```
tt()
```

# Schedule

1. Scales - recap and examples
2. Show and tell
3. Let’s talk about code
4. D3 Concepts: Axes and Ticks
5. Let’s get started

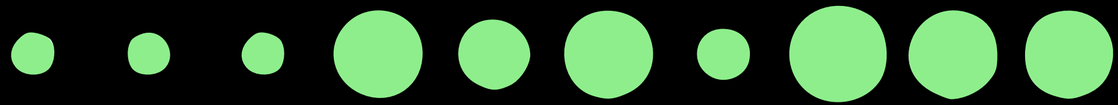
# Schedule

1. **Scales - recap and examples**
2. Show and tell
3. Let’s talk about code
4. D3 Concepts: Axes and Ticks
5. Let’s get started

# Scales - Recap

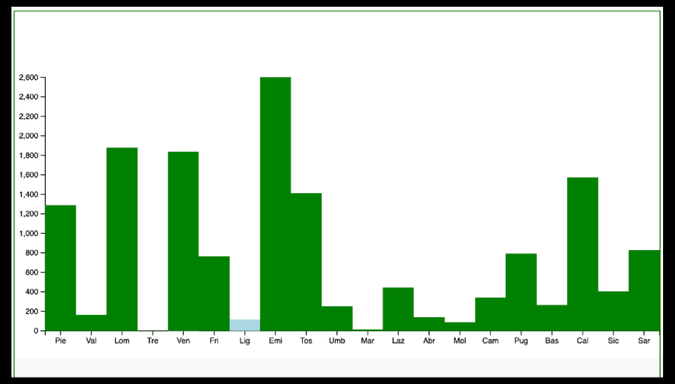
Scales allow to visualize data through:

size



position

(In order to fit in your graphic view)



# Scales

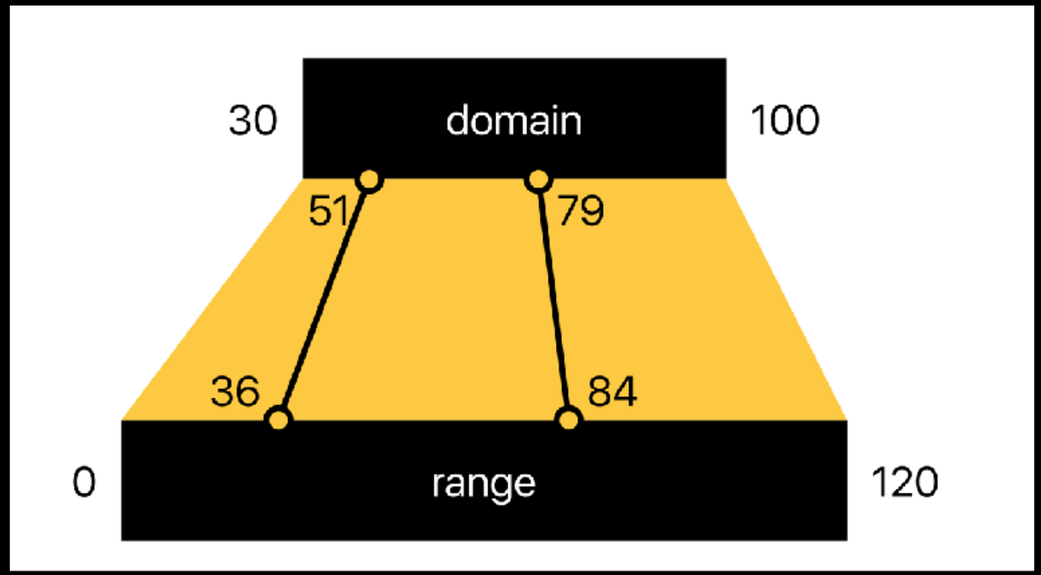
Scales calculate

- how big elements of a graph should be
- where their position

They are **functions**

domain = what you need to show

range = the options you have




We will cover this tomorrow

# Scales

Scales allow to visualize data through:

size

position

and **color**

# Scale examples

## color scale

```
var color = d3.scaleLinear()
  .domain([10, 100]) //data
  .range(['brown', 'steelblue']) //colorrange

color(20) // Output: '#9a3439'
color(50) // Output: ’#7b5167'
```

