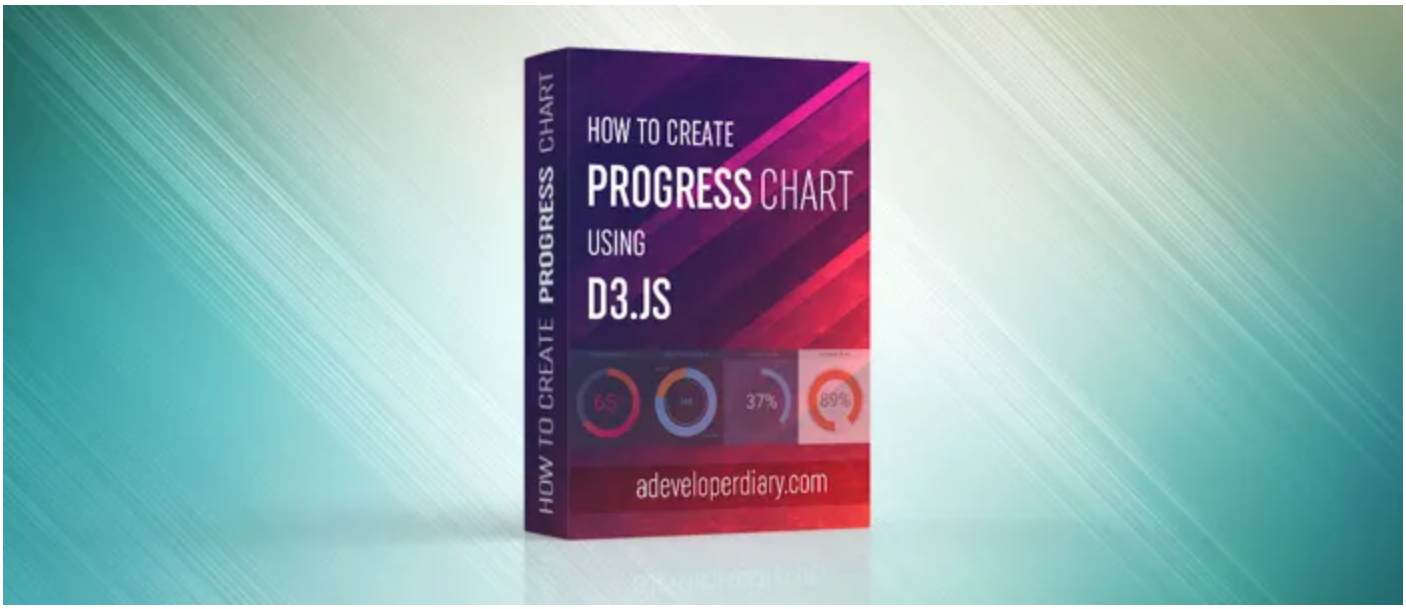
A Developer Diary

{about:"code learn and share"}

January 11, 2016 By Abhisek Jana — 3 Comments (Edit)

How to create Progress chart using d3.js



We have learnt how to create donut chart using d3.js and we have also animated the chart. In this lesson we will now learn How to create Progress chart using d3.js. We will create four different types of progress chart.

If you need to understand how to create basic donut chart using d3 then you can read my earlier article here.

Basically progress chart consists of two charts,one for the container and another for displaying the progress. We will also use svg gradient to add the extra visual element.

PROGRESS CHART 1:

The `createGradient()` function is used to create the `linearGradient`. Rest of it is straight forward, here is the Java Script Code:

```
var createGradient=function(svg,id,color1,color2){

        var defs = svg.append("svg:defs");
```

```
        var red_gradient =
defs.append("svg:linearGradient")
        .attr("id", id)
        .attr("x1", "0%")
        .attr("y1", "0%")
        .attr("x2", "50%")
        .attr("y2", "100%")
        .attr("spreadMethod", "pad");

        red_gradient.append("svg:stop")
        .attr("offset", "50%")
        .attr("stop-color", color1)
        .attr("stop-opacity", 1);

        red_gradient.append("svg:stop")
        .attr("offset", "100%")
        .attr("stop-color", color2)
        .attr("stop-opacity", 1);
};

var percent = 65;

var ratio=percent/100;

var pie=d3.layout.pie()
    .value(function(d){return d})
    .sort(null);

var w=300,h=300;

var outerRadius=(w/2)-10;
var innerRadius=110;
```

