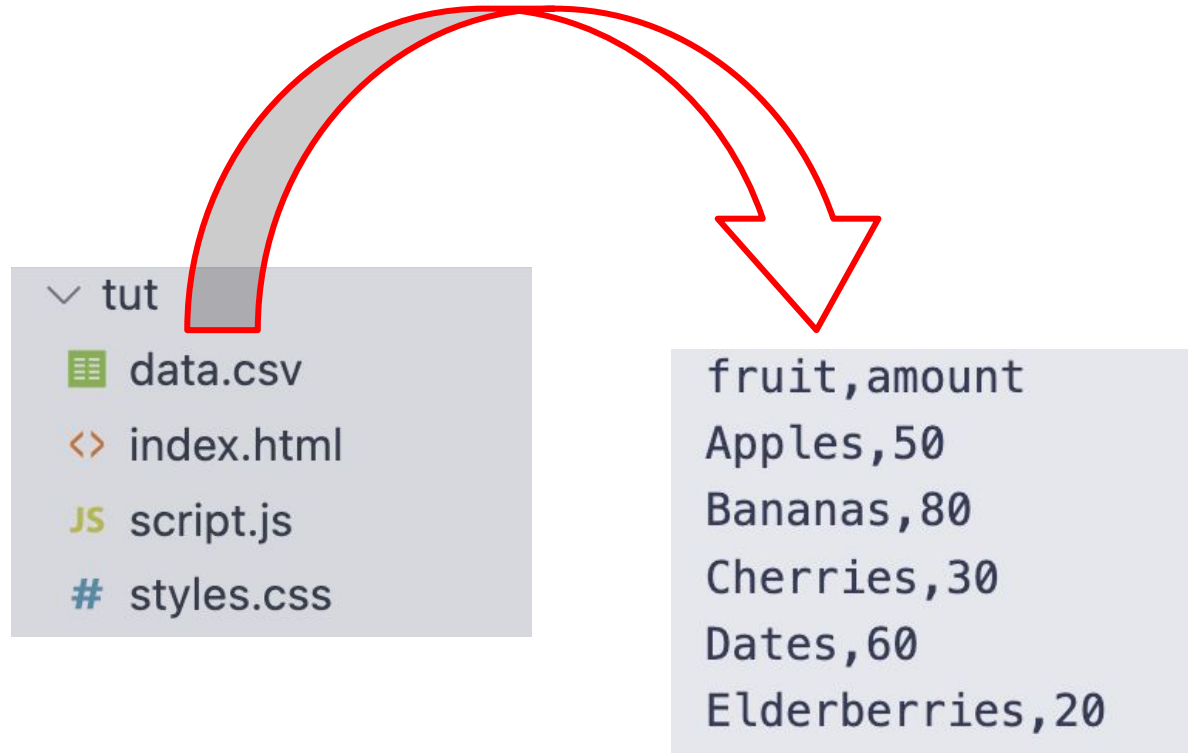


# Intro to D3

Data Visualisation - Tutorial 1

What does D3 do?

# Setting Up





# Loading & formatting data

```
// 1. Load the data
```

```
d3.csv("data.csv").then(function(data) {
```

```
// 2. Format the data
```

```
data.forEach(d => {
```

```
    d.amount = +d.amount; // The '+' forces text into a number
```

```
});
```

# The Scales

```
// 3. Set Dimensions and Margins
const width = 500;
const height = 300;
const margin = {top: 40, right: 150, bottom: 50, left: 50};
```

```
// 4. Create the Translators (Scales)
```

```
const xScale = d3.scaleBand()           Categorical
  .domain(data.map(d => d.fruit))
  .range([margin.left, width - margin.right])
  .padding(0.2);
```

```
const yScale = d3.scaleLinear()         Continuous Numerical
  .domain([0, 100])
  .range([height - margin.bottom, margin.top]); // Flipped for SVG
```

Top-left is (0,0)

# Bar graph Setup

```
const svgBar = d3.select("#bar-chart").append("svg")  
  .attr("width", width)  
  .attr("height", height);
```

Scalable Vector Graphic

```
// Draw X Axis  
svgBar.append("g")  
  .attr("transform", `translate(0, ${height - margin.bottom})`)  
  .call(d3.axisBottom(xScale));
```

```
// Draw Y Axis  
svgBar.append("g")  
  .attr("transform", `translate(${margin.left}, 0)`)  
  .call(d3.axisLeft(yScale));
```

