Introduction to D3.js

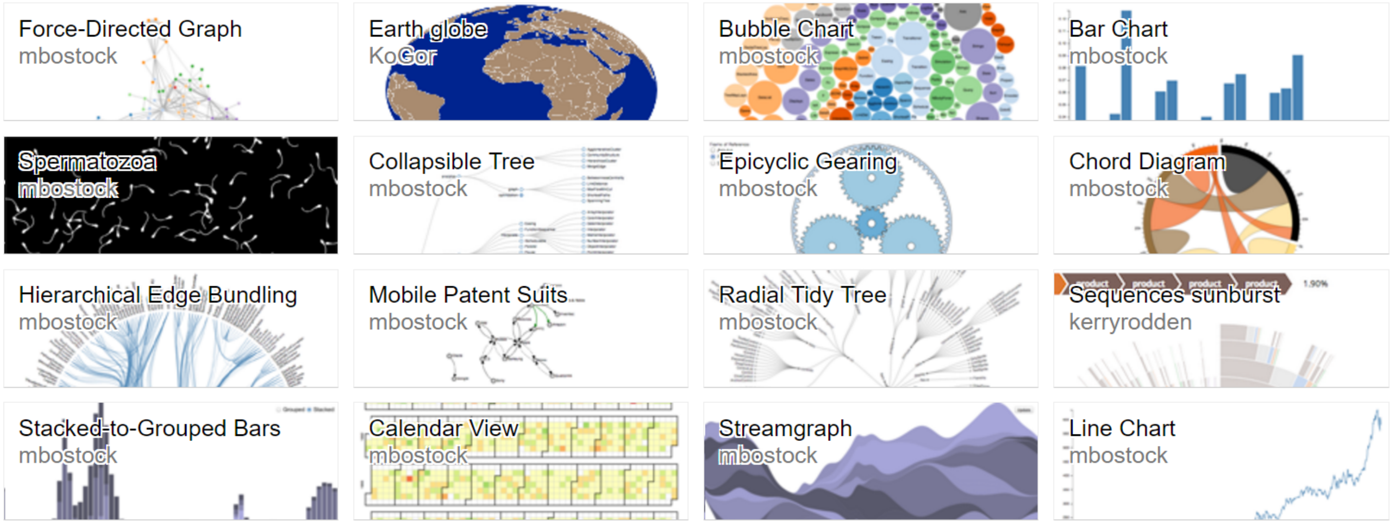
and Bar Chart in D3.js

About D3.js

- D3 represents **Data-Driven Documents**
  - A JavaScript library by Mike Bostock
  - Dynamic, interactive data visualizations.
- Nicely integrated with HTML5
  - Supported by major browsers

Data visualization with D3.js (Examples)

- https://gist.github.com/mbostock



Force-Directed Graph
mbostock

Earth globe
KoGor

Bubble Chart
mbostock

Bar Chart
mbostock

Spermatozoa
mbostock

Collapsible Tree
mbostock

Epicyclic Gearing
mbostock

Chord Diagram
mbostock

Hierarchical Edge Bundling
mbostock

Mobile Patent Suits
mbostock

Radial Tidy Tree
mbostock

Sequences sunburst
kerryrodden

Stacked-to-Grouped Bars
mbostock

Calendar View
mbostock

Streamgraph
mbostock

Line Chart
mbostock

Our Goals Today

- To create some bar charts
  - A simple HTML bar chart
  - A D3-enabled bar chart like this https://gist.github.com/mbostock/7341714
  - Adding user interaction in the bar chart

First, An HTML Bar Chart

- https://gist.github.com/mbostock/7331260

- The key element: SVG
  - Scalable Vector Graphics: graphics for the web

- CANVAS is another HTML element for graphics

- CANVAS vs SVG
  - CANVAS: weak interactivity, strong for large datasets
  - SVG: strong interactivity, weak in handling large datasets

