



Tvorba interaktivní grafiky v prostředí D3.js

ZX801 Geotechnology Camp
Adam Mertel



Data-Driven Documents



Osnova

1. úvod, motivace
2. statická grafika
3. události a interakce
4. základní formy prezentace dat (pie chart, line chart,...)
5. kartografie a D3

1. ÚVOD

- **Autor** Stanford Visualization Group - Mike Bostock
- **Predchodca** - [Protovis](#)
- **Inšpirácia** - [Tufte](#), [Munzner](#)
- Aktuálne možnosti tvorby grafiky (interaktívnej?)
- Buzzwords : chaining, scalability, community based, visual analytics, real time, ...
- [bl.ocks.org](#), [bost.ocks.org/mike/](#)

1. MOTIVÁCIA

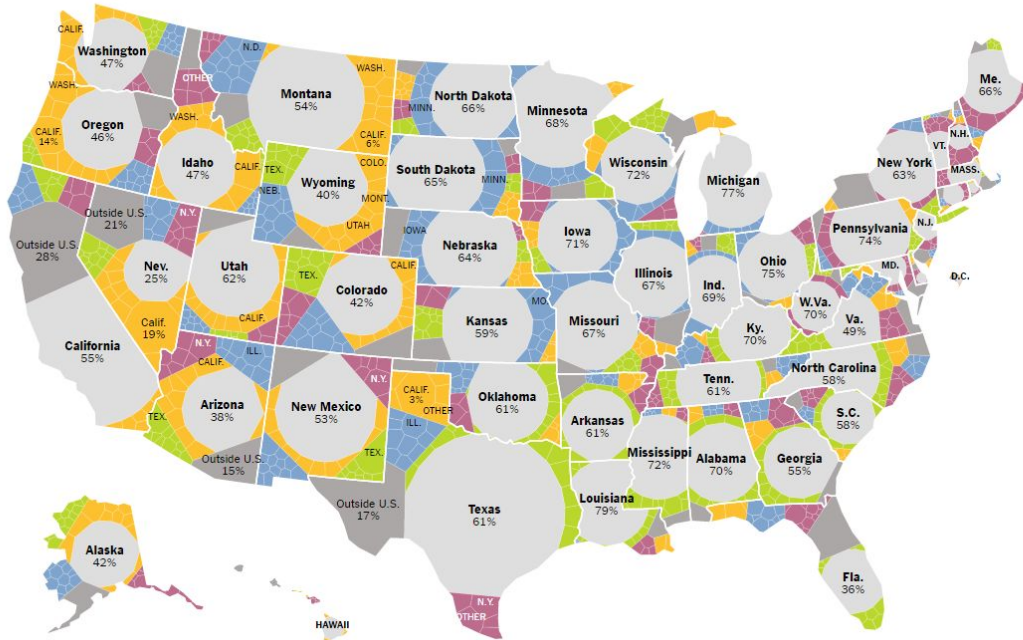
Where people who lived in each state in 2012 were born

Each shape represents where the people living in a state were born. Within a state, larger shapes mean a group makes up a larger share of the population.

■ Northeast ■ South ■ Midwest ■ West ■ Outside the U.S.*

SELECT A YEAR

1900 | 1950 | **2012**



<http://www.nytimes.com/2014/08/16/upshot/mapping-migration-in-the-united-states-since-1900.html>

1. MOTIVÁCIA

ORBIS The Stanford Geospatial Network Model of the Roman World

About Tutorial Walter Scheidel
Elijah Meeks

Route Network Flow

FROM: Constantinopolis

TO: Augusta Treverorum

DEPARTING: MONTH SEASON

❄️ 🌿 ☀️ 🌿

PRIORITY: Fastest Cheapest Shortest

NETWORK MODES

Road Coastal Sea
 River Open Sea
 High Resolution

MODE TRANSFER COST

ROAD Foot (30km/day) 0

RIVER Civillian 0

SEA Fast 0

Calculate Route

According to the Fastest routes from **Constantinopolis** to the rest of the Roman world in **July**, sites are this far away.

