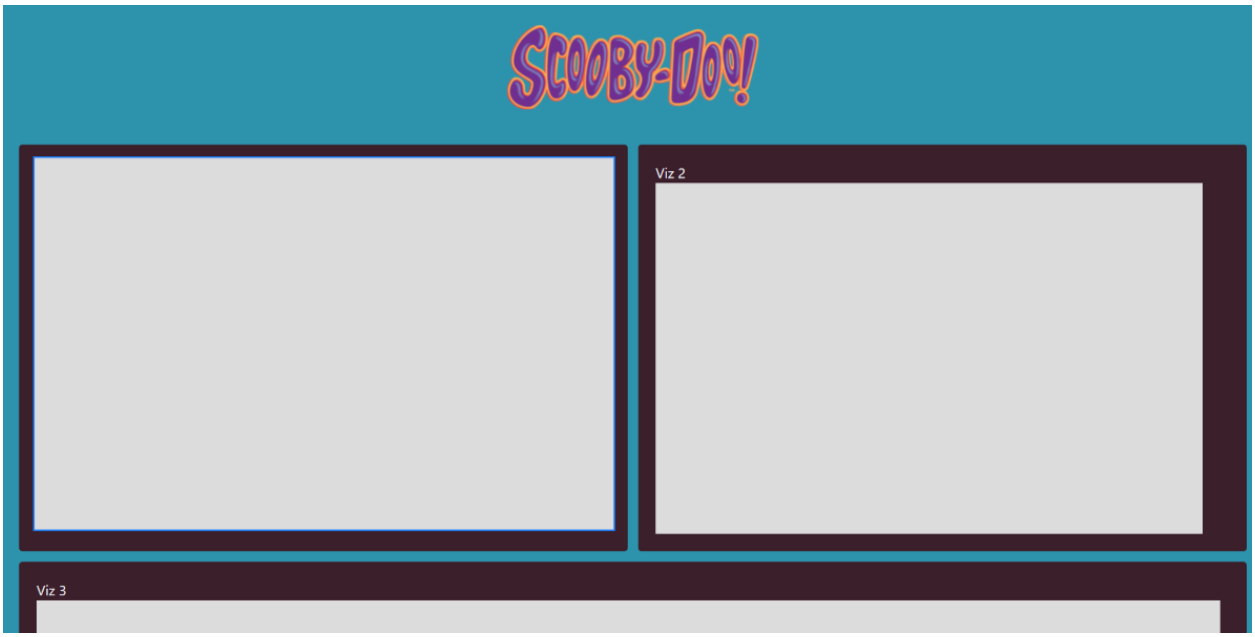
Scooby Doo Dataviz

Insights from the Scooby Doo series (1969 to 2020).

https://scoobydooviz.netlify.app/

https://github.com/thedivtagguy/scoob

With so many variables, I knew I wouldn't be able to select just one aspect of it to visualize, so I decided to make a patchwork of 2-3 individual visualizations. This wasn't possible in the p5.js web editor, so I downloaded the p5.js extension on Visual Studio code to create a development environment there. I also included a copy of Tailwind CSS, since I use it for all rapid-prototyping work. I created a grid with three large cells (or 'cards') like so:



For this, I would need a different canvas in each card. This is not possible by default. After navigating through a lot of Stackoverflow questions, I found that p5.js included an instancing mode which would allow me to link each canvas to a specific div-id.

Global and instance mode · processing/p5.js Wiki

p5.js is a client-side JS platform that empowers artists, designers, students, and anyone to learn to code and express themselves creatively on the web. It is based on the core principles of

https://github.com/processing/p5.js/wiki/Global-and-instance-mode

processing/**p5.js**

p5.js is a client-side JS platform that empowers artists, designers, students, and anyone to learn to code and express themselves...

