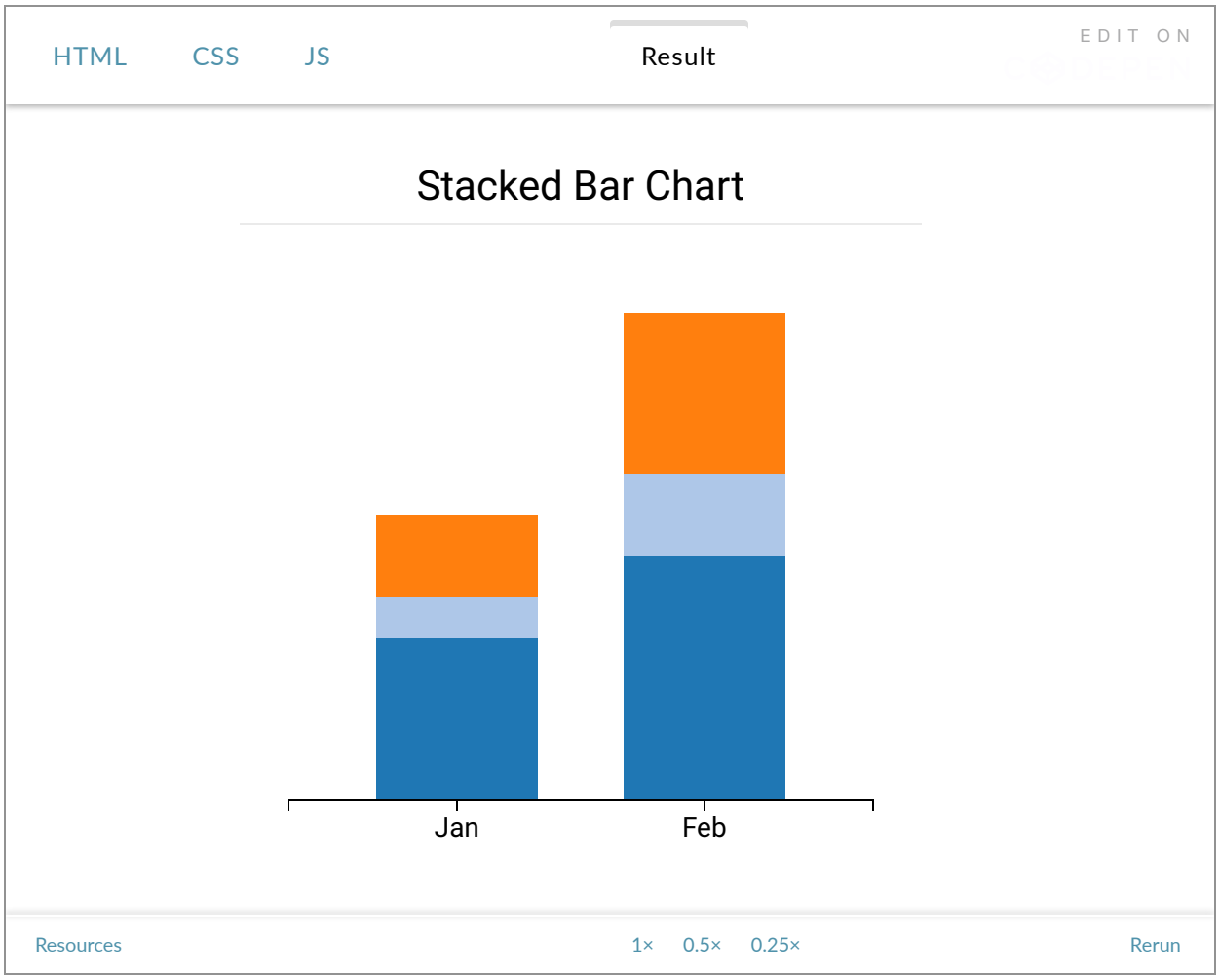
March 22, 2016 By Abhisek Jana — 15 Comments (Edit)

How to Create Stacked Bar Chart using d3.js

In this **How to Create Stacked Bar Chart using d3.js** post we will learn not only to code but the mathematical calculation behind creating a stacked bar chart using d3. Even if you have probably copy pasted a working version the code, I strongly recommend you to go though this tutorial in order to get a solid understanding on how this works.

No fancy stuff today, we will create a very simple basic stacked bar chart. Here is the demo we will be creating. Let's assume we have three products A , B & C. We want to display the monthly sales in using a stacked bar chart.