The most distant major sites are:

- Londinium (46 Days)
- Lugdunum (36 Days)
- Corduba (35 Days)
- Mediolanum (30 Days)

Cartogram
Zones

Distance from Constantinopolis

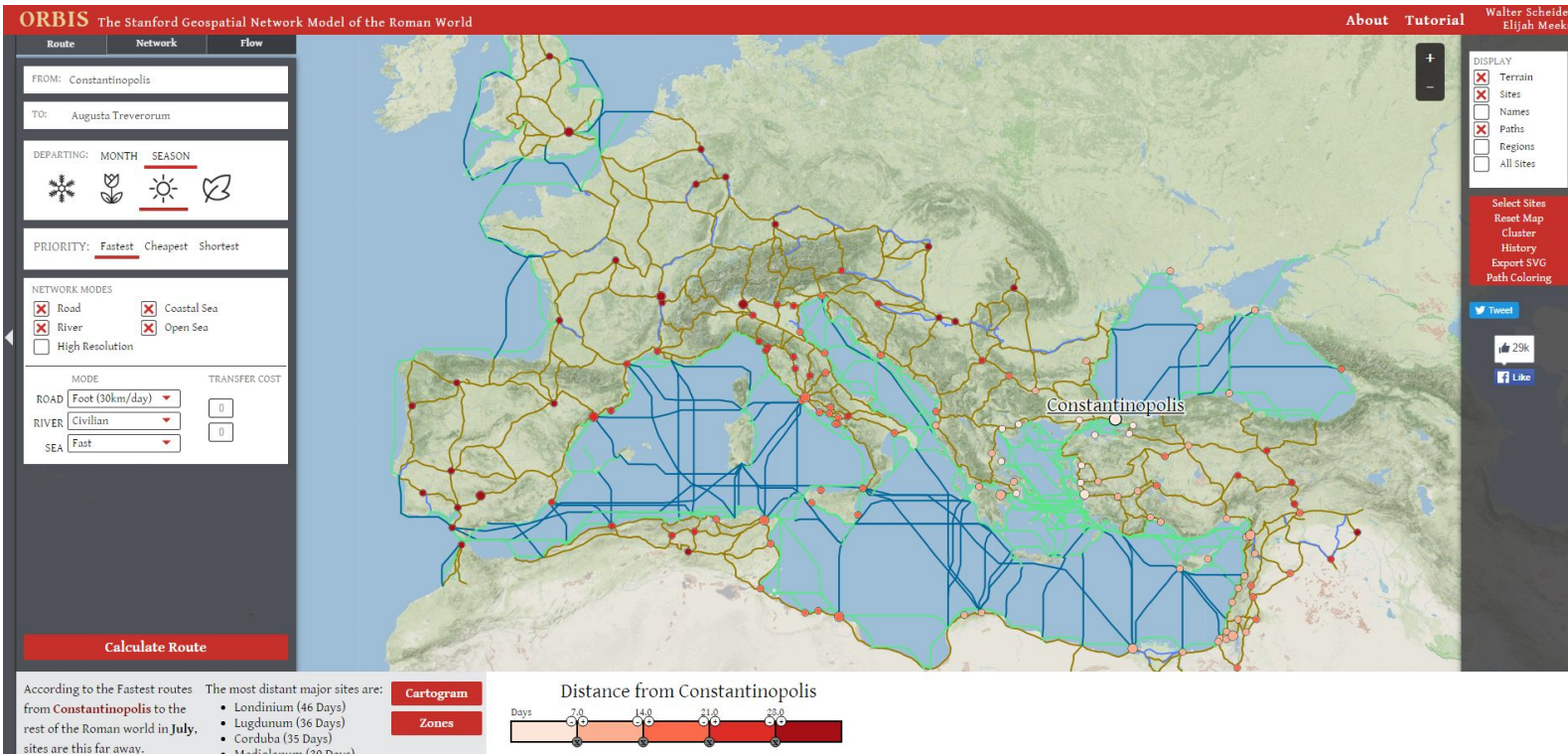