# Scale examples

## time scale

```
var time = d3.scaleTime()
  .domain([new Date(2000, 0, 1), new Date(2000, 0, 2)])
  .range([0, 960])

time(new Date(2000, 0, 1,  5)) // Output: 200
time(new Date(2000, 0, 1, 16)) // Output: 640
```

# Scale examples

```
var scale = d3.scaleLinear()
  .domain([10, 130])
  .range([0, 960])

scale(-10) // Output: -160, outside range

scale.clamp(true)
scale(-10) // Output: 0, clamped to range
```

## clamping scales

By default, D3 scales will try to use the still return a scaled value if the data you give it is outside the domain. This could be a weird outlier in your dataset for example or just a bug in your API.

If you want your scale to always stay within the range, you can add .clamp(true) to your scale function.

# Choosing scales - Laura's tip

| Welke variabele | Wat wil ik afbeelden | Wat zijn mijn wensen |
|---|---|---|
| | Domain | Range |
| X-coördinaat | 0..11 passagiers | 0..800 px |
| (Point scale) | | Equally spaced points |
| Oppervlakte cirkel | 0..11 passagiers | 0..45 px diameter (?) |
| (scale Sqr) | | |
| Kleur auto | 0..11 passagiers | ['red', 'orange', 'green'] |
| (Scale quantile) | Array of values | 0..3 red, 4..6 orange enz. |

# Schedule

1. Scales - recap and examples
2. **Show and tell**
3. Let’s talk about code
4. D3 Concepts: Axes and Ticks
5. Let’s get started

# Huiswerk

- Werk in tweetallen
- Maak een **point scale** die er voor zorgt dat, hoeveel ritten er ook worden afgebeeld door een auto, ze altijd passen in een rij van 800px
- Maak een **kleurscale** die de user laat zien hoe rendabel de rit is, bijvoorbeeld:

- **Zie: https://www.d3indepth.com/scales/**

**Wij zijn benieuwd!!!!**

# Show and tell

# Schedule

1. Scales - recap and examples
2. Show and tell
3. **Let’s talk about code**
4. D3 Concepts: Axes and Ticks
5. Let’s get started

# Let’s talk about code

# Code quality - readability

1. Code should be readable and understandable to your colleagues (and your teachers)

2. Make sure your code is ES6
   uses the => instead of function()
   your style of coding is uniform

3. Make sure you understand the code - every single row

# Schedule

1. Scales - recap and examples
2. Show and tell
3. Let’s talk about code
4. **D3 Concepts: Axes and Ticks**
5. Let’s get started

