

# A Developer Diary

{about:"code learn and share"}

[Home](#)[Data Science](#)[Java](#)[JavaScript](#)[jBPM](#)[Tools](#)[Tips](#)[About](#)

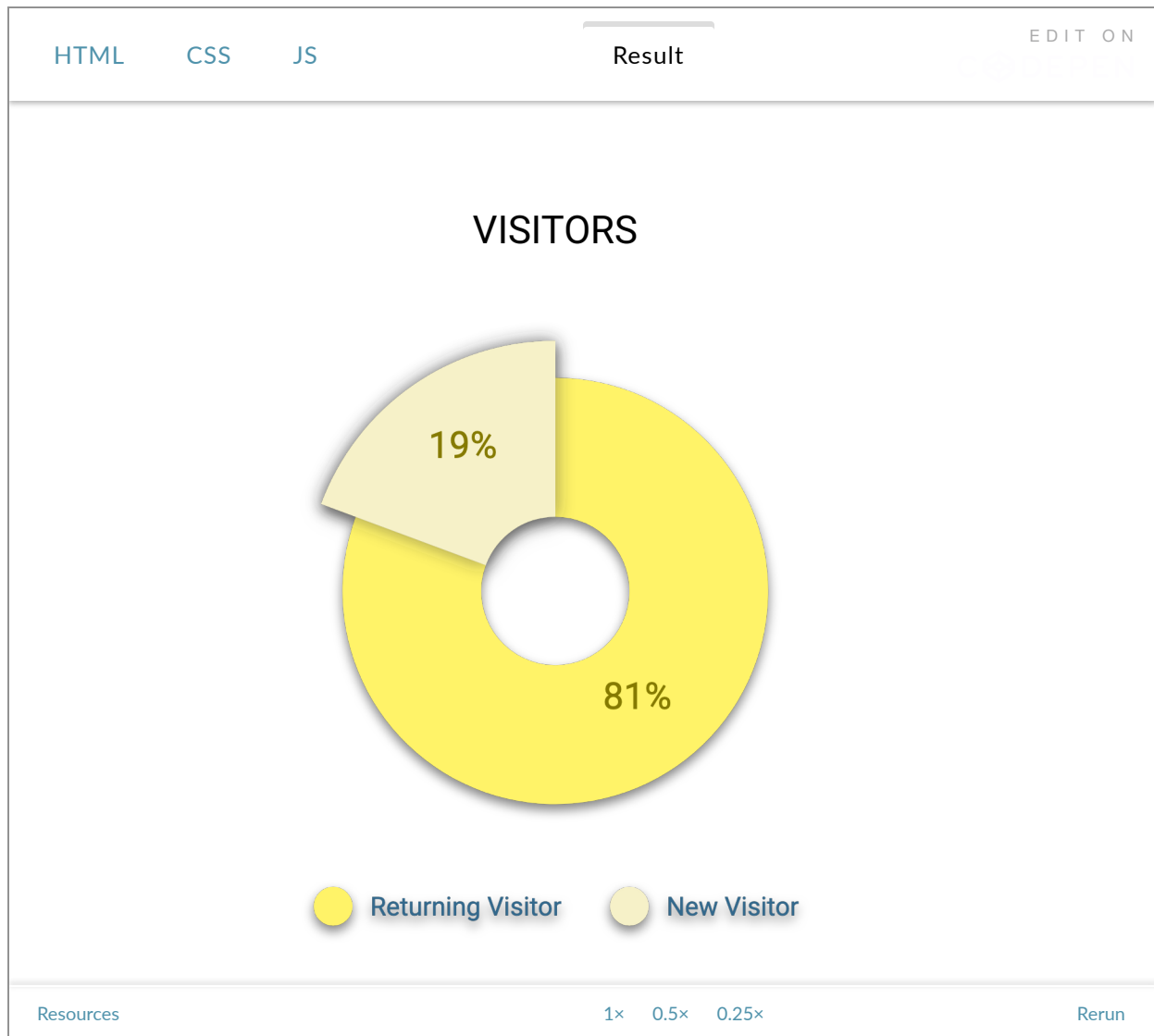
February 18, 2016 By [Abhisek Jana](#) — [Leave a Comment \(Edit\)](#)

## Create Pie Area Chart using d3.js



This is a quick tutorial on how to Create Pie Area Chart using d3.js. While we are exploring different ways of creating the Pie chart, its very easy and simple to create the Pie Area Chart.

Lets look at the demo first :



In case you want to recap how to create a simple donut chart refer my other post on

The JSON data needs to have the width variation data. The `growth` element contains that.

```
var dataset = [  
  { name: 'Returning Visitor', count: 500,  
    growth:115 , percent:81},  
  { name: 'New Visitor', count: 120, growth:135 ,  
    percent:19}  
];
```

Next set the `data.growth` in the `outerRadius` using a function.

```
var arc=d3.svg.arc()  
    .outerRadius(function(d){ return d.data.growth;})  
    .innerRadius(innerRadius);
```

The remaining code is very straight forward. If you need help with the donut chart, please refer my earlier post on [Create a simple Donut Chart using D3.js](#)

## Visitors

```
var dataset = [  
    { name: 'Returning Visitor', count: 500, growth:115 ,  
      percent:81},  
    { name: 'New Visitor', count: 120, growth:135 ,  
      percent:19}  
];
```

