

Data Visualization with D3.js



Trainer: Amir Othman



Website: www.tertiarycourses.com.my
Email: sales2@tertiaryinfotech.com



Agenda

Module 1. Get Started

- What is D3.js?
- Downloading and installing D3
- First D3 code

Module 2. Basics of D3.js

- Selecting Elements
- Controlling HTML
- Modifying Attributes
- Modifying CSS
- Data Binding



Agenda

Module 3. Basic SVG Graphics

- SVG Canvas
- Basic SVG Shapes

Module 4. Scales & Axes

- Scales
- Linear Scale
- Axis
- Parse Date
- Time Scale
- Ordinal Scale



Agenda

Module 5. Events & Transitions

- Events
- Transitions



Prerequisites

- Basic knowledge of HTML, CSS and Javascript are required.
- Text editor (Sublime Text, Atom, Vim, etc.)
- Web server (Apache, NGINX, etc.)
 - CLI
 - GUI: Fenix



Module 1

Introduction



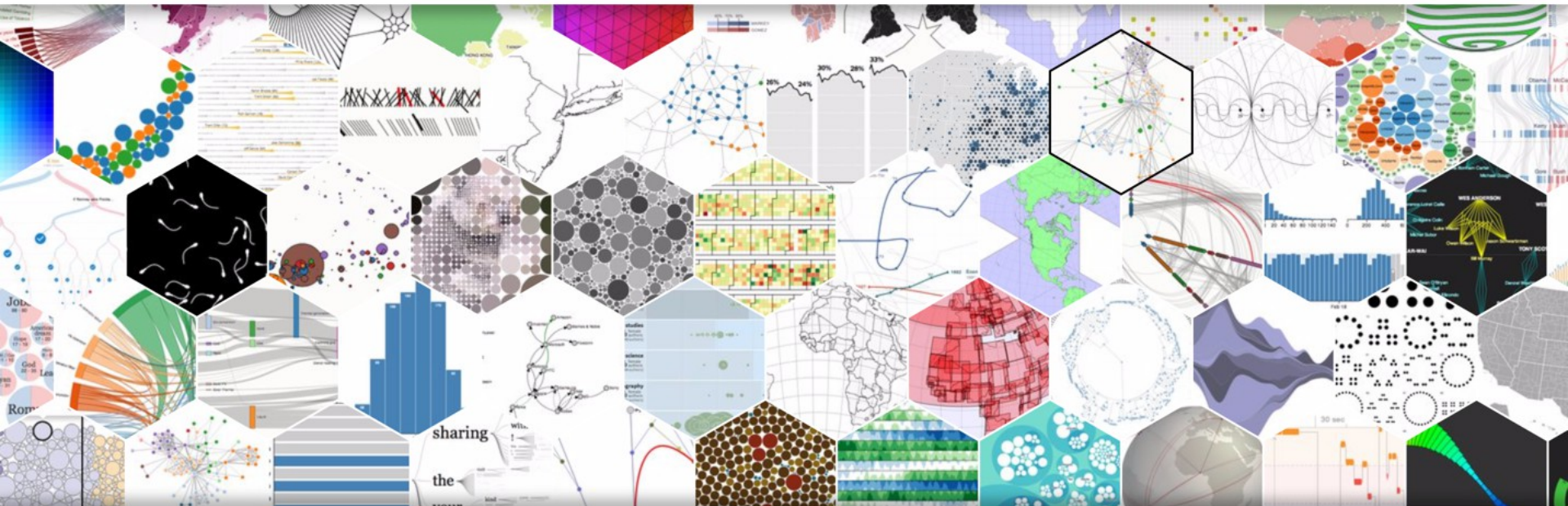
<https://d3js.org/>

<https://d3js.org/>

Fork me on GitHub



Data-Driven Documents



What is D3.js?

- D3 stands from Data-Driven Documents
- D3 is a JavaScript library
- **D3 allows you to bind data to a Document Object Model (DOM)**
- D3 allow you to apply data-driven transformations to the document.
- D3 helps you bring data to life using HTML, SVG, and CSS.