Days 7.0 14.0 21.0 28.0

DISPLAY

Terrain
 Sites
 Names
 Paths
 Regions
 All Sites

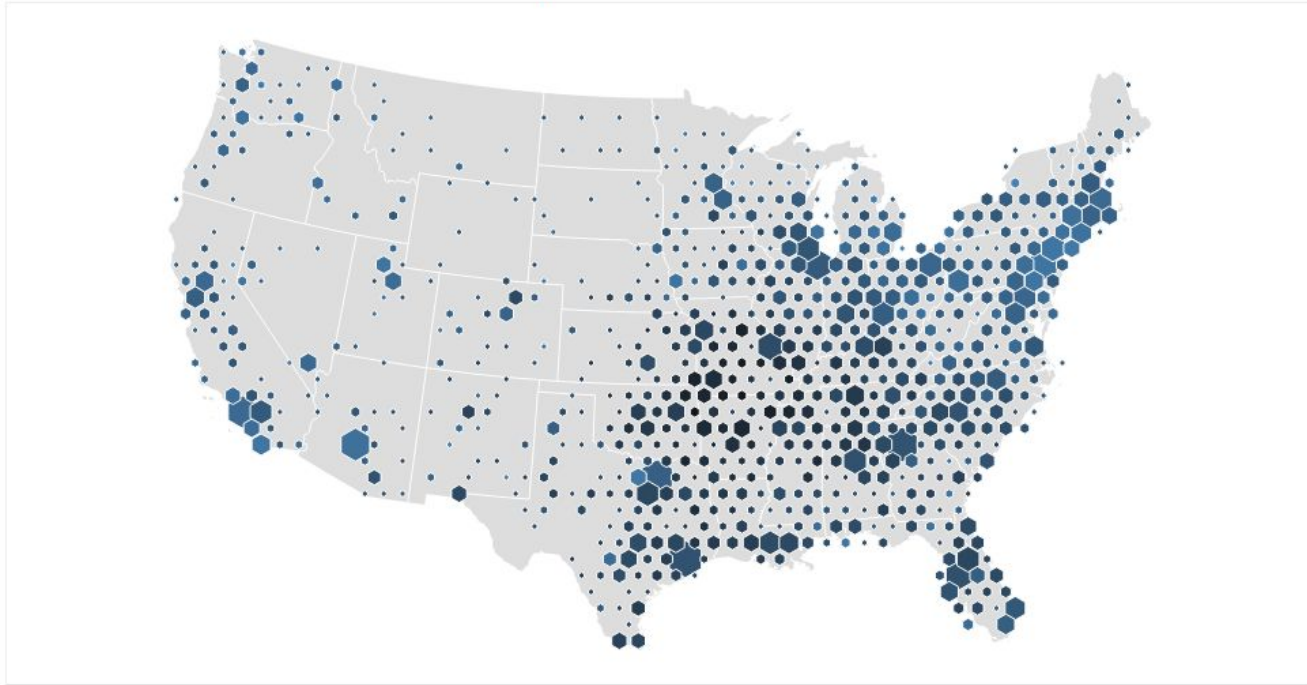
Select Sites
Reset Map
Cluster
History
Export SVG
Path Coloring