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

```
var w=300,h=350;
```

```
var innerRadius=40;
```

```
var color = d3.scale.ordinal()  
    .range(['#fff368', '#f6f1c8']);  
  
var arc=d3.svg.arc()  
    .outerRadius(function(d){ return d.data.growth;})  
    .innerRadius(innerRadius);  
  
var svg=d3.select("#chart")  
    .append("svg")  
    .attr({  
        width:w,  
        height:h,  
        class:'shadow'  
    }).append('g')  
    .attr({  
        transform:'translate('+w/2+', +(h-50)/2+)'  
    });  
  
var id = "md-shadow";  
var deviation = 2;  
var offset = 2;  
var slope = 0.25;  
  
var defs = svg.append("defs");  
  
var filter = defs.append("filter")  
    .attr("id", "drop-shadow")  
    .attr("height", "150%");
```

```
filter.append("feGaussianBlur")
    .attr("in", "SourceAlpha")
    .attr("stdDeviation", 5)
    .attr("result", "blur");

filter.append("feOffset")
    .attr("in", "blur")
    .attr("dx", 0)
    .attr("dy", 0)
    .attr("result", "offsetBlur");

filter.append("feFlood")
    .attr("in", "offsetBlur")
    .attr("flood-color", "#333")
    .attr("flood-opacity", ".5")
    .attr("result", "offsetColor");
filter.append("feComposite")
    .attr("in", "offsetColor")
    .attr("in2", "offsetBlur")
    .attr("operator", "in")
    .attr("result", "offsetBlur");

var feMerge = filter.append("feMerge");

feMerge.append("feMergeNode")
    .attr("in", "offsetBlur");
feMerge.append("feMergeNode")
    .attr("in", "SourceGraphic");

var path=svg.selectAll('path')
    .data(pie(dataset))
    .enter()
```

```
.append('path')
.attr({
  d:arc,
  fill:function(d,i){
    return color(d.data.name);
  }
})
.style("filter", function(d,i){
  if(i>0)
    return "url(#drop-shadow)";
})
);
```

```
path.transition()
  .duration(1000)
  .attrTween('d', function(d) {
    var interpolate = d3.interpolate({startAngle:
0, endAngle: 0}, d);
    return function(t) {
      return arc(interpolate(t));
    };
  });
```

```
var restOfTheData=function(){
  var text=svg.selectAll('text')
    .data(pie(dataset))
    .enter()
    .append("text")
    .attr("transform", function (d) {
      return "translate(" + arc.centroid(d) +
    " " + arc.centroid(d) +
    ")";
```

```
    })

    .attr("text-anchor", "middle")
    .text(function(d){
        return d.data.percent+"%" ;
    })
    .style({
        fill:'#857a00',
        'font-size':'20px'

    });

var legendRectSize=20;

var legend=svg.selectAll('.legend')
    .data(color.domain())
    .enter()
    .append('g')
    .attr({
        class:'legend',
        transform:function(d,i){
            //Just a calculation for x & y
            position
            return 'translate(' + ((i*160)-130) +
            ',160)';
        }
    });
legend.append('rect')
    .attr({
        width:legendRectSize,
        height:legendRectSize,
        rx:20,
        ry:20
```



```
    })
    .style({
      fill:color,
      stroke:color
    });

legend.append('text')
  .attr({
    x:30,
    y:15
  })
  .text(function(d){
    return d;
  }).style({
    fill:'#32688c',
    'font-size':'14px'
  });

};

setTimeout(restOfTheData,1000);

body {
  background-color: #0c3049;
  width: 100%;
  font-family: 'Roboto', sans-serif;
  height: 100%;
}

.widget {
  margin: 0 auto;
  width:350px;
```

```
margin-top:50px;
background-color: #222D3A;
border-radius: 5px;
box-shadow: 0px 0px 1px 0px #06060d;

}

.header{

    height:40px;
    text-align: center;
    line-height: 40px;
    border-top-left-radius: 7px;
    border-top-right-radius: 7px;
    font-size: 1.5em;
    background-color: #e5d847;
    color:#2E2A0D;
    text-shadow: 1px 1px #A19532;
    text-transform: uppercase;
    font-weight: 800;
}

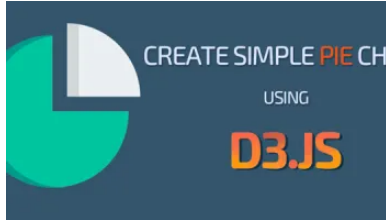
.chart-container{
    padding:25px;
}

.shadow {
    -webkit-filter: drop-shadow( 0px 3px 3px
    rgba(0,0,0,.5) );
    filter: drop-shadow( 0px 3px 3px rgba(0,0,0,.5) );
}
```

Please find the code in [github](#).

---

## Related



Create a Simple Pie Chart  
using D3.js

In "D3.js"



Create Pie Charts using  
D3.js

In "D3.js"



Create a simple Donut Chart  
using D3.js

In "D3.js"

---

Filed Under: [D3.js](#), [JavaScript](#)  
[step by step](#)

Tagged With: [Area Chart](#), [d3.js](#), [example](#), [JavaScript](#), [pie](#), [Pie Chart](#),

## Subscribe to stay in loop

\* indicates required

Email Address \*

Subscribe

## 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](#).

Copyright © 2024 A Developer Diary