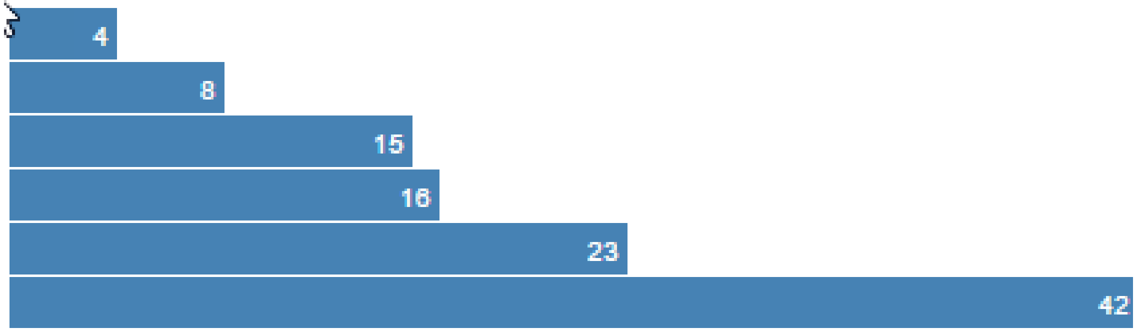
Codes of the First Example

- Two components
  - Style sheet to define the appearance of the bar chart
    - Color, font, etc.

```
<style>

.chart rect {
    fill: steelblue;
}

.chart text {
    fill: white;
    font: 10px sans-serif;
    text-anchor: end;
}

</style>
```

  - SVG to draw individual boxes and text captions.

```
<svg class="chart" width="420" height="120">
 <g transform="translate(0,0)">
   <rect width="40" height="19"></rect>
   <text x="37" y="9.5" dy=".35em">4</text>
 </g>
 <g transform="translate(0,20)">
   <rect width="80" height="19"></rect>
   <text x="77" y="9.5" dy=".35em">8</text>
 </g>
.
   .
</svg>
```