🐦 Tweet
👍 29k
👍 Like

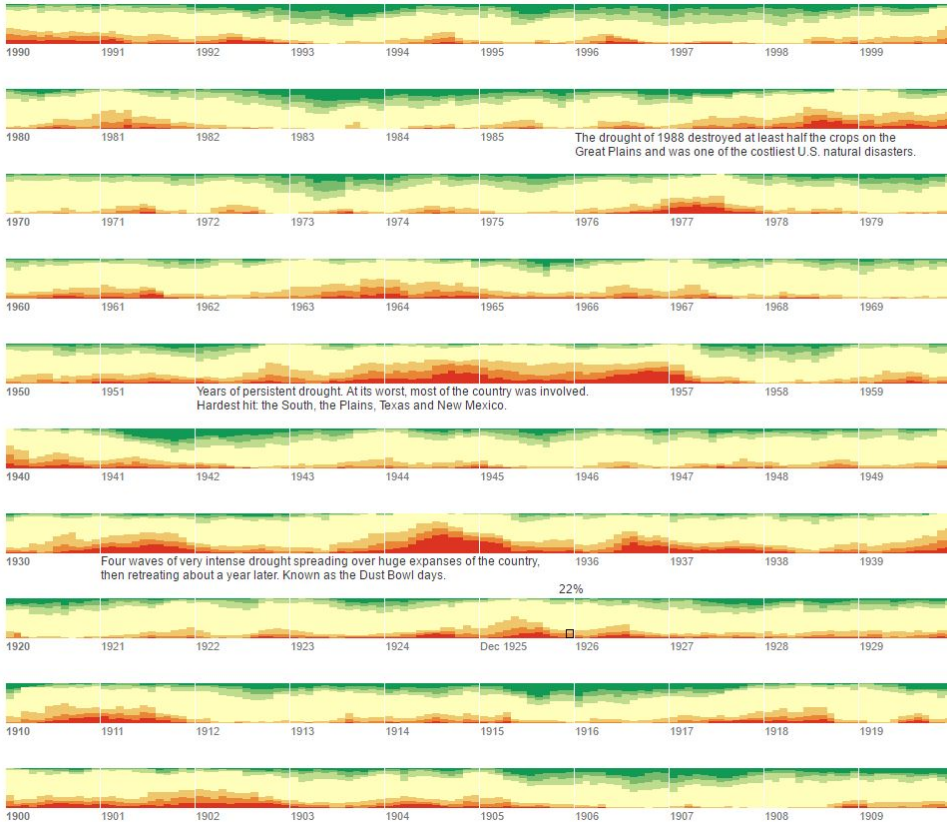


1. MOTIVÁCIA

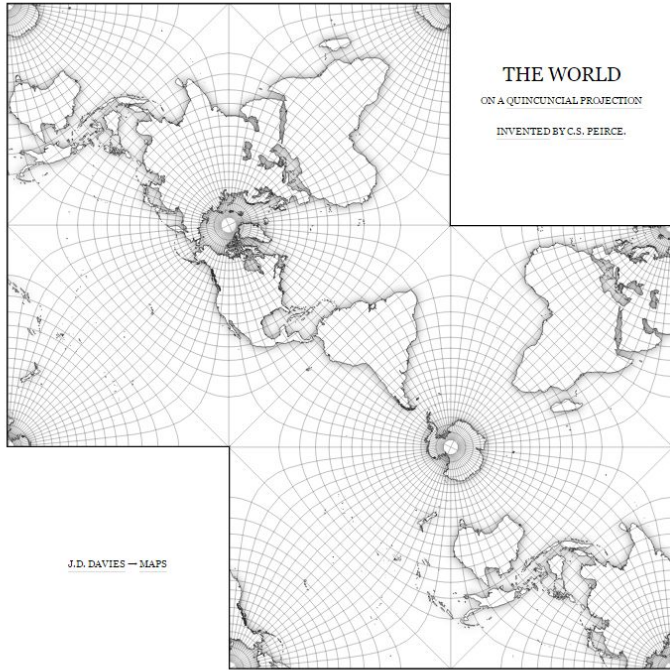
Bivariate Hexbin Map



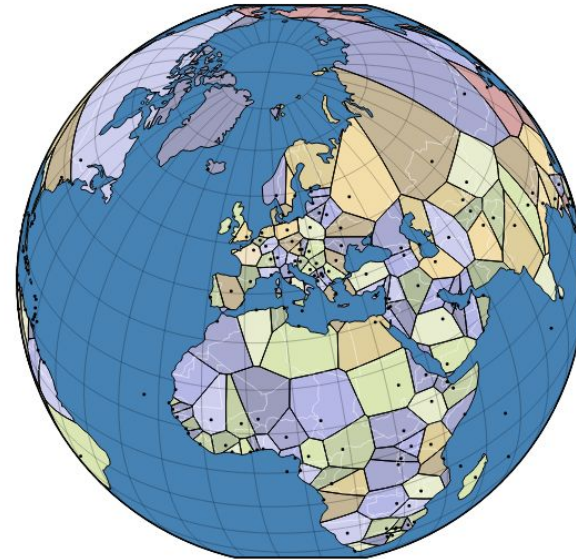
1. MOTIVÁCIA



1. MOTIVÁCIA

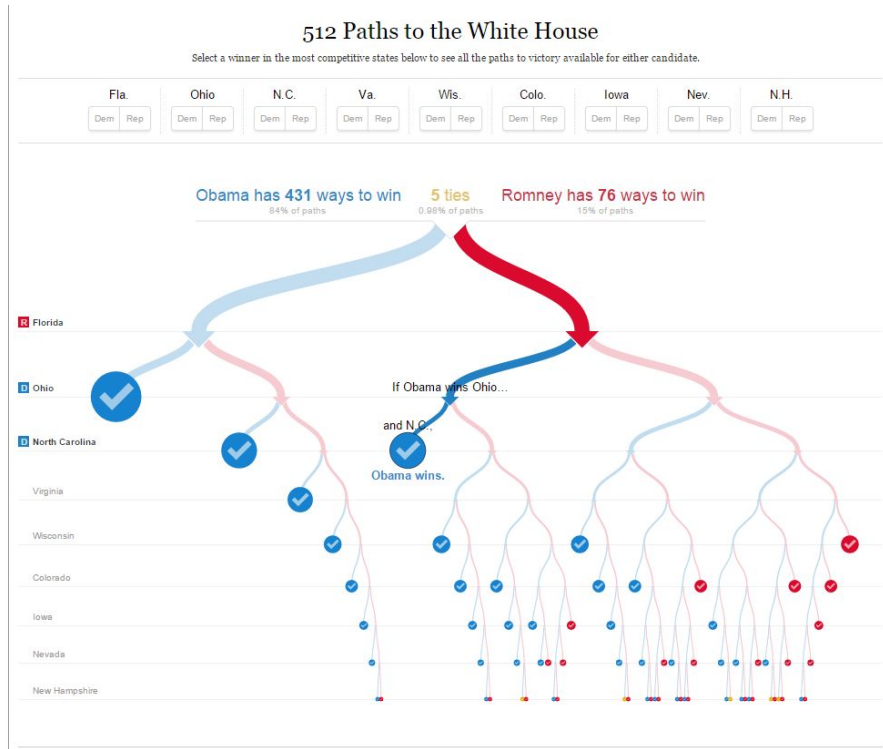


World Capitals Voronoi



Data: *Natural Earth, 1:10m Cultural Vectors: Populated Places (Admin-0 capitals).*

1. MOTIVÁCIA



[512 paths to the White House](#)

1. KONKURENCIA

- HTML5 canvas
- STATISTICA
- Css
- Excel
- Illustrator
- D3 - plot.ly, tableau,...
- Flash
- ...

D3 - kde a kedy ne/využiť

1. IDEA D3

- HTML + DOM - zobrazovanie
- SVG - organizácia grafiky
- CSS - štýlovanie
- Javascript + JQuery - manipulácia s dátami a naviazanie na html

1. CODEPEN

- <http://codepen.io/> (<http://codepen.io/Lewitje/pen/GjqbbA>)

Vending machine
A PEN BY Lewi Hussey

HTML

```
<div class="machine">
  <div class="inner">
    <div class="arm">
      <div class="hand"></div>
    </div>
    <div class="shelf" data-shelf="A">
```