# Axes & Ticks

4 different types of Axes:
axisLeft, axisRight, axisTop,
axisBottom

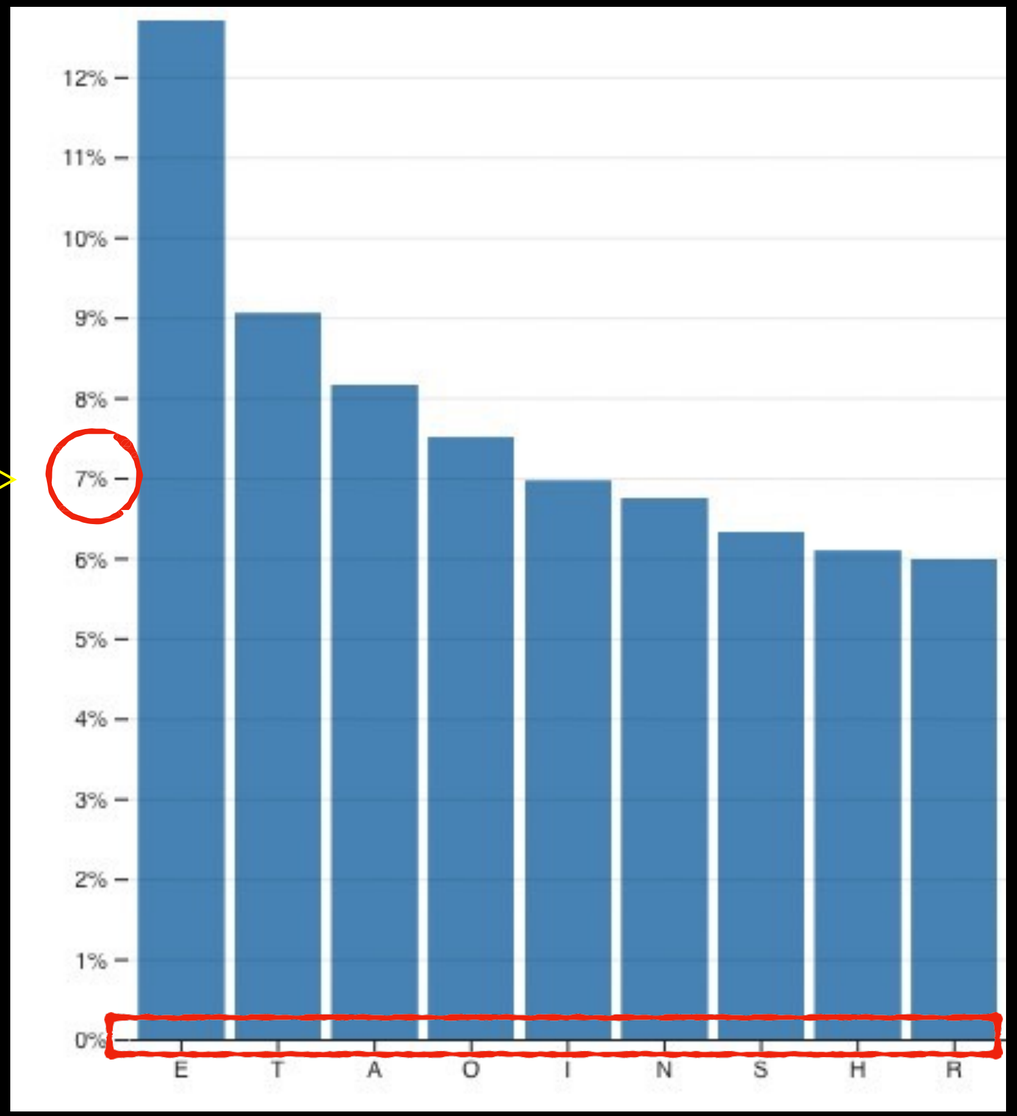
**Axes** refer to **scales**
that should have been
declared previously

```
24  const myAxisBottom = d3
25    .axisBottom(myScaleBand)
26    .tickFormat(d => d)
27
```

**Ticks** are gathered from data,
with accessor functions

Ticks ->

Axes ->



# Axes & Ticks

**Axes** should be **positioned**

Different strategies:

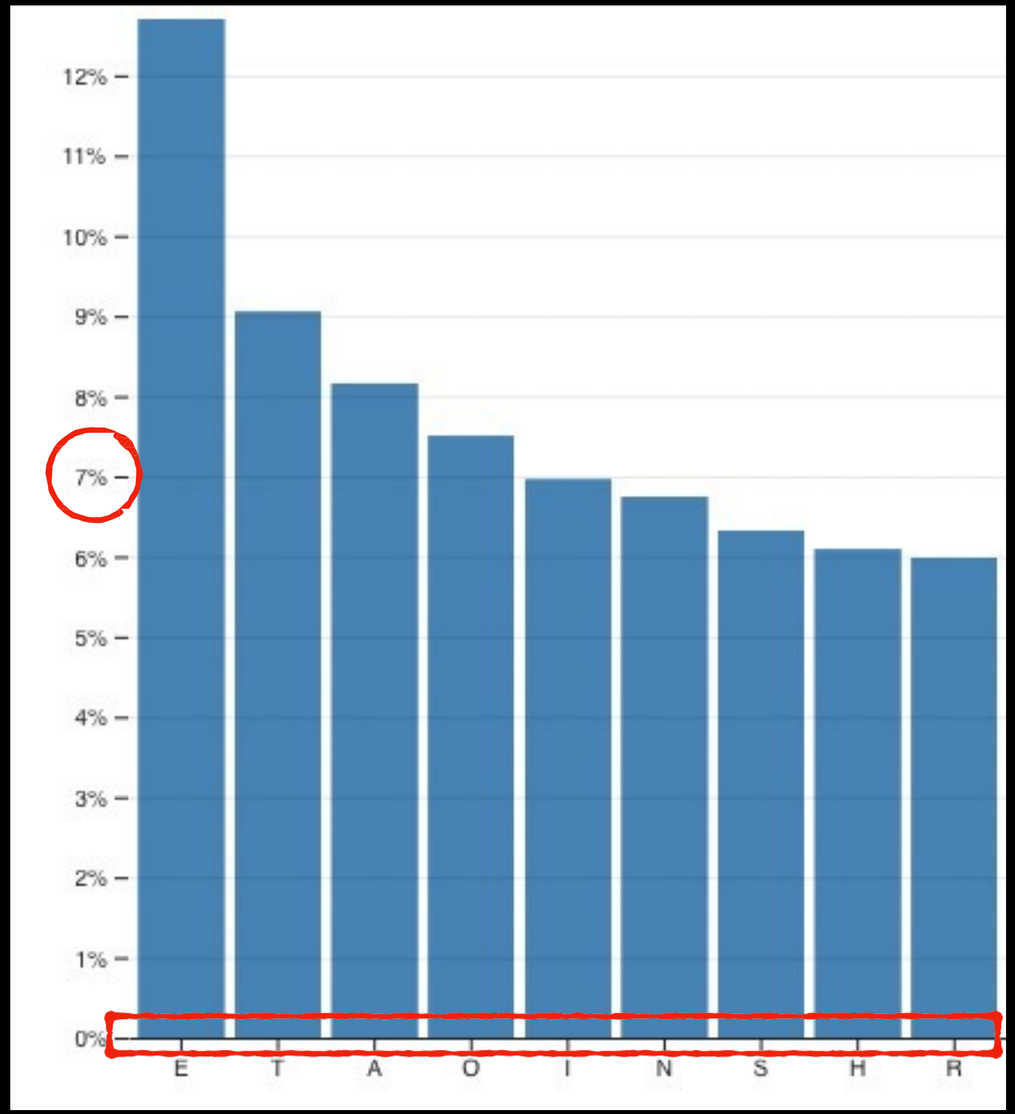
```
27 <svg id="axis">
28   <g transform="translate(50,0)"></g>
29
30 </svg>
31
```

In **HTML**: SVG el. have a *transform* attribute. Its syntax looks like the CSS property - but is different

In **d3**: using .attr(“transform”,…)

