ABL Coding Challenge

The goal of this challenge is to create a simple web application that plots data received over a WebSocket connection in real-time.

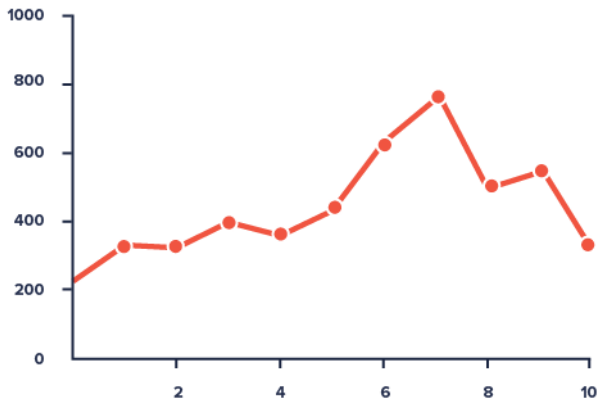
Your application will receive data from multiple sources labeled, "A", "B", and "C", but some sources will join later than others. Your solution should allow the user to select between these three sources and view a live plot of the selected source.

We have provided a simple WebSocket server that publishes the data. See `server/README.md` for instructions on running it. Once running, your client can connect to `ws://localhost:8080` to begin receiving data.

Requirements

We are looking for well-structured code and a functional application that meets the following requirements:

- The application should run in a modern web browser.
- The application provides a plot of the selected data source and updates the plot in real time as it receives new data. As an example, your plot could be as simple as this:



- Switching to view a different plot should preserve the data of the previously selected plot and continue to update in the background. In other words, the data for the sources not being displayed should still accumulate in the background.
- The application should be capable of running indefinitely, and it should always display a readable graph (for example, it should not run out of memory).
- Do not modify the server. This project is limited to building a front-end.

- Don't worry about aesthetic design or styling beyond basic functionality and usability.

Guidelines

- Think about the structure of your code and project organization (pay attention to names, folder structure, etc). Don't put all of your code into one file.
- Feel free to use any other additional libraries or frameworks such as React, but this is not required.
- Keep it simple.
  - We encourage you to use an existing library to render the actual plots; don't spend time writing your own plotting framework.
- Please make commits as you work on the challenge. There should not be a single commit that contains your entire solution.
- Provide instructions for running the application.

There is no time limit. When you are ready, submit your solution to `dave@ablspacesystems.com`.

Thanks! We look forward to seeing your work.