CSS (SCSS)

```
@import 'https://fonts.googleapis.com/css?family=Pacifico|RobotoCondensed';

$bubblegum: #F4ACB7;
$mint: #84DCC6;
$cherry: #FF6868;
$orange: #FF8A5B;
```

JS

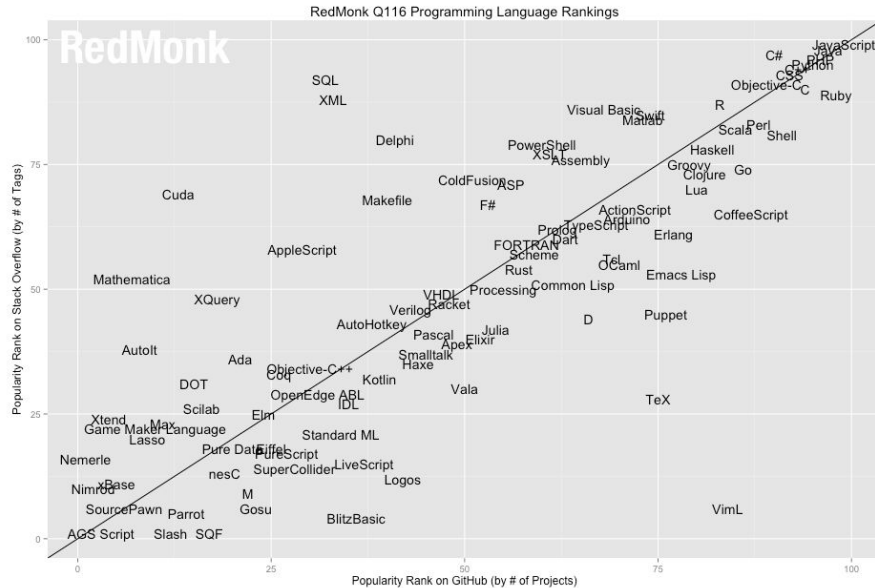
```
class VendingMachine{
  constructor(){
    this.setupLabels();
    this.setInitialDisabled();
    this.bindEventListeners();
  }
  this.scene = {
```



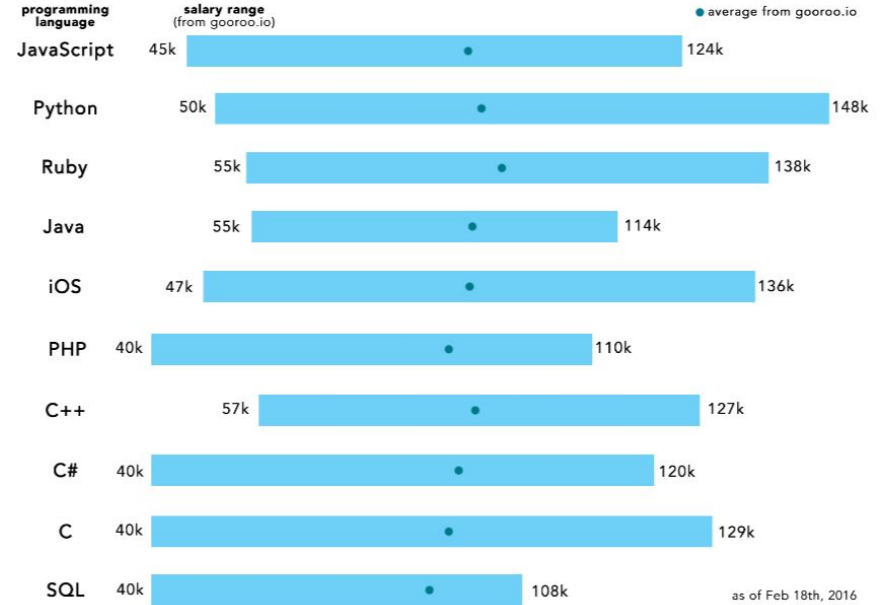
push

by @lewi

1. Javascript



2016 Programming Salary Range in the US



1. Javascript

- **Premenné** (*var a = 5*)
 - Pole (*var pole = [1, 5, 15]*)
 - Objekt/JSON (*var objekt = { "a": 1, "b": 2, "c": 3 }*)
- **Podmienka** (*if (a == 5) { a ++ }*)
- **Cyklus** (*for (var i = 0; i < a; i++) { console.log (i) }*)
- **Funkcia** (*function addOne (b) {return b + 1}*)