D3 Code

```
var svg =  
d3.select("body").append("svg").attr("height","100%").attr("width","100%");  
  
svg.append("rect").attr("x","100").attr("y","100").attr("width","200").attr("height","200").attr("fill","red");
```

D3 Quick-dive



Module 2

Basic of D3



Selecting Elements



Selecting Elements

```
d3.select("p");
```

```
d3.selectAll("p");
```



Chaining

```
d3.select("p")  
  .text("Today is Sunday")  
  .style("color","blue")  
  .style("background-color","yellow");
```



Selecting Descendent Elements

```
d3.select("#chart").select(".item").text("Hello");
```



Ex: Selection

Change every 4th item to "D3 Selection"

```
<li class='item'>Module 1: Introduction</li>  
<li class='item'>Module 2: Basics of D3</li>  
<li class='item'>Module 3: D3 Graphics</li>  
<li class='item'>D3 Selection</li>  
<li class='item'>Module 5: D3 Data Manipulation</li>  
....
```

Time: 2 min

Controlling HTML



Add HTML to Selection

```
d3.selectAll('.item')  
  .html('<strong>Hello</strong>')
```



Append Selection

```
d3.selectAll('.item')  
  .append("div")  
  .html('<strong>hello</strong>')
```

```
d3.selectAll('.item')  
  .append("span")  
  .html('<strong> hello</strong>')
```



Ex: Append

Append a paragraph after the bullet list

"I am learning **D3** today"

Time: 2 mins



Ex: Selection, Append, HTML

Create a list of fruits below for following items using D3 append & html

- . Apple
- . Orange
- . Pineapple
- . Durian
- . Mango
- . Peach
- . Grape

Time: 5 mins

Modifying Attributes and CSS



Add Attribute

```
d3.selectAll('.item')  
  .attr('width','200px')  
  .attr('height','200px')  
  .attr('class','highlight')
```



Add Style

```
d3.selectAll('.item')  
  .style('color','green')
```



Add Multiple Styles

```
d3.selectAll('.item:nth-child(2n)')  
  .style({  
    'background': '#268BD2',  
    'padding': '10px',  
    'margin' : '5px',  
    'color' : '#EEE8D5'  
  })
```



Add Class

```
d3.selectAll('.item')  
  .classed('highlight',true)
```



Add Multiple Classes

```
d3.selectAll('.item')  
  .classed({  
    'highlight': true,  
    'item': false,  
    'bigger': true  
  })
```



Data Binding



Data Binding

- Data is specified as an array of values
- Each value is passed as the first argument (d) to selection functions
- With the default join-by-index, the first element in the data array is passed to the first node in the selection, the second element to the second node, and so on.

Enter

Data numbers greater than element numbers.

Enter will create missing elements

```
var a = [1,4,9,16,25,36];  
d3.selectAll("p")  
  .data(a)  
  .enter().append("p")  
  .text(function(d) {return d;});
```

Ex: Enter

```
var a = ["Ally", "Belinda", "Jane", "Alfred", "Kevin"];
```

Auto create a list for the name above



Hint to Exercise

```
d3.select("body")  
  .append("ul")
```

```
d3.selectAll("li")  
  .data(a)  
  .enter().append("li")  
  .text(function(d) {return d;});
```



Exit

Data numbers less than element numbers.

Exit will remove extra elements

```
var a = [1,4,9];  
d3.selectAll("p")  
  .data(a)  
  .text(function(d) {return d;})  
  .exit().remove("p");
```



Ex: Exit

```
var a = ["Ally", "Belinda", "Jane", "Alfred", "Kevin"];
```

Auto create a list for the name above



Hint to Exercise

```
var a = [1,2,3,4,5];  
d3.selectAll("p")  
  .data(a)  
  .enter().append("p")  
  .text(function(d) {  
    return "The square of "+d+" is "+(d*d);}  
  );
```

Index Input to Function

```
d3.selectAll("li")  
  .text(function(d,i) {  
    return "The square of "+i+" is "+(i*i);}  
  );
```