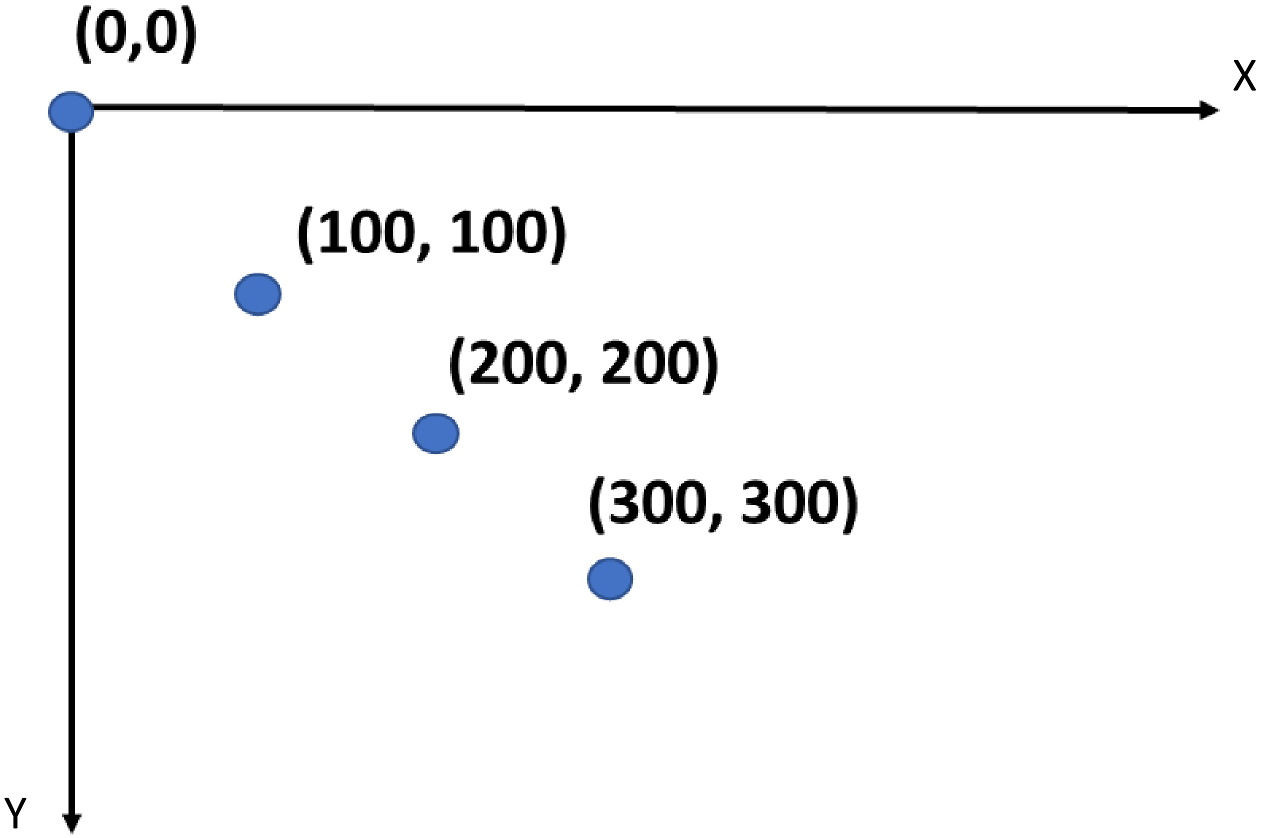
SVG components

- circle, rect, line, polygon, text, path

- g
  - Grouping a set of element
    - To be manipulated together: moving around, scaling up/down, etc.

```
<g transform="translate(0,20)">
  <rect width="80" height="19"></rect>
  <text x="77" y="9.5" dy=".35em">8</text>
 </g>
```

Coordinate System (SVG)



Exercise 1

- Create a new file named barchar1.html in your web space
- Copy the codes from the page into your file:
  - https://gist.github.com/mbostock/7331260
- Check the result.

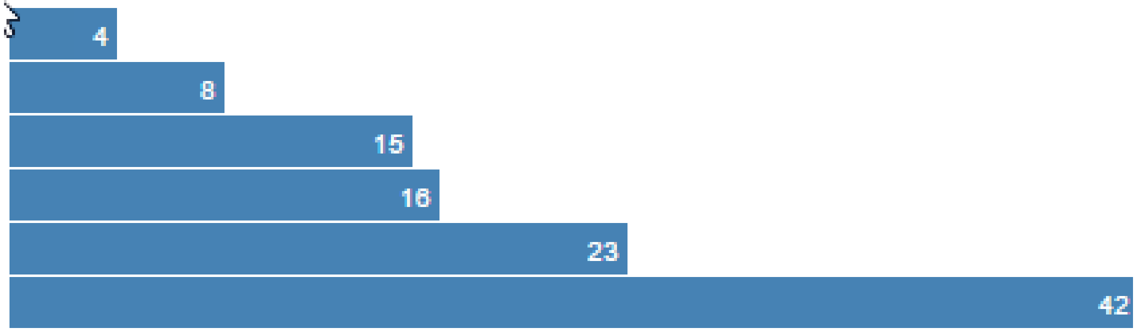
- Change the color of rectangles to another color
  - Color names: http://www.w3schools.com/colors/colors_names.asp

What If We Want to Make the Chart Bigger?

Say Double the Height of Each Bar?

How to Do That?

```
<svg class="chart" width="420" height="120">
 <g transform="translate(0,0)">
  <rect width="40" height="19"></rect>
  <text x="37" y="9.5" dy=".35em">4</text>
 </g>
 <g transform="translate(0,20)">
  <rect width="80" height="19"></rect>
  <text x="77" y="9.5" dy=".35em">8</text>
 </g>
.
  .
</svg>
```



Any Better Solution?