# Creating Bars

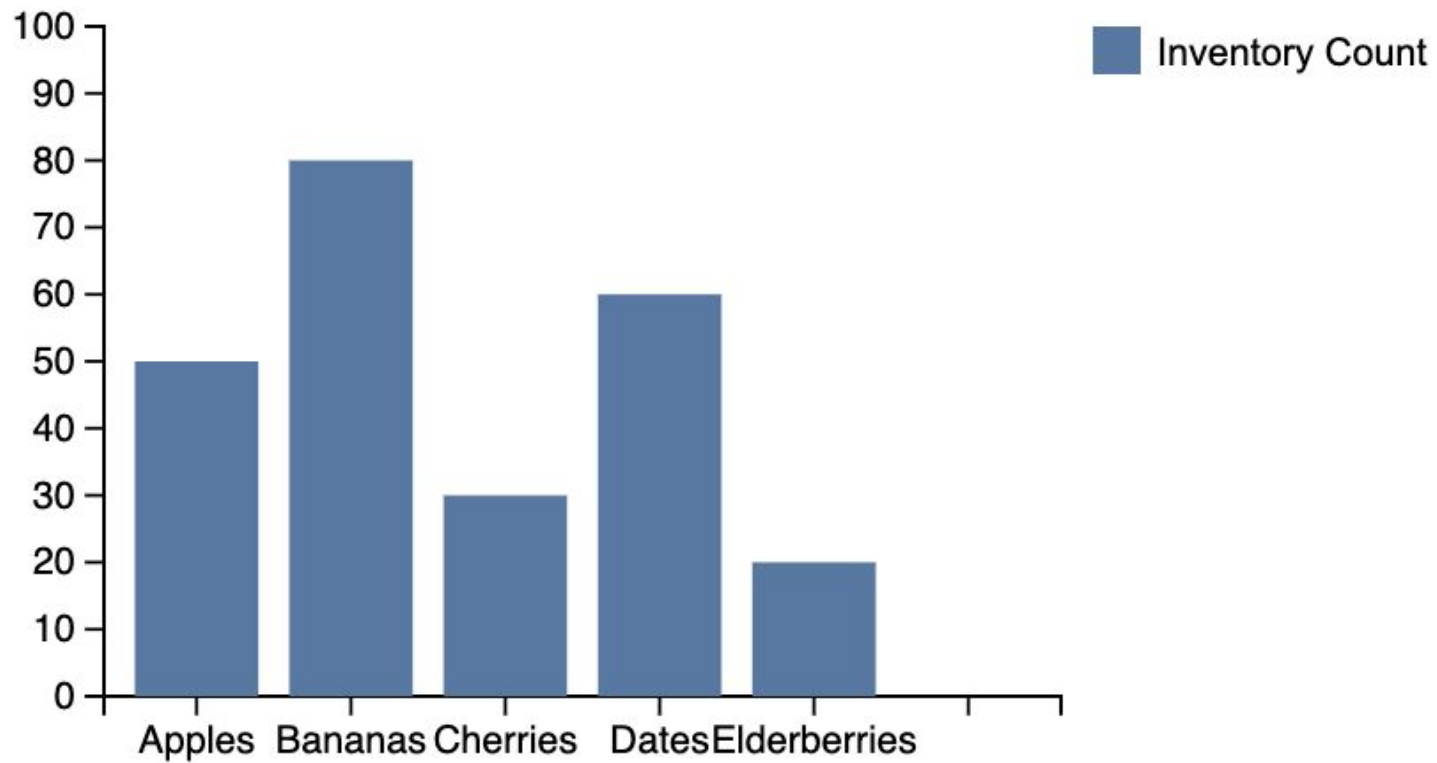
```
// Create Bars (Selection Pattern)  
svgBar.selectAll("rect")  
  .data(data)  
  .enter().append("rect")  
    .attr("class", "bar")  
    .attr("x", d => xScale(d.fruit))  
    .attr("y", d => yScale(d.amount))  
    .attr("width", xScale.bandwidth())  
    .attr("height", d => (height - margin.bottom) - yScale(d.amount));
```

Bottom - Top



## Adding a legend

```
// Add Legend to Bar Chart  
const legend = svgBar.append("g")  
  .attr("transform", `translate(${width - 140}, ${margin.top})`);  
  
legend.append("rect")  
  .attr("width", 15).attr("height", 15).attr("fill", "#4e79a7");  
  
legend.append("text")  
  .attr("x", 20).attr("y", 12)  
  .text("Inventory Count").style("font-size", "12px");
```



# Line graph

```
const svgLine = d3.select("#line-chart").append("svg")
  .attr("width", width)
  .attr("height", height);

// Draw Axes for Line Chart
svgLine.append("g")
  .attr("transform", `translate(0, ${height - margin.bottom})`)
  .call(d3.axisBottom(xScale));
svgLine.append("g")
  .attr("transform", `translate(${margin.left}, 0)`)
  .call(d3.axisLeft(yScale));
```

```
// Line Generator
```

```
const lineGen = d3.line()
```

```
  .x(d => xScale(d.fruit) + xScale.bandwidth() / 2) // Center in the band
```

```
  .y(d => yScale(d.amount));
```

```
// Draw the Path (Single Shape)
```

```
svgLine.append("path")
```

```
  .datum(data) // Use .datum for single objects
```

```
  .attr("class", "line-path")
```

```
  .attr("d", lineGen);
```