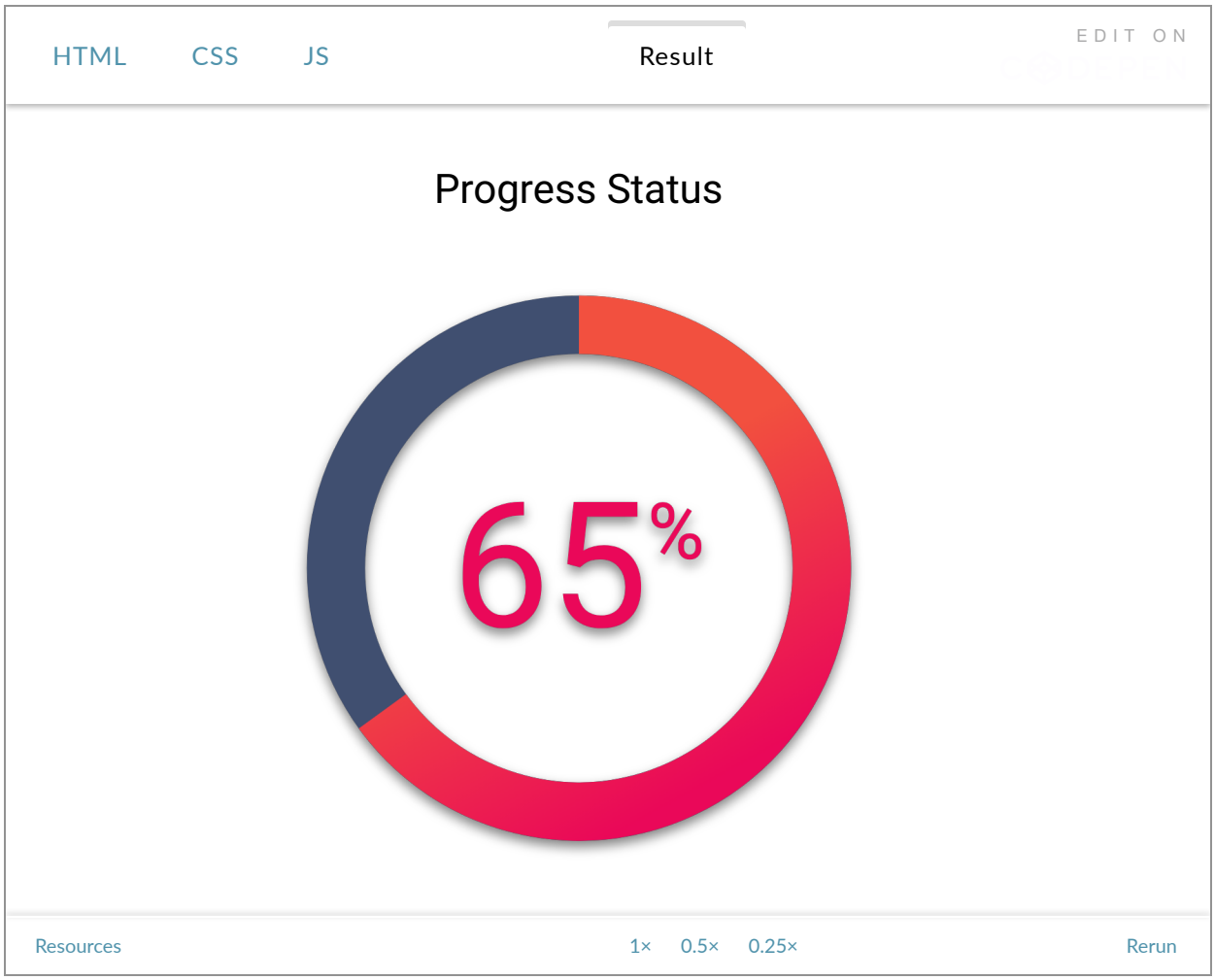
```
var color = ['#f2503f','#ea0859','#404F70'];

var svg=d3.select("#chart")
    .append("svg")
    .attr({
        width:w,
        height:h,
        class:'shadow'
    }).append('g')
    .attr({
        transform:'translate('+w/2+','+h/2+')'
    });

createGradient(svg,'gradient',color[0],color[1]);

var arc=d3.svg.arc()
    .innerRadius(innerRadius)
    .outerRadius(outerRadius)
    .startAngle(0)
    .endAngle(2*Math.PI);

var arcLine=d3.svg.arc()
    .innerRadius(innerRadius)
    .outerRadius(outerRadius)
    .startAngle(0);


var pathBackground=svg.append('path')

    .attr({
        d:arc
    })
    .style({
```

```
            fill:color[2]
        });


    var pathChart=svg.append('path')
        .datum({endAngle:0})
        .attr({
            d:arcLine
        })
        .style({
            fill:'url(#gradient)'
        });

    var middleCount=svg.append('text')
        .text(function(d){
            return d;
        })

        .attr({
            class:'middleText',
            'text-anchor':'middle',
            dy:30,
            dx:-15
        })
        .style({
            fill:color[1],
            'font-size':'90px'

        });
    svg.append('text')
        .text('%')
        .attr({
            class:'percent',
```

```
            'text-anchor':'middle',
            dx:50,
            dy:-5

        })
        .style({
            fill:color[1],
            'font-size':'40px'

        });


var arcTween=function(transition, newAngle) {
        transition.attrTween("d", function (d) {
            var interpolate = d3.interpolate(d.endAngle,
newAngle);
            var interpolateCount = d3.interpolate(0,
percent);
            return function (t) {
                d.endAngle = interpolate(t);

middleCount.text(Math.floor(interpolateCount(t)));
                return arcLine(d);
            };
        });
};


var animate=function(){
        pathChart.transition()
                .duration(750)
                .ease('cubic')
                .call(arcTween,((2*Math.PI))*ratio);
```

```
};

setTimeout(animate,0);
```