D3 Approach to a Bar Chart

- Check this web site:
  - https://gist.github.com/mbostock/7331275
- Three parts: style sheet, SVG, and script

- The style sheet part is the same
- The SVG part: empty (left for scripts to handle)

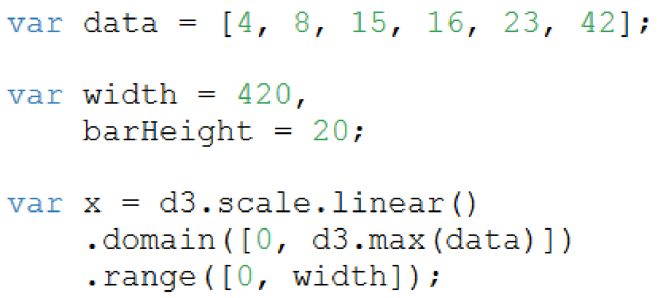
```
<svg class="chart"></svg>
```

JavaScript Part

```
var data = [4, 8, 15, 16, 23, 42];               //Define the data array used for bar chart

var width = 420,
    barHeight = 20;                              //Variables used to define the SVG size

var x = d3.scale.linear()                        //Scale data value based on SVG size
    .domain([0, d3.max(data)])
    .range([0, width]);
                                                 //Select the SVG and define its size
var chart = d3.select(".chart")
    .attr("width", width)
    .attr("height", barHeight * data.length);

var bar = chart.selectAll("g")                   //A bar is a group. Select individual groups
    .data(data)                                  //Bind data with bar
  .enter().append("g")
    .attr("transform", function(d, i) { return "translate(0," + i * barHeight + ")"; })
                                                //Define the translation of each group
bar.append("rect")
    .attr("width", x)
    .attr("height", barHeight - 1);             //Add rect and text

bar.append("text")
    .attr("x", function(d) { return x(d) - 3; })
    .attr("y", barHeight / 2)
    .attr("dy", ".35em")
    .text(function(d) { return d; });
```

Scale

- Scale functions in D3.js are used to map an input domain to an output range.
  - Linear scale function is commonly used for continuous values
    - d3.scale.linear() for D3 3.X
- domain(): input
- range(): output.



```
var data = [4, 8, 15, 16, 23, 42];

var width = 420,
    barHeight = 20;

var x = d3.scale.linear()
    .domain([0, d3.max(data)])
    .range([0, width]);
```

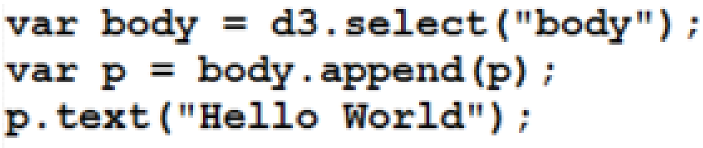
Chaining Methods

```
var x = d3.scale.linear()
    .domain([0, d3.max(data)])
    .range([0, width]);
```

=

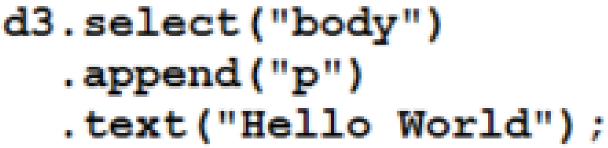
```
var x = d3.scale.linear();
    x.domain([0, d3.max(data)]);
    x.range([0, width]);
```

- Perform multiple actions with a single line of code
- Add a new paragraph with text (Hello World)



```
var body = d3.select("body");
var p = body.append(p);
p.text("Hello World");
```

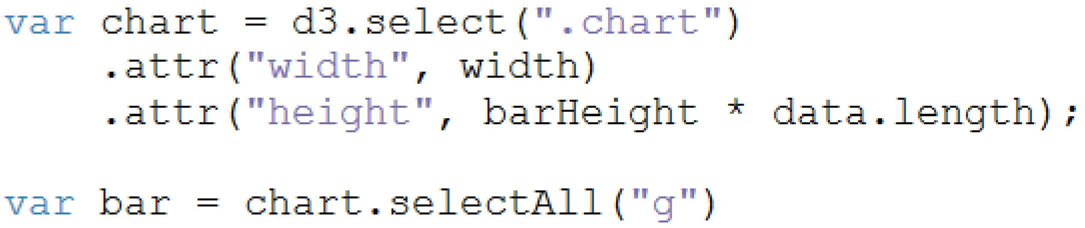
Which can be written as:



```
d3.select("body")
   .append("p")
   .text("Hello World");
```