- <https://codepen.io/adammertel/pen/ozzQxQ>

1. DOM - HTML

- Document Object Model
- `document.getElementById('sth')`
- `document.getElementsByTagName('sth')`
- `document.createElement(element)`
- `document.appendChild(element)`
- **innerHTML**
- <http://codepen.io/adammertel/pen/EggOwp>

1. CSS

- Inšpirácia - <http://www.rleonardi.com/interactive-resume/>
- Kaskádové styly - štýlovacie pravidlá, pozicionovanie, interakcia, animácie, ...
- Dokumentácia - <https://developer.mozilla.org/en-US/docs/Web/CSS/Reference>

- **Inline** - pravidlá priamo do html
 - `<div style="background-color: red">`
- **CSS súbor**
 - **Triedy** (znamienko `.`) - `<div class="cervena" ></div>` - `.cervena { background-color: red }`
 - **ID** (znamienko `#`) - `<div id="cerveny-box"></div>` - `#cerveny-box {background-color: blue}`

- <http://codepen.io/adammertel/pen/EggqRx>

1. SVG

- XML based
- **<circle />** - cx, cy, r
- **<line />** - x1, x2, y1, y2
- **<text>**blablab**</text>** - x, y
- styling (<https://www.w3.org/TR/SVG/styling.html#SVGStylingProperties>)
- <http://codepen.io/adammertel/pen/QKKJZO>

1. JQuery

- Všeobecná knižnica na uľahčenie práce s html/css/js - znamienko \$

- `$(document).ready(function(){...})` - DOM je načítaný

- `$("#p")`, `$(".my-class")`, `$("#id123")` - výber elm z DOMu

- `text()`, `html()`, `val()` - práca s hodnotami

- `append()`, `prepend()`, `before()`, `after()` - vkladanie

- `addClass()`, `removeClass()`, `css()` - práca s css

- `$.getJSON("test.json", function(data) {`

```
    console.log( data );
```

```
});
```

- <http://codepen.io/adammertel/pen/JRbdjR>



Add a number to another number in JavaScript

hallo
0
I have got a number in my JavaScript variable! Now how do I add another number to it? Please
javascript

3 Answers

22
You should definitely use JQuery. It's really great and does all things
answered 11 minutes ago
k-3jQuery
1,234 ● 2 ● 13

I agree, JQuery is really the best, it solves all kinds of browser problems and is good, as well – |sum0da 8 mins ago
+1 jquery is best quality code ever, if you don't use your a idiot – Werry_Togan 4 mins ago
add comment

4
I think there's a jQuery plugin for that. Google for jQuery basic arithmetic plugin.
answered 5 minutes ago
Timothy Goatse
4,321 ● 1 ● 12

yeah, jquery is definetly the way to go – fishripples 5 mins ago

I used the jQuery diet plugin and lost 10kg in a week – fatty 4 mins ago
add comment

-2
To add numbers together you should use the + operator, for example:
var a= 1;
var b= a+2;
alert(b); // 3

