// Set the dimensions of the canvas / graph

var margin = {top: 10, right: 20, bottom: 50, left: 50},

width = 800 - margin.left - margin.right,

height = 470 - margin.top - margin.bottom;

// parse the date / time

var parseTime = d3.timeParse("%d-%b-%y");

var padding = d3.format(".0s")(1000);

// set the ranges

var x = d3.scaleLog().range([0, width]);

var y = d3.scaleLinear().range([height, 0]);

// define the line

var valueline = d3.line()

.x(function(d) { return x(d.gdpPercap); })

.y(function(d) { return y(d.lifeExp); });

// append the svg obgect to the body of the page

// appends a 'group' element to 'svg'

// moves the 'group' element to the top left margin

var svg = d3.select("div.center").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 color = d3.scaleOrdinal(d3.schemeCategory10);

// Get the data

d3.tsv("data/gapminderDataFiveYear.tsv", function(error, data) {

if (error) throw error;

data = data.filter(function(d) {

d.year = +d.year;

d.gdpPercap = +d.gdpPercap;

d.lifeExp = +d.lifeExp;

d.pop = +d.pop;

return d.year == 1952 || d.year == 2007;

});

var rScale = d3.scaleLinear()

.domain([0, d3.max(data, function(d) {return d.pop; })])

.range([4, 10]);

//var padding = d3.format("0s");

console.log(data);

// Scale the range of the data

x.domain(d3.extent(data, function(d) { return d.gdpPercap; }));

y.domain([30, d3.max(data, function(d) { return d.lifeExp; })]);

// Add the scatterplot

svg.selectAll("dot")

.data(data)

.enter().append("circle")

.attr("r", function(d) {return rScale(d.pop); })

.attr("cx", function(d) { return x(d.gdpPercap); })

.attr("cy", function(d) { return y(d.lifeExp); })

.style("fill", function(d) {

if (d.year == 1952) {

return color(3);

}

if (d.year == 2007) {

return color(2);

}

})

.style('opacity', 0.8);

// Add the X Axis

svg.append("g")

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

.attr("font-family", "lato")

.call(d3.axisBottom(x)

.ticks(11, ".0s"));

//label for x axis

svg.append("text")

.attr("transform", "translate(" + (width / 2) + " ," +

(height + margin.top + 20) + ")")

.style("text-anchor", "middle")

.attr("font-weight", "bold")

.attr("font-size", "14px")

.attr("font-family", "sans-serif")

.text("GDP per Capita");

// Add the Y Axis

svg.append("g")

.call(d3.axisLeft(y));

//label for y axis

svg.append("text")

.attr("transform", "rotate(-90)")

.attr("font-size", "14px")

.attr("font-family", "sans-serif")

.attr("y", 0 - margin.left)

.attr("x", 0 - (height / 2))

.attr("dy", "1em")

.attr("font-weight", "bold")

.style("text-anchor", "middle")

.text("Life Expectancy");

//title

svg.append("text")

.attr("x", width / 2)

.attr("y", 0 + (margin.top / 2))

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

.attr("font-weight", "bold")

.attr("font-family", "sans-serif")

.style("font-size", "16px")

.style("text-decoration", "underline")

.text("GDP vs Life Expectancy (1952, 2007)");

//rect code for legend 2007

svg.append("rect")

.attr("x", 675)

.attr("y", 85)

.attr("width", 20)

.attr("height", 20)

.style("fill", function(d) {return color(2)});

//word 2007

svg.selectAll("labels")

.data(data)

.enter()

.append("text")

.attr("x", 700)

.attr("y", 97)

.attr("font-family", "sans-serif")

.style("font-size", "11px")

.text("2007")

.attr("text-anchor", "right");

//rect code for 1952

svg.append("rect")

.attr("x", 675)

.attr("y", 100)

.attr("width", 20)

.attr("height", 20)

.style("fill", function(d) {return color(3)});

//label 1952

svg.selectAll("labels")

.data(data)

.enter()

.append("text")

.attr("x", 700)

.attr("y", 115)

.attr("font-family", "sans-serif")

.style("font-size", "11px")

.text("1952")

.attr("text-anchor", "right");

});