Selections

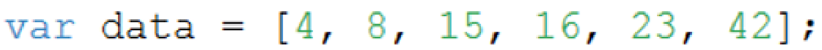
- Selecting target element(s)
  - Add/Remove element
  - Modify attributes and CSS
- Two methods for selection: d3.select() and d3.selectAll()
  - d3.select("p"): select a <p> element
  - d3.selectAll("p"): select all <p> elements
  - d3.select("#id"): select an element by id.
  - d3.select(".class"): select an element by class name.



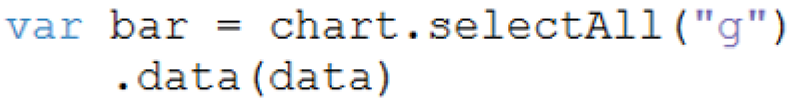
```
var chart = d3.select(".chart")
    .attr("width", width)
    .attr("height", barHeight * data.length);

var bar = chart.selectAll("g")
```

Binding Data

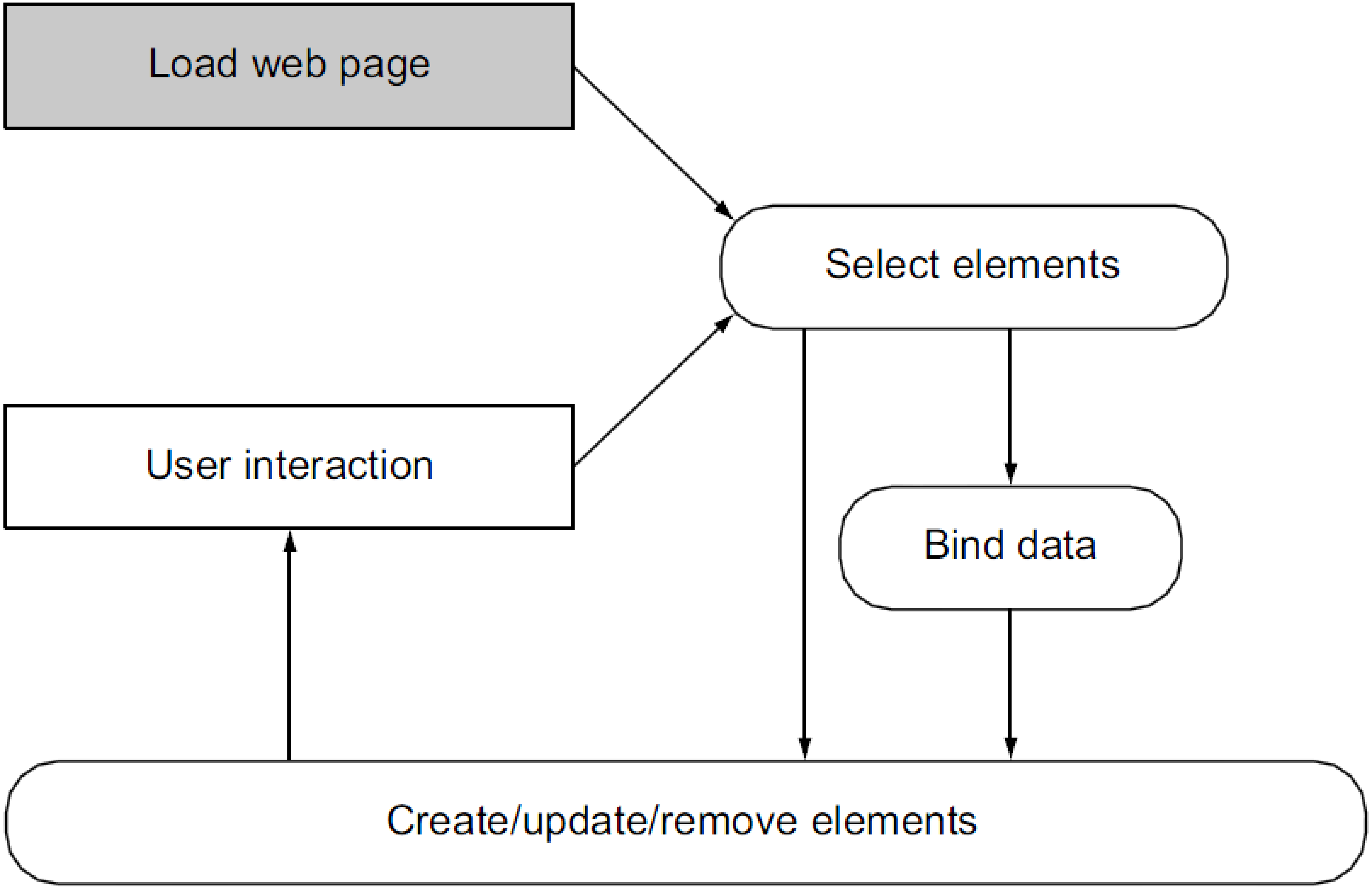
- D3 is about selecting and binding.
  - Selection: picking up HTML element(s)
  - Binding: associating data with selected element(s)



```
var data = [4, 8, 15, 16, 23, 42];
```



```
var bar = chart.selectAll("g")
    .data(data)
```