Click on `Run Pen` to see the full demo.



The StyleSheet and HTML code is same as the previous examples I have given. You can find those here.

PROGRESS CHART 2:

In this Progress chart we will highlight the major section using a progress chart. Here is the code:

```
    var dataset = [
    { name: 'Success', count: 546 },
    { name: 'Error', count: 155 }
];

var total=0;

dataset.forEach(function(d){
    total+= d.count;
});

var ratio=dataset[0].count/total;

var pie=d3.layout.pie()
        .value(function(d){return d.count})
        .sort(null);

var w=300,h=300;

var outerRadius=(w/2)-10;
var innerRadius=100;

var color = d3.scale.ordinal()
 .range(['#67BAF5','#F17F4D']);

var arc=d3.svg.arc()
        .innerRadius(innerRadius)
        .outerRadius(outerRadius);
```

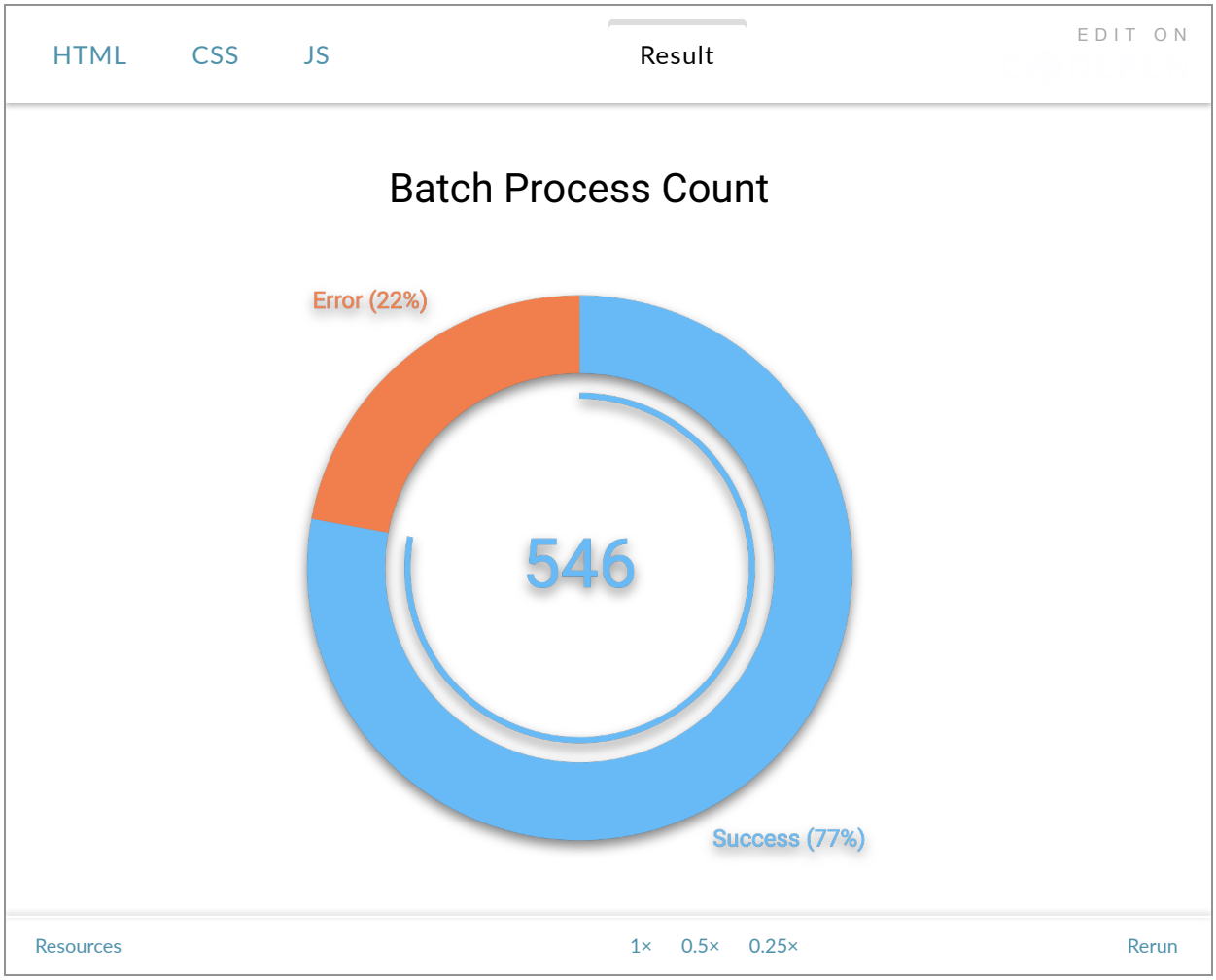
```
var arcLine=d3.svg.arc()
        .innerRadius(innerRadius-13)
        .outerRadius(innerRadius-10)
        .startAngle(0);

var svg=d3.select("#chart")
        .append("svg")
        .attr({
            width:w,
            height:h,
            class:'shadow'
        }).append('g')
        .attr({
            transform:'translate('+w/2+','+h/2+')'
        });
var path=svg.selectAll('path')
        .data(pie(dataset))
        .enter()
        .append('path')
        .attr({
            d:arc,
            fill:function(d,i){
                return color(d.data.name);
            }
        });

var pathLine=svg.append('path')
        .datum({endAngle:0})
        .attr({
            d:arcLine
        })
        .style({
            fill:color('Success')
```

```
            });

    var text=svg.selectAll('.legend')
            .data(pie(dataset))