Avoid **CSS**

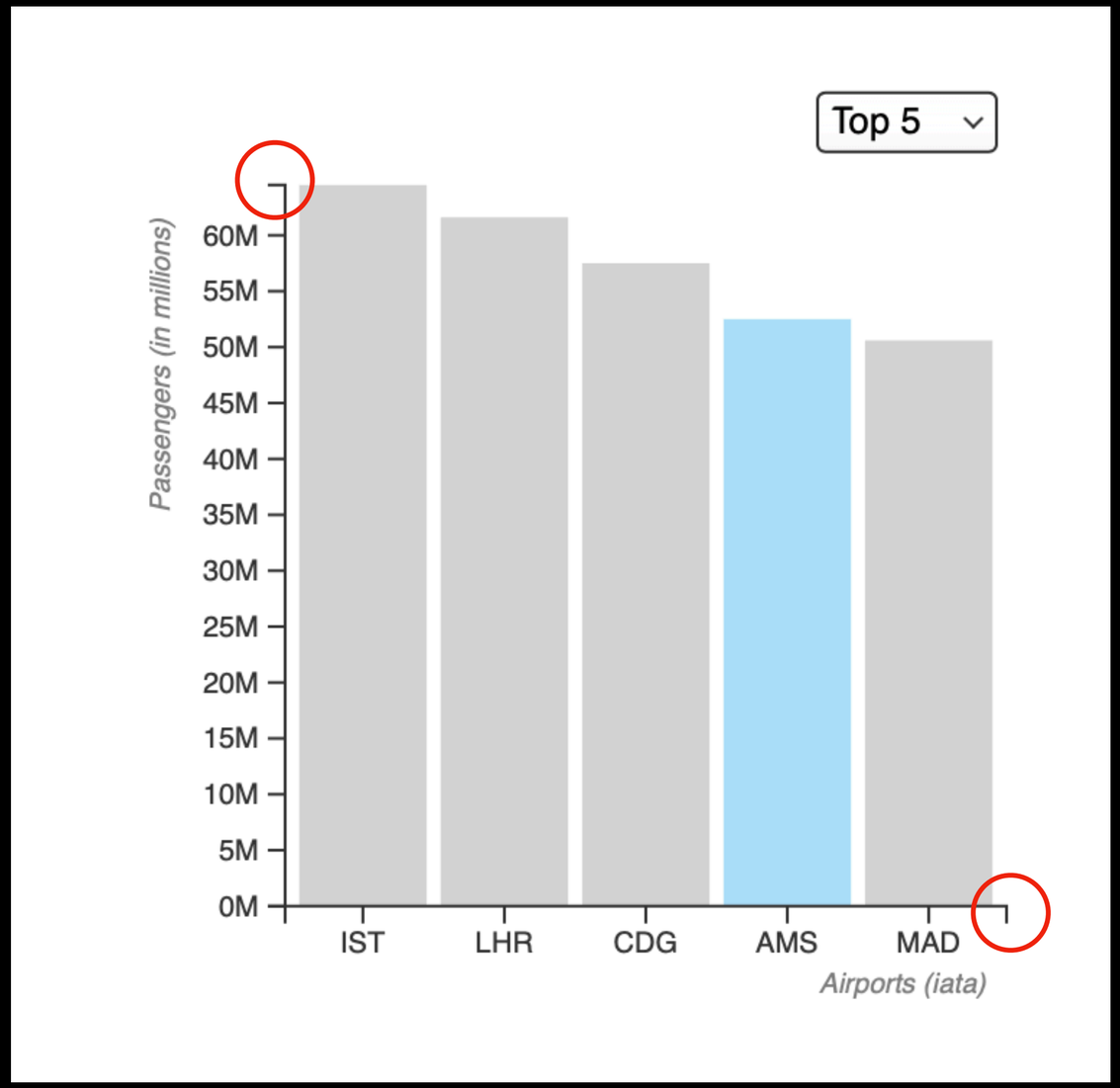
Axis-Y



# Axes

https://codepen.io/Laura_B/pen/WNVgvzd

# Axes




Sometimes your data doesn't fit on axes perfectly and you might see an extra tick without a label.

You can ask D3 to solve this by adding `.nice()` to your axis function

# Schedule

1. Scales - recap and examples
2. Show and tell
3. Let’s talk about code
4. D3 Concepts: Axes and Ticks
5. **Let’s get started**

# Homework… (tips?)

1. Browse through examples on the D3 website (d3js.org)
2. Create a bar chart using D3:
    1. https://observablehq.com/@d3/lets-make-a-bar-chart
    2. https://www.d3indepth.com/selections/
    3. https://www.d3indepth.com/datajoins/
3. Ready? Try an alternative representation of your data(Pie, tooltips… )

```
Uncaught SyntaxError
Unexpected end of input
```