Our JSON data object for the chart above would look like this:

```
var data=[

    {month:'Jan', A:20, B: 5,  C: 10},
    {month:'Feb', A:25, B: 10, C: 20}

];
```

Our Final SVG would look like below. Each category (A,B or C) would be part of one group (`g` svg element).So we will first draw all `rect` for A , then for B & C. Lets not worry about `x` pos and `width` since they are generic.

Now we need to change our JSON object (`data`) so that we can easily draw above svg elements. At first we will change our `data` to `dataIntermediate`. Here we will simplify by creating one array for each category. The x would repeat for each array.

```
var dataIntermediate=[
        [
            {x:'Jan', y:20},
            {x:'Feb', y:25}
        ],
        [
            {x:'Jan', y:5},
            {x:'Feb', y:10}
        ],
        [
            {x:'Jan', y:10},
            {x:'Feb', y:20}
        ]
];
```

Here is the code for changing the `data` to `dataIntermediate`.

```
var dataIntermediate=['A','B'].map(function(key,i){
    return data.map(function(d,j){
        return {x: d['month'], y: d[key] };
    })
})
```

Then we will pass `dataIntermediate` json to `d3.layout.stack()` and it will provide the below output. We can however create this by our own but why not d3 take care of this for us. The `d3.layout.stack()` will add a `y0` attribute to our array.

```
var dataStackLayout=[
    [
        {x:'Jan', y0:0, y:20},
        {x:'Feb', y0:0, y:25}
    ],
    [
        {x:'Jan', y0:20, y:5},
        {x:'Feb', y0:25, y:10}
    ]
    ,
    [
        {x:'Jan', y0:25, y:10},
        {x:'Feb', y0:35, y:20}
    ]
];
```

Next we will create the `xScale` & `yScale` with the domain value. The domain for `xScale` should be straight forward, just take the first element from the array and get all the `x` values, `['Jan','Feb']`) in this case.

We will set a max & min value to the `yDomain`. In order to calculate the max value, we will take the last element from the `dataStackLayout` array and sum `d.y0` & `d.y`, then get the max out of all the values. The min would be 0 in most of the cases.

```
// xScale Domain
dataStackLayout[0].map(function(d) { return d.x; })

//yScale Domain - Max value
d3.max(dataStackLayout[dataStackLayout.length - 1],
function(d) { return d.y0 + d.y; })
```

Now its time to draw the chart. We will focus on the 4 attributes, `x`, `y`, `height` & `width`. The x and width is straight forward. The y would be summation of `d.y0` & `d.y`, since we should start drawing from top. The calculation for height might be confusing to you, but remember in svg the left-top corner is the center `[0,0]` and `yScale(0) > yScale(10)`. So `yScale(d.y0)` represents the bottom position and `yScale(d.y + d.y0)` is the top position. The subtraction would give us the height of the rect element.

```
//Left Position for X
x=  xScale(d.x)

//Top Position for Y
y = yScale(d.y + d.y0)

//Bottom Y - TopY = Height of the element
height=yScale(d.y0) - yScale(d.y + d.y0)

// Width as per rangeBand()
width= xScale.rangeBand()
```

Find the full code here:

```
var data = [
    {month: 'Jan', A: 20, B: 5, C: 10},
    {month: 'Feb', A: 30, B: 10, C: 20}
];

var xData = ["A", "B", "C"];

var margin = {top: 20, right: 50, bottom: 30, left: 50},
        width = 400 - margin.left - margin.right,
        height = 300 - margin.top - margin.bottom;

var x = d3.scale.ordinal()
        .rangeRoundBands([0, width], .35);

var y = d3.scale.linear()
        .rangeRound([height, 0]);

var color = d3.scale.category20();

var xAxis = d3.svg.axis()
        .scale(x)
        .orient("bottom");

var svg = d3.select("body").append("svg")
        .attr("width", width + margin.left +
margin.right)
        .attr("height", height + margin.top +
margin.bottom)
        .append("g")
        .attr("transform", "translate(" + margin.left +
```

```
    "," + margin.top + ")");

    var dataIntermediate = xData.map(function (c) {
        return data.map(function (d) {
            return {x: d.month, y: d[c]};
        });
    });

    var dataStackLayout = d3.layout.stack()
    (dataIntermediate);

    x.domain(dataStackLayout[0].map(function (d) {
        return d.x;
    }));

    y.domain([0,
        d3.max(dataStackLayout[dataStackLayout.length - 1],
                function (d) { return d.y0 + d.y;})
        ])
      .nice();

    var layer = svg.selectAll(".stack")
            .data(dataStackLayout)
            .enter().append("g")
            .attr("class", "stack")
            .style("fill", function (d, i) {
                return color(i);
            });

    layer.selectAll("rect")
            .data(function (d) {
                return d;
            })
```

```
        .enter().append("rect")
        .attr("x", function (d) {
            return x(d.x);
        })
        .attr("y", function (d) {
            return y(d.y + d.y0);
        })
        .attr("height", function (d) {
            return y(d.y0) - y(d.y + d.y0);
        })
        .attr("width", x.rangeBand());

svg.append("g")
        .attr("class", "axis")
        .attr("transform", "translate(0," + height + ")")
        .call(xAxis);
```