            .enter()
            .append("text")
            .attr('class','legend')
            .attr("transform", function (d) {
                var c=arc.centroid(d);
                return "translate(" +(c[0] *1.4)+','+
(c[1]*1.5) + ")";
            })
            .attr("dy", ".4em")
            .attr("text-anchor", "middle")
            .text(function(d){
                return d.data.name+'
('+Math.floor((d.data.count/total)*100)+'%)';
            })
            .style({
                fill:function(d){
                    return color(d.data.name);
                },
                'font-size':'12px'
            });

    var middleCount=svg.append('text')
            .datum(0)
            .text(function(d){
                return d;
            })

            .attr({
                class:'middleText',
```

```
                'text-anchor':'middle',
                dy:10
            })
            .style({
                fill:color('Success'),
                'font-size':'35px'

            });

    var arcTween=function(transition, newAngle) {
        transition.attrTween("d", function (d) {
            var interpolate = d3.interpolate(d.endAngle,
newAngle);
            var interpolateCount = d3.interpolate(0,
dataset[0].count);
            return function (t) {
                d.endAngle = interpolate(t);

middleCount.text(Math.floor(interpolateCount(t)));
                return arcLine(d);
            };
        });
    };


    var animate=function(){
        pathLine.transition()
                .duration(750)
                .call(arcTween,((2*Math.PI))*ratio);



    };
```

```
setTimeout(animate,100);
```

Click on `Run Pen` to see the full demo.