Hint to Exercise

```
var a = [1,4,9,16,25,36];  
d3.selectAll("p")  
  .data(a)  
  .enter().append("p")  
  .text(function(d,i) {  
    return "The square of " + (i+1) + " is " + d;  
  })
```

JSON Data

```
var person = [  
  {name:"Ally",height:160},  
  {name:"Belinda",height:155},  
  {name:"Jane",height:170},  
  {name:"Alfred",height:170},  
  {name:"Kevin",height:175},  
  {name:"John",height:140}  
];
```



Binding Data to JSON Data

```
d3.selectAll("p")  
  .data(person)  
  .enter().append("p")  
  .text(function(d) {  
    return d.name + " height is " + d.height;  
  });
```



Module 3

Basic SVG Graphics



What is SVG

SVG stands for Scalable Vector Graphics

- Add vector graphics support to HTML5
- SVG is HTML Like.
- Can be styled through CSS



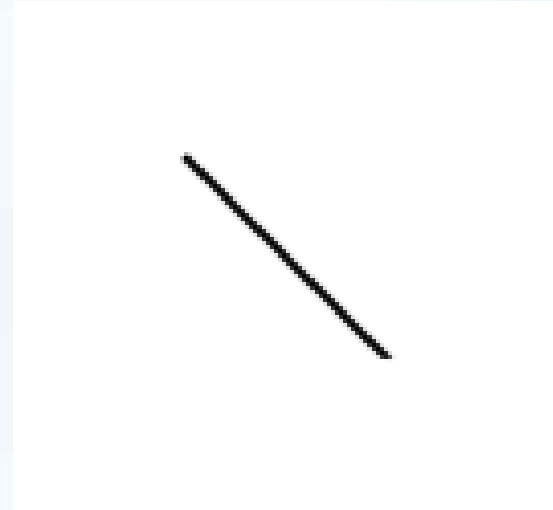
SVG Canvas

```
var svg = d3.select("body")  
    .append("svg")  
    .attr("height", "100%")  
    .attr("width", "100%");
```



Line

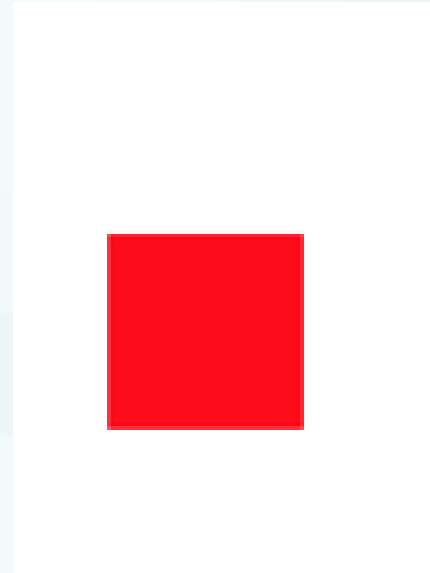
```
svg.append("line")  
  .attr("x1","0")  
  .attr("y1","200")  
  .attr("x2","400")  
  .attr("y2","0")  
  .attr("stroke","black")  
  .attr("stroke-width","3");
```



How would one draw a visualization with line?

Rectangle

```
svg.append("rect")  
  .attr("x","100")  
  .attr("y","100")  
  .attr("height","100")  
  .attr("width","50")  
  .attr("fill","red");
```



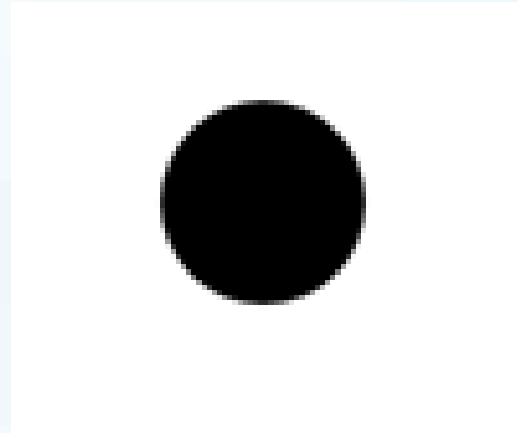
How would one draw a visualization with rect?

