<!DOCTYPE html>

<html>

<meta charset="utf-8">

<style> /\* set the CSS \*/

body { font: 12px Arial;}

path {

stroke: steelblue;

stroke-width: 2;

fill: none;

}

.axis path,

.axis line {

fill: none;

stroke: grey;

stroke-width: 1;

shape-rendering: crispEdges;

}

.legend {

font-size: 16px;

font-weight: bold;

text-anchor: middle;

}

</style>

<body>

<!-- load the d3.js library -->

<script src="http://d3js.org/d3.v3.min.js"></script>

<script>

// Set the dimensions of the canvas / graph

var margin = {top: 30, right: 20, bottom: 70, left: 50},

width = 600 - margin.left - margin.right,

height = 300 - margin.top - margin.bottom;

// Parse the date / time

var parseDate = d3.time.format("%b %Y").parse;

// Set the ranges

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

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

// Define the axes

var xAxis = d3.svg.axis().scale(x)

.orient("bottom").ticks(5);

var yAxis = d3.svg.axis().scale(y)

.orient("left").ticks(5);

// Define the line

var priceline = d3.svg.line()

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

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

// Adds the svg canvas

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 + ")");

// Get the data

d3.csv("stocks.csv", function(error, data) {

data.forEach(function(d) {

d.date = parseDate(d.date);

d.price = +d.price;

});

// Scale the range of the data

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

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

// Nest the entries by symbol

var dataNest = d3.nest()

.key(function(d) {return d.symbol;})

.entries(data);

var color = d3.scale.category10(); // set the colour scale

legendSpace = width/dataNest.length; // spacing for legend

// Loop through each symbol / key

dataNest.forEach(function(d,i) {

svg.append("path")

.attr("class", "line")

.style("stroke", function() { // Add the colours dynamically

return d.color = color(d.key); })

.attr("d", priceline(d.values));

// Add the Legend

svg.append("text")

.attr("x", (legendSpace/2)+i\*legendSpace) // spacing

.attr("y", height + (margin.bottom/2)+ 5)

.attr("class", "legend") // style the legend

.style("fill", function() { // dynamic colours

return d.color = color(d.key); })

.text(d.key);

});

// Add the X Axis

svg.append("g")

.attr("class", "x axis")

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

.call(xAxis);

// Add the Y Axis

svg.append("g")

.attr("class", "y axis")

.call(yAxis);

});

</script>

</body>

</html>