PROGRESS CHART 3:

Here is another progress chart. This time we we animate the chart in regular interval.

```
var percent = 65;

var ratio=percent/100;
```

```
var pie=d3.layout.pie()
        .value(function(d){return d})
        .sort(null);

var w=300,h=300;

var outerRadius=(w/2)-10;
var innerRadius=110;


var color = ['#404F70','#67BAF5','#2d384d'];

var arc=d3.svg.arc()
        .innerRadius(innerRadius)
        .outerRadius(outerRadius)
        .startAngle(0)
        .endAngle(2*Math.PI);


var arcLine=d3.svg.arc()
        .innerRadius(innerRadius)
        .outerRadius(outerRadius)
        .cornerRadius(20)
        .startAngle(-0.05);

var svg=d3.select("#chart")
        .append("svg")
        .attr({
            width:w,
            height:h,
            class:'shadow'
        }).append('g')
        .attr({
```

```
            transform:'translate('+w/2+','+h/2+')'
        });

var defs = svg.append("svg:defs");

var inset_shadow = defs.append("svg:filter")
        .attr("id", "inset-shadow");

inset_shadow.append("svg:feOffset")
        .attr({
            dx:0,
            dy:0
        });

inset_shadow.append("svg:feGaussianBlur")
        .attr({
            stdDeviation:8,
            result:'offset-blur'
        });

inset_shadow.append("svg:feComposite")
        .attr({
            operator:'out',
            in:'SourceGraphic',
            in2:'offset-blur',
            result:'inverse'
        });

inset_shadow.append("svg:feFlood")
        .attr({
            'flood-color':'black',
            'flood-opacity':1,
            result:'color'
```