461 Contributors · 189 Issues · 17k Stars · 3k Forks

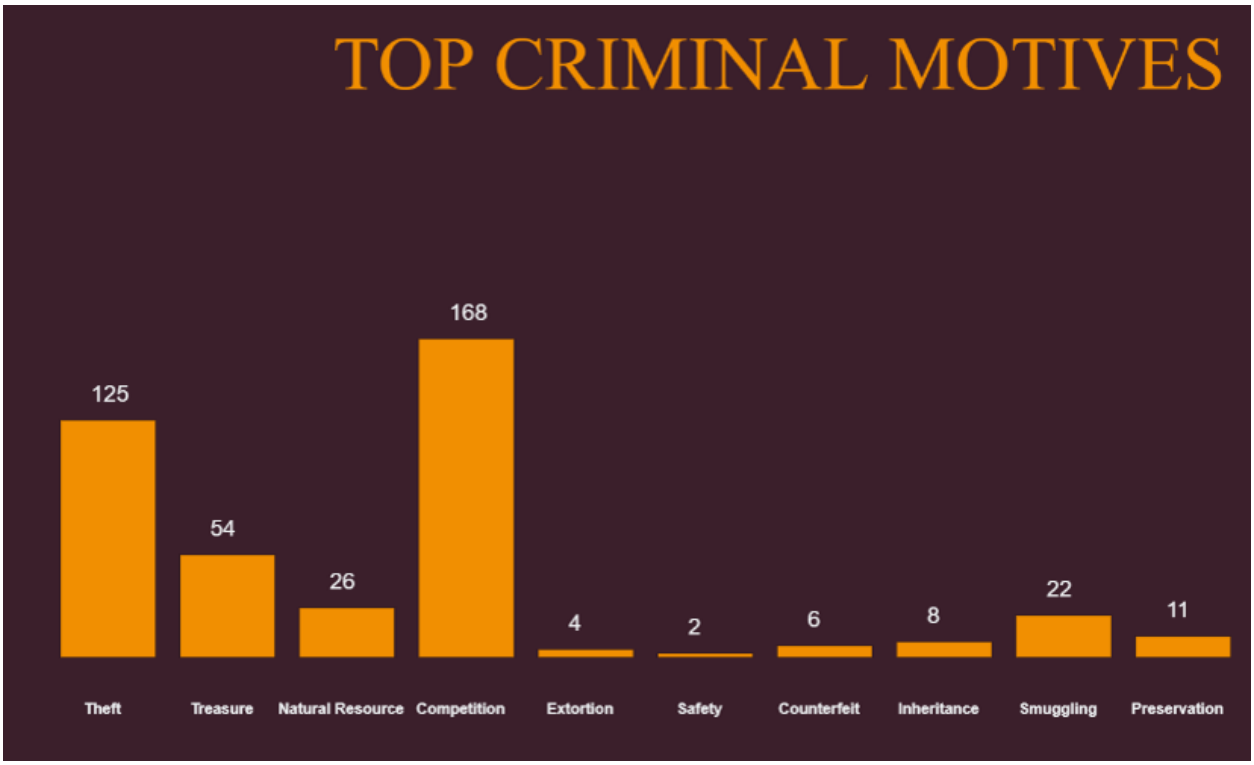
The code would be written like so:

```
const s = ( sketch ) => {
  sketch.setup = () => {
    sketch.createCanvas(200, 200);
  };

  sketch.draw = () => {
    sketch.background(0);
    sketch.fill(255);
    sketch.rect(x,y,50,50);
  };
};
```

```
let myp5 = new p5(s);
```

Next, I downloaded the dataset and set about trying to figure out how to access and transform data into visualizations. This was challenging because I am used to D3 and p5.js does not have as many pre-built options for primitives such as axes. My first visualization was a bar chart of the top motives.



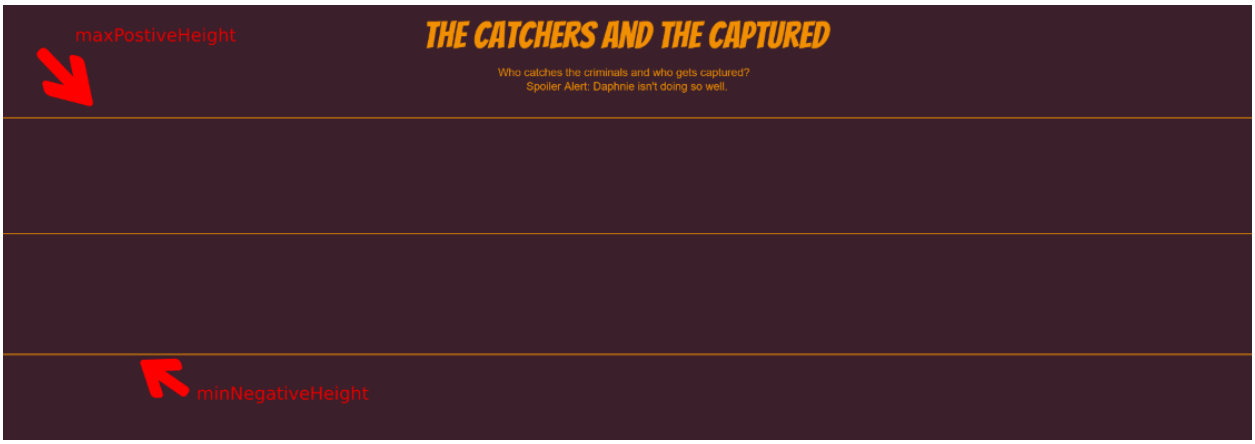
This taught me a lot about how to access data and use for loops for rendering shapes without copy-pasting a lot. With these basics in mind, my process went a lot faster from here on. I narrowed down my questions of interest as follows:

1. What are the top motives?
2. Who gives Scooby his snacks?
3. What are the trends in who gets captured and who catches criminals?

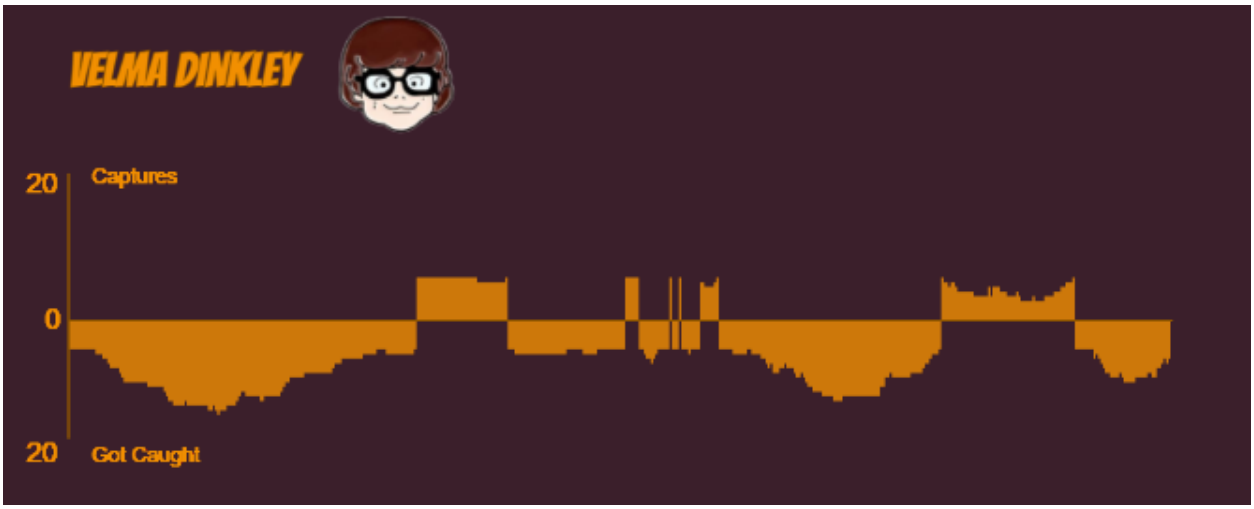
While the first two questions were easy to answer and visualize, since they were simple frequency counting exercises, the third one proved a lot harder to answer. I realized that this data could not be manipulated in Javascript alone and that I would need to bring in R. The decision proved fruitful and I constructed a second data frame with the following variables:

1. Character name
2. Index (Chronological episode number)
3. Value (-1 for being captured by a monster, +1 for catching one and 0 for no activity)
4. Cumulative rolling sum of values (to visualize the overall trend over a larger, 50 episode window).

I decided to visualize the data in the form of an area chart, by mapping the -1 and +1 values between a certain height like so:



Hence, anything below the main central axis would be considered as a capture and anything above as a catch. This was not easy for me to understand and I ran into a lot of problems, but in the end I figured out how to create a function that would render such a graph for each character, like so:



The purpose of this would be mainly observing the overall trend rather than displaying individual data points.

Final Visualization

The resultant visualization is viewable at the link below

Scooby Doo Dataviz

Insights from the Scooby Doo series (1969 to 2020).

https://scoobydooviz.netlify.app/

And the source code for all p5Js charts is in my Github repository

https://github.com/thedivtagguy/scoob