IO Lab:
D3.js

October 28, 2013

INFO 290TA (Information Organization Lab)
Kate Rushton & Raymon Sutedjo-The

D3.js

A Javascript library for manipulating documents **based on data**.

Binds arbitrary data to the DOM, then applies data-driven transformations to the document.

Works in tandem with other web technologies.

D3.js

SVG

Recall SVG

```
<svg xmlns="http://www.w3.org/2000/svg"
version="1.1" width="500" height="200">

   <rect x="0" y="0" width="100" height="100"
fill="red"/>

</svg>
```

Recall SVG

```
<svg xmlns="http://www.w3.org/2000/svg"
version="1.1" width="500" height="200">

   <rect x="0" y="0" width="100" height="100"
fill="red"/>

</svg>
```

Data Binding

```
[50, 40, 30]
```

Data Binding

```
[50, 40, 30]
```

```
<rect>
```

```
<rect>
```

```
<rect>
```

Data Binding

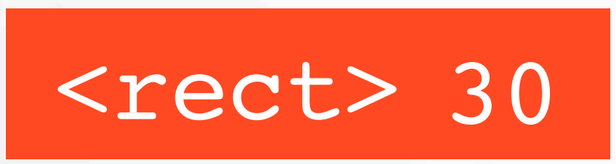
```
[50, 40, 30]
```

```
<rect>          50
```

```
<rect>      40
```

```
<rect> 30
```

Data Binding




```
var svg = d3.select("#viz");

svg.selectAll("rect")
.data([50, 40, 30])
.enter().append("rect")
.attr({
    "width": function(d,i) {
        return d;
    }
});
```

Breaking It Down

Select

Bind

Enter

Set

```
var svg = d3.select("#viz");

svg.selectAll("rect")
    .data([50, 40, 30])
    .enter().append("rect")
.attr({
    "width": function(d,i) {
        return d;
    }
});
```

Breaking It Down

Select
... the DOM element(s)

Bind

Enter

Set

```
var svg = d3.select("#viz");

svg.selectAll("rect")
    .data([50, 40, 30])
    .enter().append("rect")
    .attr({
        "width": function(d,i) {
            return d;
        }
    });
```

Breaking It Down

Select

Bind
... data point(s) to the selected element(s)

Enter

Set

```
var svg = d3.select("#viz");

svg.selectAll("rect")
    .data([50, 40, 30])
    .enter().append("rect")
    .attr({
        "width": function(d,i) {
            return d;
        }
    });
```

Breaking It Down

Select

Bind

Enter

... and add elements if the # of elements do not match the # of data points

Set

```
var svg = d3.select("#viz");

svg.selectAll("rect")
    .data([50, 40, 30])
    .enter().append("rect")
    .attr({
        "width": function(d,i) {
            return d;
        }
    });
```

Breaking It Down

Select

Bind

Enter

Set

... the element's attribute(s) and/or style(s)

```
var svg = d3.select("#viz");

svg.selectAll("rect")
    .data([50, 40, 30])
    .enter().append("rect")
.attr({
    "width": function(d,i) {
        return d;
    }
});
```

Extra Attributes

You can add **title & description** attributes to an SVG element.

This may come in handy if you need to store datum-specific metadata.

```
<rect width="75" height="20" x="0" y="10"
title="Rating" desc="75/100"></rect>
```

Extra Attributes

You can also add **custom** attributes to an SVG element.

Although it works in practice, custom attributes are not officially supported by W3C (yet), so use with caution.

```
<rect width="75" height="20" x="0" y="10"
rating="75/100" rating-category="Mid-Tier"></
rect>
```

Resources

SVG Documentation

https://developer.mozilla.org/en-US/docs/Web/SVG/Element

https://developer.mozilla.org/en-US/docs/Web/SVG/Attribute

http://oreilly.com/catalog/svgess/chapter/ch03.html

D3.js Examples & Tutorials

https://github.com/mbostock/d3/wiki/Gallery

http://christopheviau.com/d3list/gallery.html

http://shop.oreilly.com/product/0636920026938.do
(book; free to access online)

http://alignedleft.com/tutorials
(great introduction to D3.js)

Other Visualization Tools

Raphael
http://raphaeljs.com/

Highcharts
http://www.highcharts.com/

Google Charts
https://developers.google.com/chart/

Next Class

Next Class

D3.js part 2