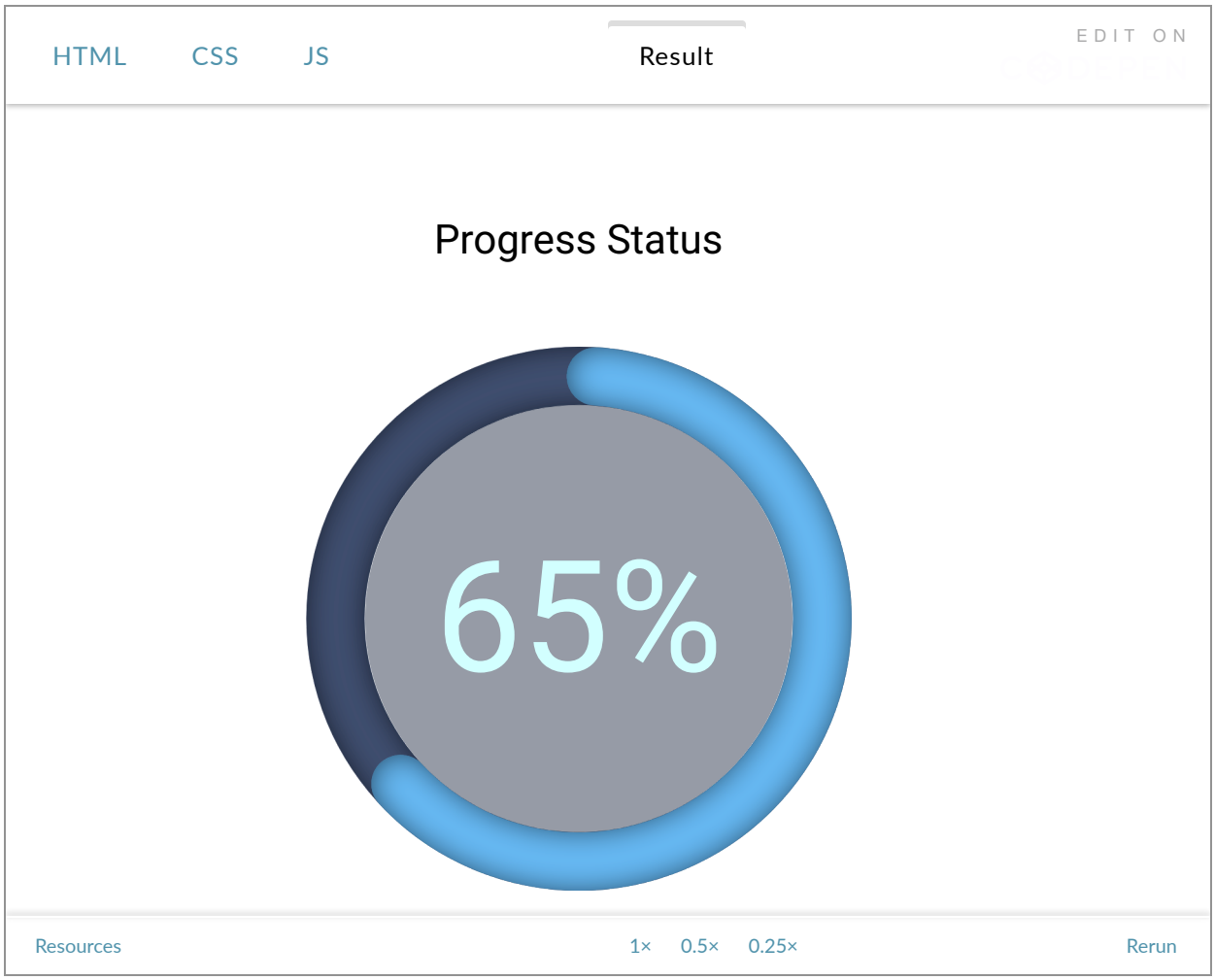
```
        });

    inset_shadow.append("svg:feComposite")
        .attr({
            operator:'in',
            in:'color',
            in2:'inverse',
            result:'shadow'
        });

    inset_shadow.append("svg:feComposite")
        .attr({
            operator:'over',
            in:'shadow',
            in2:'SourceGraphic'
        });

    var pathBackground=svg.append('path')
        .attr({
            d:arc
        })
        .style({
            fill:color[0],
            filter:'url(#inset-shadow)'
        });


    var pathForeground=svg.append('path')
        .datum({endAngle:0})
        .attr({
            d:arcLine
        })
        .style({
```

```
            fill:color[1],
            filter:'url(#inset-shadow)'
        });
    var circle=svg.append('circle')
        .attr({
            cx:0,
            cy:0,
            r:innerRadius
        })
        .style({
            fill:color[2],
            'fill-opacity':.5
        });

    var middleCount=svg.append('text')
        .datum(0)
        .text(function(d){
            return d+'%';
        })

        .attr({
            class:'middleText',
            'text-anchor':'middle',
            dy:27
        })
        .style({
            fill:d3.rgb(color[1]).brighter(2),
            'font-size':'80px'


        });

    var oldValue=0;
```

```
var arcTween=function(transition, newValue,oldValue) {
    transition.attrTween("d", function (d) {
        var interpolate = d3.interpolate(d.endAngle,
((2*Math.PI))*(newValue/100));

        var interpolateCount = d3.interpolate(oldValue,
newValue);
        return function (t) {
            d.endAngle = interpolate(t);

middleCount.text(Math.floor(interpolateCount(t))+'%');
            return arcLine(d);
        };
    });
};


var animate=function(){

    pathForeground.transition()
            .duration(750)
            .ease('cubic')
            .call(arcTween,percent,oldValue);
    oldValue=percent;
    percent=(Math.random() * 60) + 20;
    setTimeout(animate,1500);
};

setTimeout(animate,0);



body {
```

```
    background-color: #1B1F2A;
    width: 100%;
    font-family: 'Roboto', sans-serif;
    height: 100%;
}

.widget {
    margin: 0 auto;
    width:350px;
    margin-top:50px;
    background-color: #3f5175;
    border-radius: 5px;
    box-shadow: 0px 0px 1px 0px #06060d;

}

.header{
    background-color: #29384D;
    height:40px;
    color:#929DAF;
    text-align: center;
    line-height: 40px;
    border-top-left-radius: 7px;
    border-top-right-radius: 7px;
    font-weight: 400;
    font-size: 1.5em;
    text-shadow: 1px 1px #06060d;
}

.chart-container{
    padding:25px;
}
```

Click on `Run Pen` to see the full demo.



PROGRESS CHART 4:

The last type of Progress chart we will be creating has some important changes. First of all we will be using svg shadow (`feGaussianBlur`). We will also rotate the chart by 180 Degree. We will use the `rotate()` function `transform:'rotate(180)'` to transform it.

```
var percent = 85;

var ratio=percent/100;

var pie=d3.layout.pie()
```

```
        .value(function(d){return d})
        .sort(null);

var w=300,h=300;

var outerRadius=(w/2)-10;
var innerRadius=90;


var color = ['#ececec','#f06b3e','#888888'];

var arc=d3.svg.arc()
        .innerRadius(0)
        .outerRadius(outerRadius)
        .startAngle(0)
        .endAngle(2*Math.PI);


var arcLine=d3.svg.arc()
        .innerRadius(innerRadius)
        .outerRadius(outerRadius-10)
        .startAngle(0);

var svg=d3.select("#chart")
        .append("svg")
        .attr({
            width:w,
            height:h,
            class:'shadow'
        }).append('g')
        .attr({
            transform:'translate('+w/2+','+h/2+')'
        });
```

```
var defs = svg.append("svg:defs");

var inset_shadow = defs.append("svg:filter")
        .attr("id", "inset-shadow");

inset_shadow.append("svg:feOffset")
        .attr({
            dx:0,
            dy:0
        });

inset_shadow.append("svg:feGaussianBlur")
        .attr({
            stdDeviation:4,
            result:'offset-blur'
        });

inset_shadow.append("svg:feComposite")
        .attr({
            operator:'out',
            in:'SourceGraphic',
            in2:'offset-blur',
            result:'inverse'
        });

inset_shadow.append("svg:feFlood")
        .attr({
            'flood-color':'black',
            'flood-opacity':.7,
            result:'color'
        });
```

```
inset_shadow.append("svg:feComposite")
        .attr({
            operator:'in',
            in:'color',
            in2:'inverse',
            result:'shadow'
        });

inset_shadow.append("svg:feComposite")
        .attr({
            operator:'over',
            in:'shadow',
            in2:'SourceGraphic'
        });

var path=svg.append('path')
        .attr({
            d:arc
        })
        .style({
            fill:color[0],
            filter:'url(#inset-shadow)'
        });


var pathForeground=svg.append('path')
        .datum({endAngle:0})
        .attr({
            d:arcLine,
            transform:'rotate(180)'
        })
        .style({
            fill:color[1],
```

```
            filter:'url(#inset-shadow)'
        });


var middleCount=svg.append('text')
        .datum(0)
        .text(function(d){
            return d;
        })

        .attr({
            class:'middleText',
            'text-anchor':'middle',
            dy:25
        })
        .style({
            fill:d3.rgb(color[2]),
            'font-size':'80px',
            filter:'url(#inset-shadow)'


        });

var oldValue=0;
var arcTween=function(transition, newValue,oldValue) {
    transition.attrTween("d", function (d) {
        var interpolate = d3.interpolate(d.endAngle,
((2*Math.PI))*(newValue/100));

        var interpolateCount = d3.interpolate(oldValue,
newValue);
        return function (t) {
            d.endAngle = interpolate(t);
```