Hint to Exercise

```
svg.selectAll("rect")  
  .data(a)  
  .enter().append("rect")  
  .attr("x",function(d,i){return 30+i*50;})  
  .attr("y",function(d,i){return 50-d})  
  .attr("width","30")  
  .attr("height",function(d,i){return d;})  
  .attr("fill","red");
```

Circle

```
svg.append("circle")  
  .attr("cx","100")  
  .attr("cy","100")  
  .attr("r","20")  
  .attr("fill","black");
```



Ex : Circle

Create 3 circles with radius with radius 5, 11 and 18



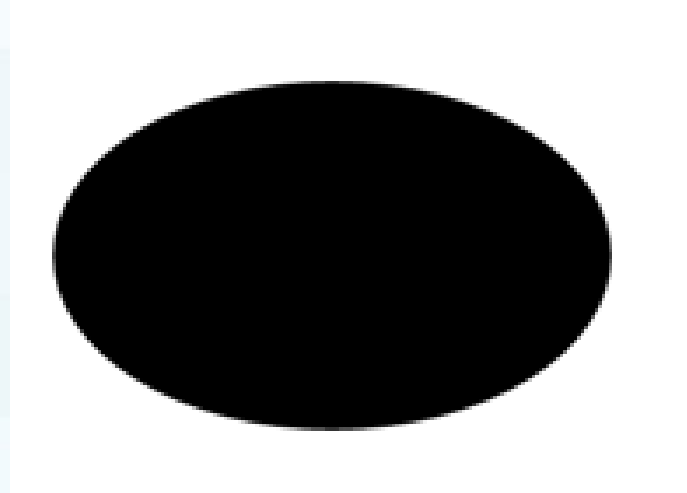
Hint to Exercise

```
svg.selectAll("circle")  
  .data(a)  
  .enter().append("circle")  
  .attr("cx",function(d,i){return 50+i*50;})  
  .attr("cy","50")  
  .attr("r",function(d,i){return d;})  
  .attr("fill","black");
```



Ellipse

```
svg.append("ellipse")  
  .attr("cx","100")  
  .attr("cy","100")  
  .attr("rx","80")  
  .attr("ry","50")  
  .attr("fill","black");
```



Ex : Ellipse

Create 3 ellipse with x radius with radius 5, 11 and 18 and fixed y radius of 5



Hint to Exercise

```
svg.selectAll("ellipse")  
  .data(a)  
  .enter().append("ellipse")  
  .attr("cx",function(d,i){return 50+i*100;})  
  .attr("cy","50")  
  .attr("rx",function(d,i){return d;})  
  .attr("ry",5)  
  .attr("fill","black");
```



Text

```
svg.append("text")  
  .attr("x","100")  
  .attr("y","100")  
  .attr("font-size","20")  
  .attr("color","red")  
  .text("Hello World");
```

