

Data Processing

D3

Agenda

JavaScript

D3 - an introduction

D3 code

D3 scatterplot

Code Quality

JavaScript

JavaScript and HTML

Window.onload

Het is hetzelfde als de JS code tussen de <scripts> plakken
Window.onload = function(d) etc...

D3

Data-Driven Documents



JavaScript library voor dynamische, interactieve web based visualisaties

<https://github.com/d3/d3/wiki/Gallery>

<https://datavizcatalogue.com/index.html>

D3 code

D3 bar chart

D3 versus vanilla JavaScript

Other practical matters

D3.js, d3vsjs.js, libraries.html, debugging.js (debug + (d, i)),

D3 scatterplot

Create the HTML, CSS and JavaScript files

Load in D3

API data -> 3 or 4 variables

Append chart, create scales, plot the graph

Interactive HTML element (Button, dropdown, slider, etc.)

Update function

Using transition

Fourth variable, represented by the radius of the circles

Versie 5 lijkt heel veel op versie 4, dus kijk naar die tutorials/voorbeelden als je iets zoekt

Transition and fourth zijn extra credit

<https://www.gapminder.org/tools/>

htmlElements folder

Code Quality

Project

Validators

Rubric

Project

Files

Folders

Etc.

Opzetje project laten zien, zie code
HCCode -> project

Validators

HTML

CSS

JSON

JavaScript

<https://validator.w3.org/>

Jsonlint

Rubric

Documentation

Presentation

Algorithms

Structure

Documentation

Is the code well-annotated to ensure rapid understanding?

Comments

Names

Comments: consistent, inhoudelijk nuttig en waar, platform standards (`//` kleine letter en `/* whatever */`) Function headers/ file headers

Names: consistent, nuttig, waar, niet te lang, platform standards (camel) variabelen en files

Presentation

Is the code visually organized for a quick read?

Formatting

Organization

Formatting: Indentatie, spacing, brackets is consistent en verdeeld het in mooie blokken

Organization: GEEN UITGECOMMENDE CODE, GEEN DODE CODE, GEEN DEBUGGING (LOGS)

functies/variabelen in logische volgorde,

Algorithms

Is each part of the code as simple as possible?

Flow

Idiom

Flow: Flow is simple, expressions zijn simpel, scope is zo klein mogelijk, geen onnodige repetition

Idiom: Control structures zijn logisch en nodig, goed gebruik van ingebouwde functies/libraries, goede data types `bool = "false"

Structure

Is the code organized for quick understanding of parts and the whole?

Modules

Functions

Modules: JS files, zijn compact en hebben clear responsibilities

Functions: Normale length, degelijk aantal parameters, iedere functie doet eigenlijk maar 1 ding