Load web page

Select elements

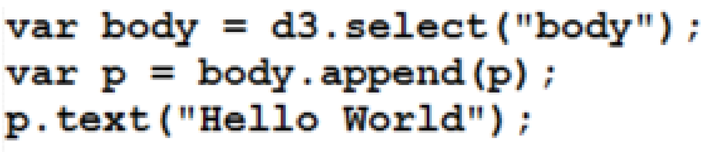
User interaction

Bind data

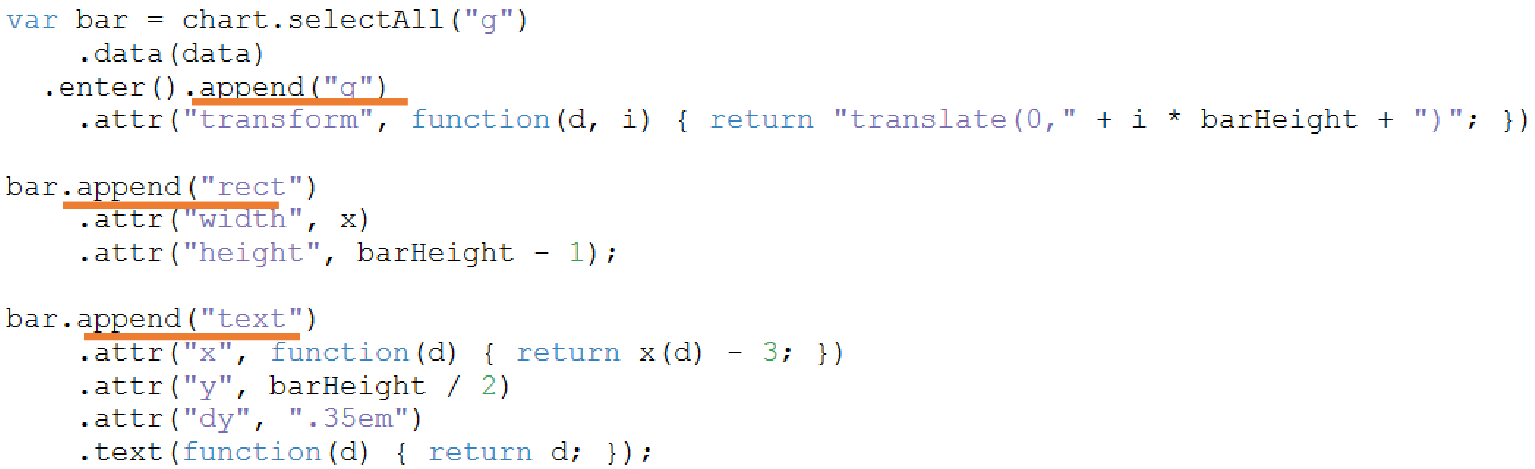
Create/update/remove elements

Add New Elements

- Add new element with D3 selector
- append() function can be used to add an new element.
- Add a new paragraph with text (Hello World)