Hello World



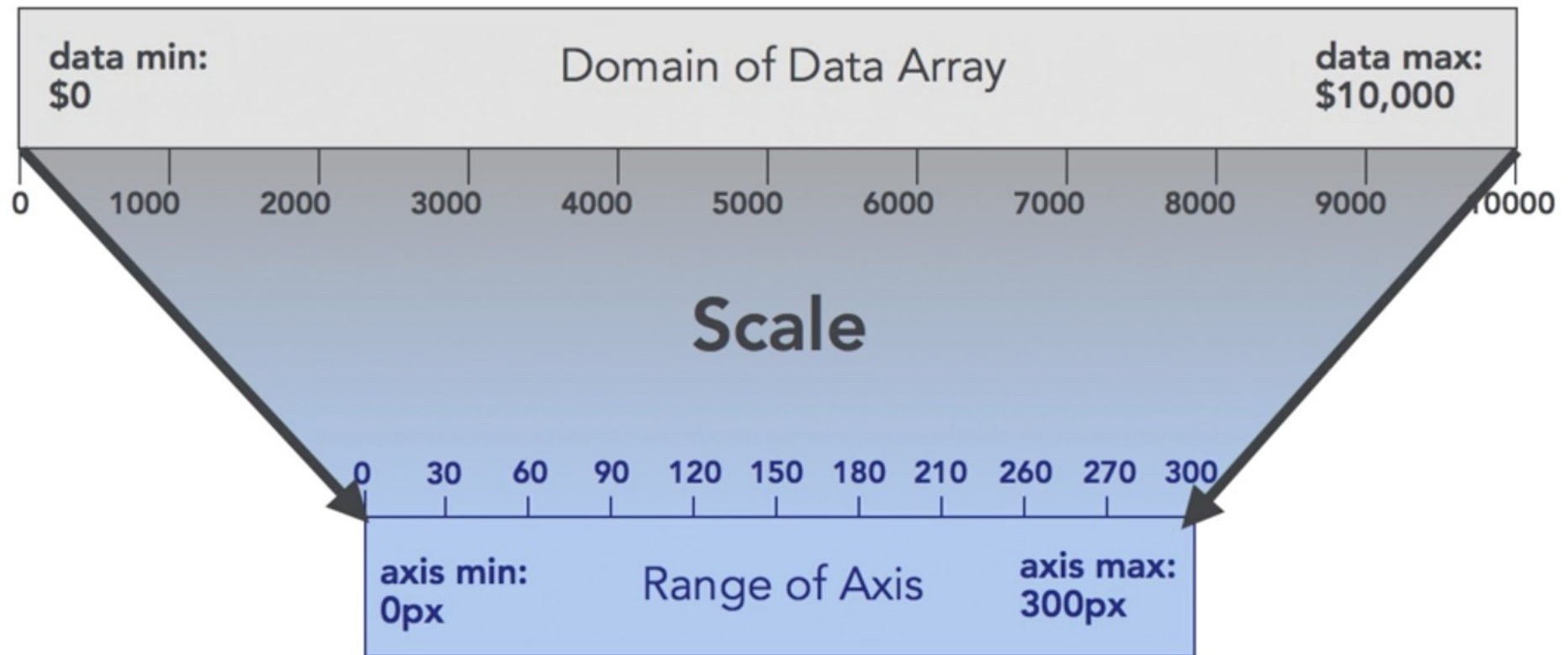
Module 4

Scales & Axes



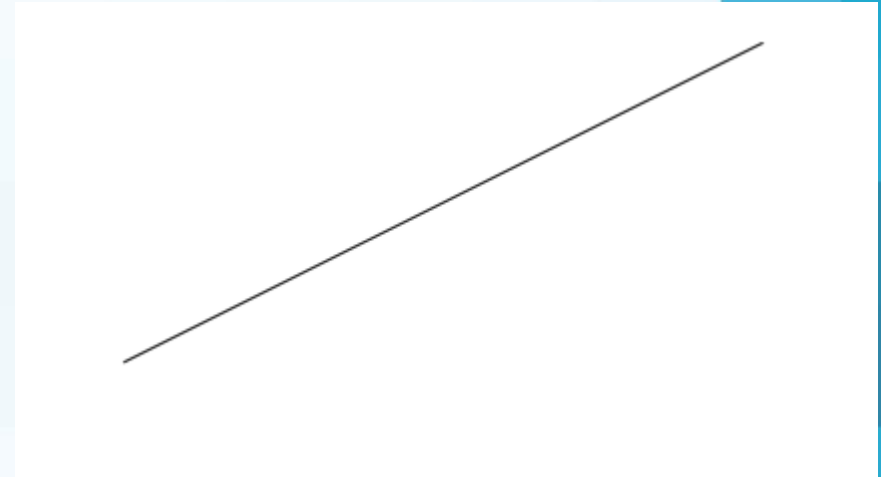
Scales

Scales are functions that map from an input domain to an output range.



