/\*

\* Power BI Visual CLI

\*

\* Copyright (c) Microsoft Corporation

\* All rights reserved.

\* MIT License

\*

\* Permission is hereby granted, free of charge, to any person obtaining a copy

\* of this software and associated documentation files (the ""Software""), to deal

\* in the Software without restriction, including without limitation the rights

\* to use, copy, modify, merge, publish, distribute, sublicense, and/or sell

\* copies of the Software, and to permit persons to whom the Software is

\* furnished to do so, subject to the following conditions:

\*

\* The above copyright notice and this permission notice shall be included in

\* all copies or substantial portions of the Software.

\*

\* THE SOFTWARE IS PROVIDED \*AS IS\*, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR

\* IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,

\* FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE

\* AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER

\* LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,

\* OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN

\* THE SOFTWARE.

\*/

module powerbi.extensibility.visual {

let staticData=[

{

value: 2,

category: 'Japan'

},

{

value: 5,

category: 'India'

},

{

value: 9,

category: 'USA'

},

{

value: 7,

category: 'Canada'

}

]; //creating static data array where each data piece will be a javascript obj

"use strict";

export class Visual implements IVisual {

private svg: d3.Selection<SVGElement>;

private barContainer: d3.Selection<SVGElement>; //holds the bars

constructor(options: VisualConstructorOptions) {

this.svg= d3.select(options.element) //initializing the property

.append('svg')

.classed('barChart', true); //creating svg element

this.barContainer= this.svg

.append('g') //g signifies grouping that all bars will be in it

.classed('barContainer',true);

}

public update(options: VisualUpdateOptions) {

let width=options.viewport.width;

let height= options.viewport.height; //accessing viewport to asses the dimensions

this.svg.attr({

width: width,

height: height //resizing

});//assigning attributes

let yScale= d3.scale.linear()

.domain([0,11]) //min and max values in our data

.range([height, 0]); //output values to scale it to

let xScale=d3.scale.ordinal()//coz x axis is text

.domain(staticData.map(dataPoint => dataPoint.category)) //map to extract categories as array

.rangeRoundBands([0,width],0.1,0.2);

let bars= this.barContainer //bars is accessing barcontainer

.selectAll('.bar') //select all elements which has a css class of bar

.data(staticData); //provided by d3 where d3 figures out how many bars to create based on the data

bars.enter() //provided by d3 which makes sure all the data that we have enters the view but does in a

// clever way by seeing which points are already rendered and which are required to be rendered

.append('rect') // we are telling here for each data points you want to render please create a rectangle

.classed('bar', true);

bars.attr({

width:xScale.rangeBand(),

height: data => height- yScale(<number> data.value),

x: data => xScale(data.category),

y: data => yScale(<number> data.value)

});

bars.exit().remove(); //remov elements when necessary and repaint them

}

}

}

Code for dynamic data extraction directly in visual.ts file in barchart project folder