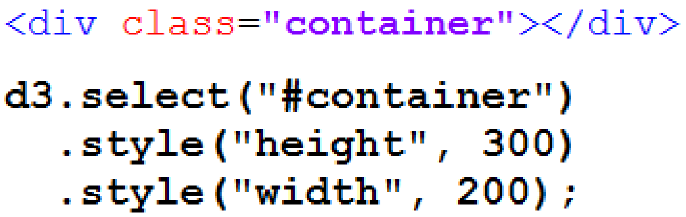
```
var body = d3.select("body");
var p = body.append(p);
p.text("Hello World");
```



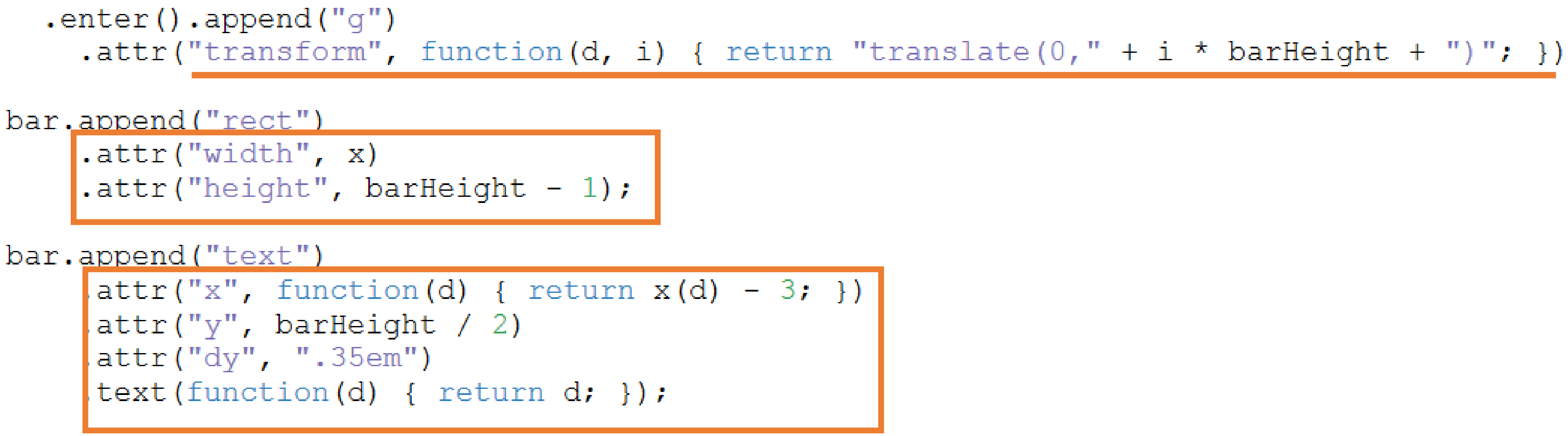
```
var bar = chart.selectAll("g")
    .data(data)
  .enter().append("g")
    .attr("transform", function(d, i) { return "translate(0," + i * barHeight + ")"; })

bar.append("rect")
    .attr("width", x)
    .attr("height", barHeight - 1);

bar.append("text")
    .attr("x", function(d) { return x(d) - 3; })
    .attr("y", barHeight / 2)
    .attr("dy", ".35em")
    .text(function(d) { return d; });
```

Dynamic Properties

- Attributes and styles of elements can be set and/or modified dynamically with D3.js
- Set the width and height of existing div element



```
<div class="container"></div>

d3.select("#container")
   .style("height", 300)
   .style("width", 200);
```



```
  .enter().append("g")
    .attr("transform", function(d, i) { return "translate(0," + i * barHeight + ")"; })

bar.append("rect")
    .attr("width", x)
    .attr("height", barHeight - 1);

bar.append("text")
    .attr("x", function(d) { return x(d) - 3; })
    .attr("y", barHeight / 2)
    .attr("dy", ".35em")
    .text(function(d) { return d; });
```

Exercise 2

- Create another HTML file, barchart2.html
- Copy the codes from:
  - https://gist.github.com/mbostock/7331275
- Check the result
- Modify the code to double the height of each bar.