Linear Scale

```
var width = 400;  
var height = 200;  
var xS = d3.scaleLinear()  
    .domain([0,5])  
    .range([0,width]);  
var yS = d3.scaleLinear()  
    .domain([0,10])  
    .range([height,0]);
```



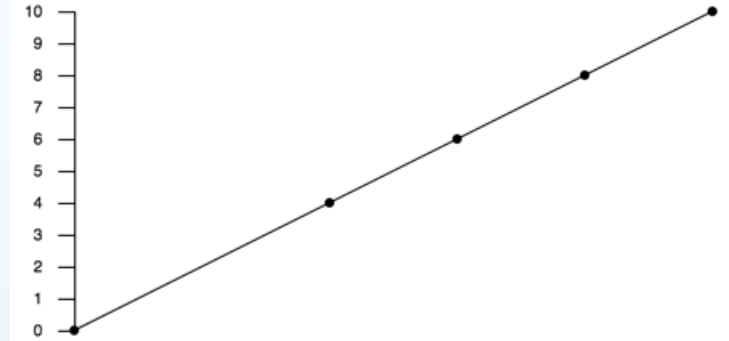
Hint to Exercise

```
svg.selectAll("circle")  
  .data(pts)  
  .enter().append("circle")  
  .attr("cx",function(d,i){return xS(d.x);})  
  .attr("cy",function(d,i){return yS(d.y);})  
  .attr("r","3");
```



Axis

```
var yAxis = d3.axisLeft(yS)  
    .ticks(10)  
    .tickPadding(10)  
    .tickSize(10);
```

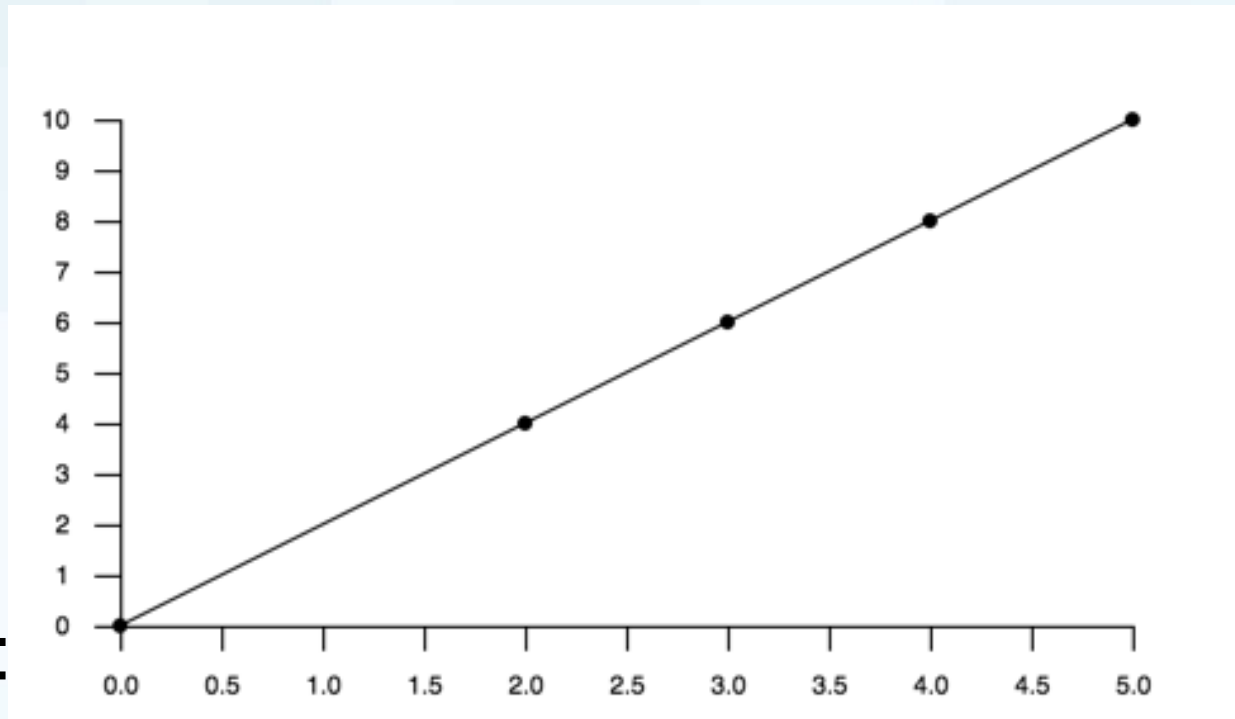


```
var chart = svg.append("g")  
    .attr("transform", "translate(50,50)");
```

```
chart.append("g")  
    .call(yAxis);
```