I hope now you would know How to Create Stacked Bar Chart using d3.js. Later we will use React to create stacked bar charts.

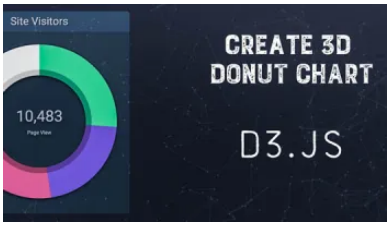
Related

How to create reusable charts with React and D3 Part3

In "D3.js"

Create Pie Charts using D3.js

In "D3.js"



Create 3D Donut Chart using D3.js

In "D3.js"

Filed Under: D3.js Tagged With: bar, Code, d3.js, Learn, Programming, stack chart, stacked bar, step by step, tutorial, Visualization

Comments

Mark Olbert says
May 19, 2016 at 12:33 am

(Edit)

Thank you! I struggled for most of a day getting a stacked bar chart to work in d3. Every other example I found online left out important details, like the shape of the data before and after it was 'normalized' to work with layout.stack().

With what I read in your article I was able to get my code to work. Kudos to you!

– Mark

Reply

A Developer Diary says

May 19, 2016 at 10:42 am

(Edit)

Hi Mark,
Thank you for the feedback, glad to know that this tutorial has helped you.

I had faced similar challenges while creating the stack chart for the first time.

Thanks,
A Developer Diary

Reply

Paramveer says

July 16, 2016 at 6:28 am

(Edit)

Hey!

I am getting this error :"Uncaught TypeError: Cannot read property 'ordinal' of undefined"..
Help me rectify.I copied ur code to verify.Error remains.
Reply

Rahul says
July 26, 2016 at 8:31 am

(Edit)

I agree with Mark about data transformation. Thank you A Developer Diary

Reply

A Developer Diary says
August 10, 2016 at 4:10 pm

(Edit)

Thanks for your feedback Rahul !

Reply

Saurabh says
August 9, 2016 at 7:36 am

(Edit)

Very Good Explanation, Thank you so much.

Reply

A Developer Diary says
August 10, 2016 at 4:11 pm

(Edit)

You are most welcome Saurabh.

Reply

Prachi says
October 5, 2016 at 6:06 pm

(Edit)

Amazing! Was struggling to create stacked bar graph with angular.js for about almost 24 hours. Read many online blogs but nothing could match your solution and explanation. Got working graph in first attempt.

Thanks a lot!!

Reply

A Developer Diary says
October 5, 2016 at 11:30 pm

(Edit)

Hi Prachi,
Great to hear that the article was helpful. Thanks for your feedback !!!

Reply

Ajmal Yazdani says
December 14, 2016 at 3:08 am

(Edit)

Hi,
Could you please help me to update the code for V4?

Reply

ofey says
July 25, 2017 at 5:33 pm

(Edit)

Yes for v4 would be great.
Thanks for the article

Reply

Grzegorz Kaznocha says
January 16, 2017 at 6:33 pm

(Edit)

Hi! I want change color of all rects without layer where I hold mouse. Have you any idea?

Reply

steve says
February 27, 2017 at 9:50 pm

(Edit)

your ‘full code’ on this page uses d3.select(“body”).append(“svg”) when it should be d3.select(“chart”).append(“svg”)

Cheers

Reply

Mitchell Cravens says
March 13, 2017 at 1:49 pm

(Edit)

Haha Abhisek I just found this via Google because I’m working as a Javascript developer for a different company now and need to make a stacked bar chart with D3! What’s up man! Hope you’re well!

(- Mitch from Cognizant)

Reply

Abhisek Jana says
March 13, 2017 at 1:55 pm

(Edit)

> WOW .. the world is really small ! I am very excited to hear back from you. I will email you ….
>
> 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.