A Few Things About D3

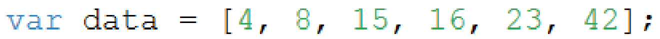
- You must call d3 library to get it work.

  `<script src="//d3js.org/d3.v3.min.js" ></script>`
- The library can be local.
  - You can download the library and put it in a directory.

    `<script src="d3.min.js"></script>`
  - This is very useful when you are offline!
- Be careful about versions.
  - Some significant difference between d3.v3 and d3.v4.
  - We use v3 here.
- Download the library here:
  - https://github.com/d3/d3/releases/tag/v3.5.17

Still One Problem Here

- Look at this line of code



```
var data = [4, 8, 15, 16, 23, 42];
```

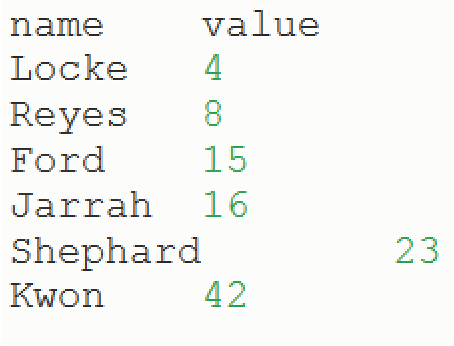
- Data are manually assigned.

Can We Make the Codes More Flexible?

Yes, We Can!

- https://gist.github.com/mbostock/7341714
- Key difference from the previous example:

```
d3.tsv("data.tsv", type, function(error, data) {
    .
    .
}


function type(d) {
    d.value = +d.value; // coerce to number
    return d;
}
```



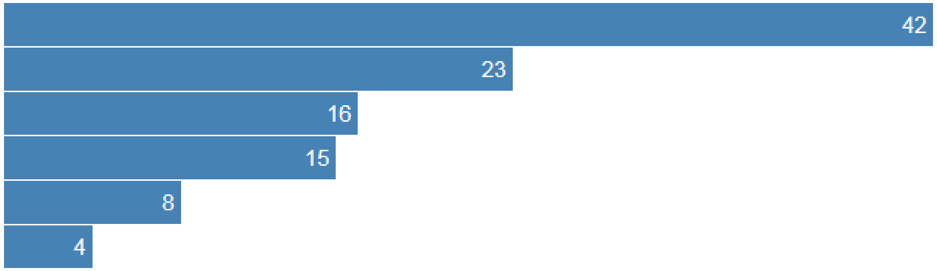
```
name      value
Locke     4
Reyes     8
Ford      15
Jarrah    16
Shephard          23
Kwon      42
```

Loop Becomes Simpler in D3

```
var bar = chart.selectAll("g")
    .data(data)
    .enter().append("g")
    .attr("transform", function(d, i) { return "translate(0," + i * barHeight + ")"; });

 bar.append("rect")
    .attr("width", function(d) { return x(d.value); })
    .attr("height", barHeight - 1);

 bar.append("text")
    .attr("x", function(d) { return x(d.value) - 3; })
    .attr("y", barHeight / 2)
    .attr("dy", ".35em")
    .text(function(d) { return d.value; });
```

Exercise 3

- Create an HTML file, barchart3.html
- Copy the code from
  - https://gist.github.com/mbostock/7341714
- Test the codes
- Modify the codes, not the data array, to reverse the position of each bar (see the figure)



- Modify the codes to make each bar a unique color.

Resource for Learning D3.js

- Tutorial for D3.js by Scott Murray
  - http://alignedleft.com/tutorials
- Data Visualization with D3.js linkedin.com
  - https://www.linkedin.com/learning/learning-data-visualization-with-d3-js/
- D3 API Reference
  - https://github.com/d3/d3/wiki/