Ex : Axis

Add x-axis to the straight line curve



Time:

Hint to Exercise

```
var xAxis = d3.axisBottom(xS)
    .tickPadding(10)
    .tickSize(10)
    .ticks(10);

chart.append("g")
    .attr("transform", "translate(0,200)")
    .call(xAxis);
```

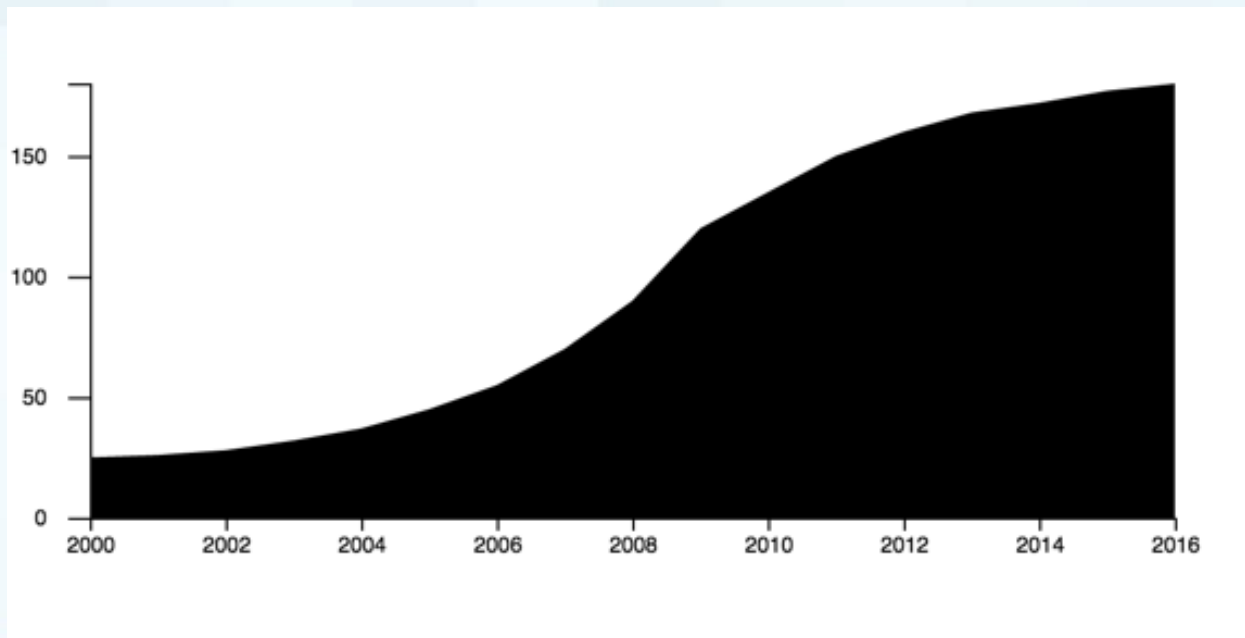
Parse Date

```
var parseDate = d3.timeParse("%Y");
```



Time Scales

```
var x = d3.scaleTime()  
  .domain([parseDate('2000'),parseDate('2016')])  
  .range([0,width]);
```



Ordinal Scales

```
var dataArray = [5,11,18];  
var dataDays = ['Mon','Wed','Fri'];
```

```
var x = d3.scaleBand()  
        .domain(dataDays)  
        .range([0,170]);
```



