

D3.js

In this exercise we will look at the D3.js JavaScript library

Exercises

1. Create a blank HTML file with a CSS section and a JavaScript section.

```
<!DOCTYPE html>
<html>
  <head>
    <title>D3.js Demo</title>
    <meta charset="UTF-8">
    <style type="text/css"></style>
  </head>
  <body>
    <script type="text/javascript"></script>
  </body>
</html>
```

2. Include the d3.js (V4 or V5) and jQuery JavaScript libraries within the head tags of your HTML document. Use a CDN to provide/link the libraries on the fly
3. Using the lecture slides as reference: download the JSON EUR/USD file using Ajax from the JSONBlob server (<https://jsonblob.com/api/904557502042226688>). Verify that the file has downloaded correctly by performing a `console.log()` of the JSON data.
4. Create a function called `plotCurrencyData()` that will execute once the JSON has completed downloading.
5. In `plotCurrencyData()`:
 - 5.1. Setup the SVG size and margins
 - 5.2. Next, create x and y scales for EUR/USD value (y-axis) and time (x-axis) - Modify `d3.timeParse` to achieve this (Consult API)
 - 5.3. Create axis (x and y) and line (`d3.line()`) objects - specify number of ticks in the axis
 - 5.4. Append an SVG element to the webpage
 - 5.5. Next, add axes and the line object to the SVG element (within a group `<g>`) - Draw x-axis on bottom and y-axis on left.

5.6. Make the graph line red.

5.7. Add labels for axes, increase default font sizes and add a chart label (Consult API for this)

Advanced exercises

1. Use D3 to add zoom functionality (only to the x-axis)
2. Implement brush and zoom as illustrated in:
<https://bl.ocks.org/EfratVil/92f894ac0ba265192411e73f633a3e2f>

Notes

- [D3.js website](#).