Universitat de Girona
Fundació UdG: Innovació i Formació

Visualization for Transparency Foundation

MVTEC

Data Visualization Programming Anton Bardera Data Visualization Programming Anton Bardera Data Visualization Programming Anton Bardera **Data Visualization Programming** Anton Bardera Data Visualization Programming Anton Bardera Data Visualization Programming Anton Bardera Data Visualization Programming Anton Bardera Data Visualization Programming Anton Bardera Data Visualization

Observable

- Observable helps you sketch with live data and prototype visualizations
- Many users share their notebooks
- A notebook is made up of a series of cells, and each cell is defined by its JavaScript source code
- Simple
- Allows an easy prototyping
- Easy to share with your colleagues
- Created by Mike Bostock (again!!)

Observable

- How can we get our notebooks and place them on a “standard” webpage?

- “Code” cells in a notebook are separate scripts that run independently
- In our examples, we could directly assign the different cells in a new **var** elements on a JavaScript code
- We can try to do it better and structure it in logical functions

Things to consider

Integrated development environment (IDE)

- provides comprehensive facilities to computer programmers for software development
- I use Visual Studio Code, but there are many more
- *LiveServer* extension is a useful tool
- Web navigator inspector tools are also very useful

Things to consider

Promise

- The Promise object represents the eventual completion (or failure) of an asynchronous operation and its resulting value
- Used to wait until the data is loaded
- D3 also provides support for that (d3-fetch module):

```
d3.json("./data/dataset.json") //.csv, .tsv, .xml,...
  .then(data =>{
  ...
})
```

- To load multiple files at once, you could use Promise.all

Things to consider

Functions and code structure

- Useful to give structure and legibility to our code
- We can define functions to:
  - prepareData
  - drawChart
  - ...
- Insert comments to clarify the code
- Indentation is also very important for the legibility

Things to consider

var, let, and const

- **var** declares a global scope variable for the function regardless of block scope. Allows hoisting
- **let** declares a local scope variable for the function, or block. It is reassignable and does not allow hoisting
- **const** declares a local scope variable for the function or block. It is not reassignable, but it is mutable. It does not allow hoisting
- Try to use **let** and **const**, since they allow more control on the objects

Things to consider

SVG

- Typically, we will have an empty **svg** element on the HTML page with a specific id
- In Observable, we create a new svg element (and we set some attributes as width and height) and, at the end of the Observable cell, it is returned
- In a standard webpage, the empty svg element on the HTML is first selected and the different elements are appended. At the end it isn't necessary to return anything