link | edit | delete | flag

answered 50 seconds ago
bobince
some ● ● ● ● ●

-1 not enough jQuery – |sum0da 30 secs ago

you suck – Timothy Goatse 3 secs ago

2. STATICKÁ GRAFIKA - DOM

- **d3.select("body")** — Finds the body in the DOM and hands a reference off to the next step in the chain.
- **.selectAll("p")** — Selects all paragraphs in the DOM. Since none exist yet, this returns an empty selection. Think of this empty selection as representing the paragraphs that will soon exist.
- **.data(dataset)** — Counts and parses our data values. There are five values in our data set, so everything past this point is executed five times, once for each value.
- **.enter()** — To create new, data-bound elements, you must use enter(). This method looks at the DOM, and then at the data being handed to it. If there are more data values than corresponding DOM elements, then enter() creates a new placeholder element on which you may work your magic. It then hands off a reference to this new placeholder to the next step in the chain.
- **.append("p")** — Takes the placeholder selection created by enter() and inserts a p element into the DOM. Hooray! Then it hands off a reference to the element it just created to the next step in the chain.
- **.text("New paragraph!")** — Takes the reference to the newly created pand inserts a text value.
- <http://codepen.io/adammertel/pen/amBkXQ>

2. STATICKÁ GRAFIKA - SVG

- ```
var svg = d3.select("body")
 .append("svg")
 .attr("width", 500)
 .attr("height", 50);
```
- Svg geometries -
  - **circle** - cx, cy, r
  - **rect** - x, y, width, height
  - **text** - x, y, text
  - **ellipse** - cx, cy, rx, ry
  - **line** - x1, x2, y1, y2
  - **path** - d
  - ...
  - + štýlovanie
- <http://codepen.io/adammertel/pen/yaVrXx>

# 3. INTERAKTÍVNA GRAFIKA

- `var scale = d3.scale.linear().domain([0, 10]).range(["white","red"]);` -> `scale(5)` vráti ružovú
- `.on("mouseover", function(d, i) {... - mousemove, mouseout`
- `d3.event.pageX, d3.event.pageY` - vracia pozíciu kurzoru
- `remove()` - odstráni vybraný element
- `filter(function(d, i){return d == 1})` - berie iba dáta o zvolenej hodnote
- <http://codepen.io/adammertel/pen/XjNRBN>

# 4. ZÁKLADNÉ FORMY PREZENTÁCIE DÁT

- **Layouts** - <https://github.com/d3/d3-3.x-api-reference/blob/master/Layouts.md>
  - **Pie chart** - The pie layout simply makes it easier to convert an array of data into an array of objects with `startAngle` and `endAngle` attributes that range from 0 to  $2\pi$ , which you can then pass to the arc shape generator
    - `d3.svg.arc()`
    - <http://codepen.io/adammertel/pen/qaqorQ>
- **SVG Shapes** - <https://github.com/d3/d3-3.x-api-reference/blob/master/SVG-Shapes.md>
  - **Line chart** -
    - <http://codepen.io/adammertel/pen/ORbkaO>



# 5. MAPA

- var **projection** = d3.geo.gnomonic() + scale, center, translate ...
- d3.**geo.path**() -> vyrába path z geojsonu ()
  - <https://developer.mozilla.org/en/docs/Web/SVG/Tutorial/Paths>
  - <http://jxnblk.com/paths/>
- **scale** - choropletova mapa,...
- <http://codepen.io/adammertel/pen/NRbyrm>

# Zdroje

- [https://www.reddit.com/r/dataisbeautiful/comments/3k3if4/hi\\_im\\_mike\\_bostock\\_creator\\_of\\_d3js\\_and\\_a\\_former/](https://www.reddit.com/r/dataisbeautiful/comments/3k3if4/hi_im_mike_bostock_creator_of_d3js_and_a_former/)
- <http://benschmidt.org/D3-trail/minard.html>
- <https://plot.ly/javascript/>
- <http://bl.ocks.org/>
- [https://www.reddit.com/r/dataisbeautiful/comments/3k3if4/hi\\_im\\_mike\\_bostock\\_creator\\_of\\_d3js\\_and\\_a\\_former/](https://www.reddit.com/r/dataisbeautiful/comments/3k3if4/hi_im_mike_bostock_creator_of_d3js_and_a_former/)
- <http://alignedleft.com/tutorials/d3/>
- <https://www.dashingd3js.com>
- <https://www.jasondavies.com/>