**D3 Bar Chart implementation steps**

1. First follow all the steps in the documentation for implementing Pie Chart
2. Add the following styles under styles.css

.myBarGraph svg {

width: 100%;

height: 100%;

position: center;

}

.myBarGraph .toolTip {

font-family: "Helvetica Neue", Helvetica, Arial, sans-serif;

position: absolute;

display: none;

width: auto;

height: auto;

background: none repeat scroll 0 0 white;

border: 0 none;

border-radius: 8px 8px 8px 8px;

box-shadow: -3px 3px 15px #888888;

color: black;

font: 12px sans-serif;

padding: 5px;

text-align: center;

}

.myBarGraph text {

font: 28px sans-serif;

color: white;

}

.myBarGraph text.value {

fill: white;

}

.myBarGraph .axisHorizontal path{

fill: none;

}

.myBarGraph .axisHorizontal .tick line {

stroke-width: 1;

stroke: rgba(0, 0, 0, 0.2);

}

.myBarGraph .bar {

fill: steelblue;

fill-opacity: .9;

}

1. Add the following code in BarChart.js

import React, { Component } from 'react';

import PropTypes from 'prop-types';

import { scaleLinear, scaleOrdinal } from 'd3-scale';

import { max } from 'd3-array';

import { schemeCategory10, select, pie, arc, event, axisBottom } from 'd3';

class BarChart extends Component {

constructor(props, context) {

super(props, context);

this.createBarChart = this.createBarChart.bind(this);

}

componentDidMount() {

this.createBarChart();

}

componentDidUpdate() {

this.createBarChart();

}

createBarChart() {

let data = this.props.data;

const node = this.node;

let div = select(node).append("div").attr("class", "toolTip");

let axisMargin = 20;

let margin = 40;

let valueMargin = 4;

let width = 960;

let height = 400;

let barHeight = (height - axisMargin - margin \* 2) \* 0.4 / data.length;

let barPadding = (height - axisMargin - margin \* 2) \* 0.6 / data.length;

let bar = 0;

let svg = null;

let scale = 0;

let xAxis = 0;

let labelWidth = 0;

let color = scaleOrdinal(["red","blue","green"]);

let maxChart = max(data, function (d) { return d.keyValue; });

svg = select(node)

.append("svg")

.attr("viewBox", "0 0 960 400")

.attr("preserveAspectRatio", "xMidYMid meet");

bar = svg.selectAll("g")

.data(data)

.enter()

.append("g");

bar.attr("class", "bar")

.attr("cx", 0)

.attr("transform", function (d, i) {

return "translate(" + margin + "," + (i \* (barHeight + barPadding) + barPadding) + ")";

});

bar.append("text")

.attr("class", "label")

.attr("y", barHeight / 2)

.attr("dy", ".35em") //vertical align middle

.text(function (d) {

return d.key;

}).each(function () {

labelWidth = Math.ceil(Math.max(labelWidth, this.getBBox().width));

});

scale = scaleLinear()

.domain([0, maxChart])

.range([0, width - margin \* 2 - labelWidth]);

xAxis = axisBottom(scale)

.tickSize(-height + 2 \* margin + axisMargin);

bar.append("rect")

.attr("transform", "translate(" + labelWidth + ", 0)")

.attr("height", barHeight)

.attr("fill", function (d) { return color(d.keyValue); })

.attr("width", function (d) {

return scale(d.keyValue);

});

bar.append("text")

.attr("class", "value")

.attr("y", barHeight / 2)

.attr("dx", -valueMargin + labelWidth) //margin right

.attr("dy", ".35em") //vertical align middle

.attr("text-anchor", "end")

.text(function (d) {

return (d.keyValue );

})

.attr("x", function (d) {

let width = this.getBBox().width;

return Math.max(width + valueMargin, scale(d.keyValue));

});

bar.on("mousemove", function (d) {

div.style("left", 100 + "px");

div.style("top", 10 + "px");

div.style("display", "inline-block");

div.html((d.key) + "<br>" + (d.keyValue) );

});

bar.on("mouseout", function (d) {

div.style("display", "none");

});

svg.insert("g", ":first-child")

.attr("class", "axisHorizontal")

.attr("transform", "translate(" + (margin + labelWidth) + "," + (height - axisMargin - margin) + ")")

.call(xAxis);

}

render() {

return (<div>

<div ref={node => this.node = node} className="myBarGraph"></div>

</div>

);

}

}

BarChart.propTypes = {

data: PropTypes.array

};

export default BarChart;

1. Import the BarChart.js in the component where we want bar chart to be displayed.
2. Add the following BarChart control in your code where you want to display the pie chart. This control should have an attribute **data** which accepts a json in the below following format.

<BarChart data={[

{ key: 'High School', keyValue: 20 },

{ key: 'Middle School', keyValue: 35 },

{ key: 'Elementory School', keyValue: 45 }

]} />

1. Will get the below output .