Ordinal Scales Axis

```
var xAxis = d3.axisBottom(x);
```

```
svg.selectAll("rect")  
  .data(dataArray)  
  .enter().append("rect")  
  .attr("height",function(d,i){ return d*15; })  
  .attr("width","50")  
  .attr("x",function(d,i){ return 60*i; })  
  .attr("y",function(d,i){ return 300-(d*15); });
```

Module 5

Events & Animation



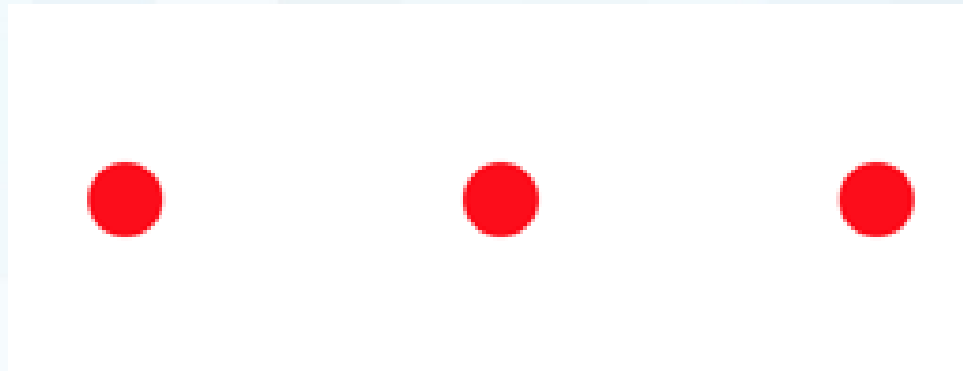
On Mouseover and Mouseout

```
.on("mouseover", function() {  
    d3.select(this)  
    .style("opacity", "0.5")  
    .attr("fill", "blue");  
})  
.on("mouseout", function() {  
    d3.select(this)  
    .style("opacity", "1")  
    .attr("fill", "red");  
})
```



Ex : Mouse Over and Out

Draw 3 red circles, when mouse over change to blue and mouse out change back to red.



Time: 10 mins



Hint to Exercise

```
svg.selectAll('circle')  
  .on("mouseover", function(){  
    d3.select(this)  
    .attr("fill", "blue");  
  })  
  .on("mouseout", function(){  
    d3.select(this)  
    .attr("fill", "red");  
  });
```



On Click

```
svg.append("rect")  
  .attr("fill", "red")  
  .on("click", function() {  
    d3.select(this)  
      .style("opacity", "0.5")  
      .attr("fill", "blue");  
  })
```



Transition

```
svg.append("rect")  
  .attr("x","100")  
  .attr("y","100")  
  .attr("height","100")  
  .attr("width","100")  
  .style("fill","red")  
  .transition().duration(2000).delay(1000)  
  .style("fill","blue");
```

Ex : Transition

Create a transition to move the rect 300px to the right



Time: 10 mins

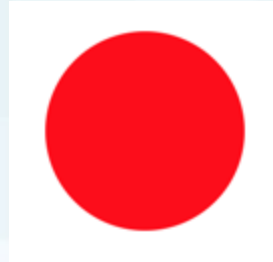


Hint to Exercise

```
svg.append("rect")  
  .attr("x","100")  
  .attr("y","100")  
  .attr("height","100")  
  .attr("width","100")  
  .attr("fill","red")  
  .transition().duration(2000).delay(1000)  
  .attr("transform","translate(300,0)");
```

Ex: Transitions

Add an animation to expand the circle radius from 10 to t_0



Time: 5 mins



Hint to Exercise

```
function animate(){  
  d3.select(this)  
    .transition()  
    .duration(1000)  
    .attr("r", 50)  
};
```



Resources

- Examples - <https://bl.ocks.org>
- D3 in Depth - d3indepth.com
- D3 Tips and Tricks - <https://leanpub.com/d3-t-and-t-v4/read>
- D3 API Reference - <https://github.com/d3/d3/blob/master/API.md>



Summary Parting Message



**Practice
Makes
Perfect**

Q & A Feedback

<https://goo.gl/EDezXH>



Thank You!

Amir Othman

011-369 789 24

othman.amir@gmail.com