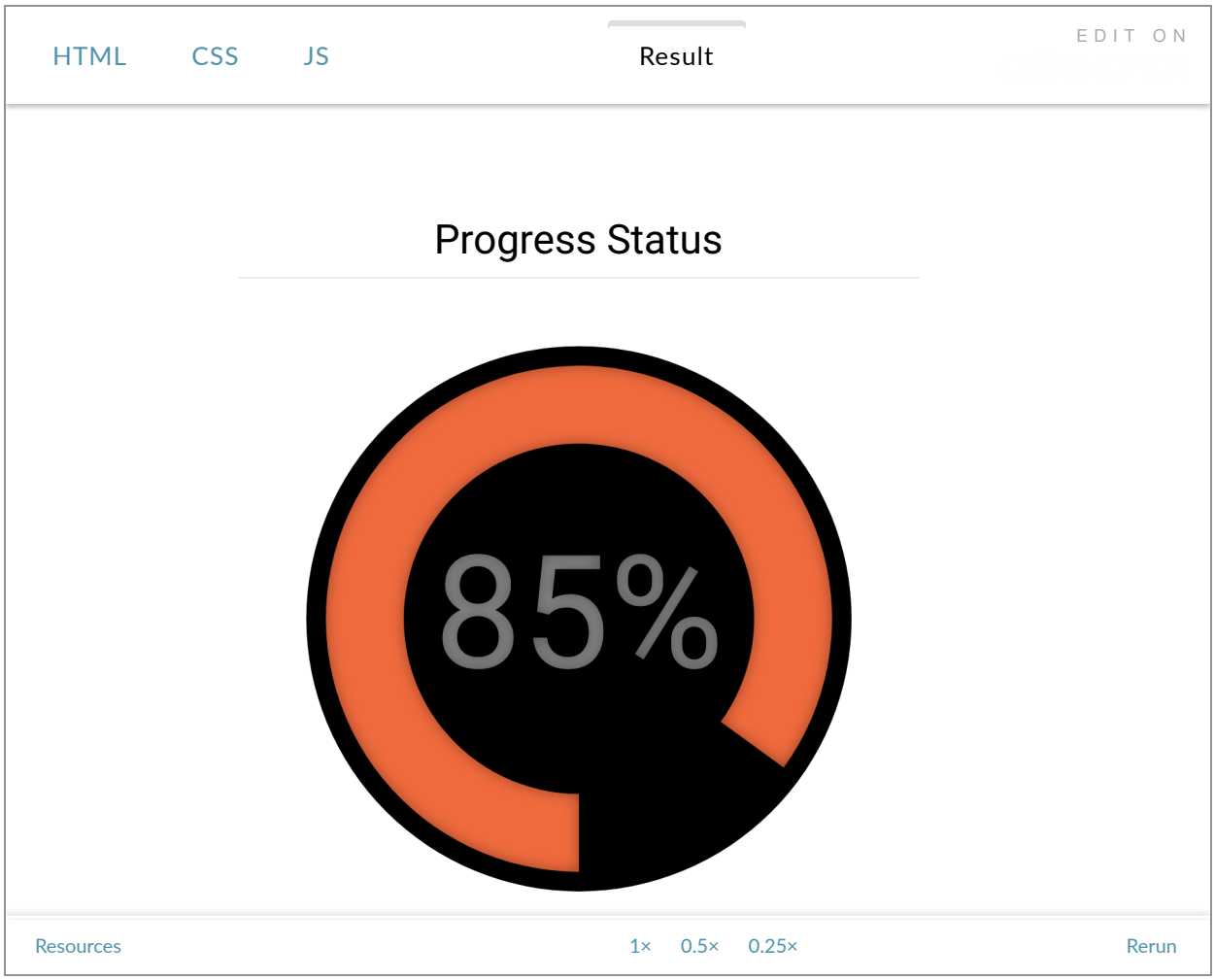
```
middleCount.text(Math.floor(interpolateCount(t))+'%');
            return arcLine(d);
        };
    });
};


var animate=function(){

    pathForeground.transition()
            .duration(750)
            .ease('cubic')
            .call(arcTween,percent,oldValue);
    oldValue=percent;
    percent=(Math.random() * 80) + 20;
    setTimeout(animate,1500);
};

setTimeout(animate,0);
```

We will use a light theme for this chart. Here is the CSS for that.

```
body {
    background-color: #ccc;
    width: 100%;
    font-family: 'Roboto', sans-serif;
    height: 100%;
}

.widget {
    margin: 0 auto;
```

```
    width:350px;
    margin-top:50px;
    background-color:#fff;
    -background-color: #222D3A;
    border-radius: 5px;
    box-shadow: 1px 1px 4px 0px rgba(0,0,0,0.3);

}

.header{
    background-color: #eee;
    height:40px;
    color:#555;
    text-align: center;
    line-height: 40px;
    border-top-left-radius: 7px;
    border-top-right-radius: 7px;
    font-weight: 400;
    font-size: 1.5em;
    text-shadow: 1px 1px #fff;
    border-bottom: 1px solid #eaeaea;
}

.chart-container{
    padding:25px;
}
```

Click on `Run Pen` to see the full demo.



In Next lesson we will further customize the Progress Charts.

Related

Create custom progress chart using d3.js – Part2
In "D3.js"

Create custom progress chart using d3.js – Part1
In "D3.js"

Create iWatch Activity Chart using d3.js
In "D3.js"

Filed Under: D3.js, JavaScript | Tagged With: chart, d3.js, JavaScript, Learn, Programming, Progress Chart, step by step, tutorial, Visualization

Comments

Alex says
June 6, 2016 at 6:25 am

(Edit)

Great work!

I was wondering if there was an easy way to configure this so you can show 3 or 4 progress charts within the same div, (I suppose this could be done with any of the examples?)

Reply

A Developer Diary says
June 10, 2016 at 4:18 am

(Edit)

Hi Alex,
Appreciate your feedback !

Yes, you should be able to do by wrapping the entire code in a function. You can pass the id of the div elements to the function.

I have created a sample example for you. Here is the link:

http://codepen.io/adeveloperdiary/pen/ezJVmm

Reply

Robindra says
February 8, 2017 at 10:50 am

(Edit)

This is really awesome tutorial. It help me a lot. Thank you very very much.

Reply

Leave a Reply

Logged in as Abhisek Jana. Edit your profile. Log out? Required fields are marked *

Comment *

Post Comment

This site uses Akismet to reduce spam. Learn how your comment data is processed.