



# Übung Open Data:

## Daten mit D3.js visualisieren

**Termin 6, 26. März 2015**

Dr. Matthias Stürmer und Prof. Dr. Thomas Myrach

Universität Bern, Institut für Wirtschaftsinformatik

Abteilung Informationsmanagement

Forschungsstelle Digitale Nachhaltigkeit

# Opendata.ch/2015

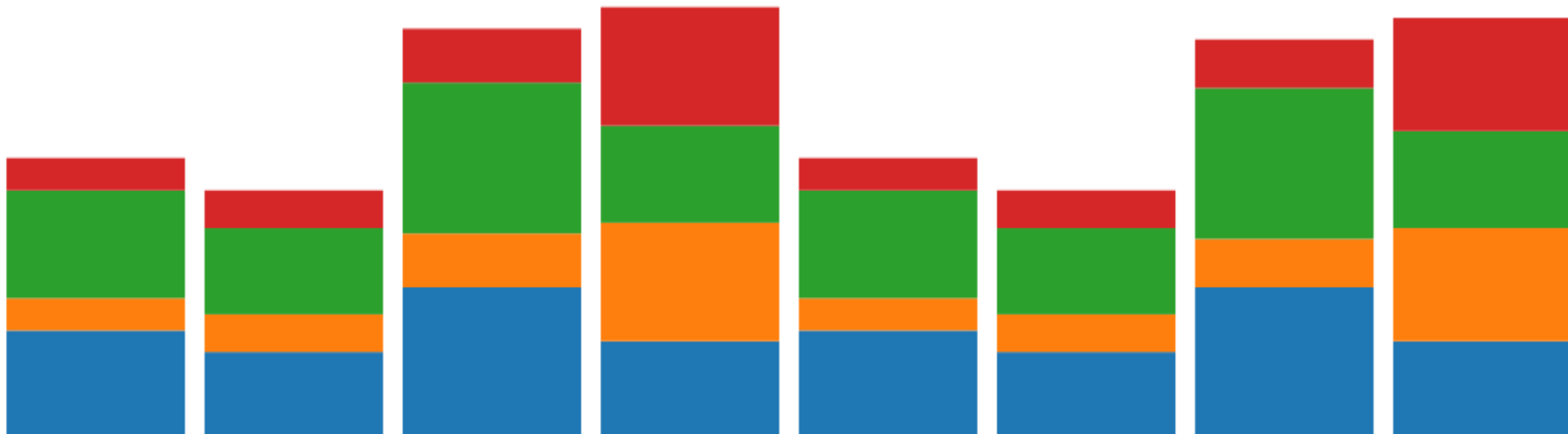
## 1. Juli, Uni Bern

**Nationale Open Data Konferenz mit Plenum und Fachsessions**  
**Mittwoch, 1. Juli 2015 im Hauptgebäude der Universität Bern**

- > **Gratis-Tickets** für alle Teilnehmende der Open Data Vorlesung
- > **Open Data Hack Room** von 9h bis 17h im Raum 105 (HG)
  - 15 Tische mit ca. 50 Sitzplätzen, einmal pro Stunde App-Demo,
  - Eigene App weiterentwickeln, Networking mit anderen Entwicklern, Unternehmern, möglichen Arbeitgebern etc.
  - Free Food, Getränke, Strom und Internet
  - Anmeldung bis 15. Juni 2015 bei [mirjam.laederach@iwi.unibe.ch](mailto:mirjam.laederach@iwi.unibe.ch)

# Agenda

1. **Einen Balken mit div zeichnen**
2. Mehrere Balken zeichnen
3. Flexibel Daten hinzufügen

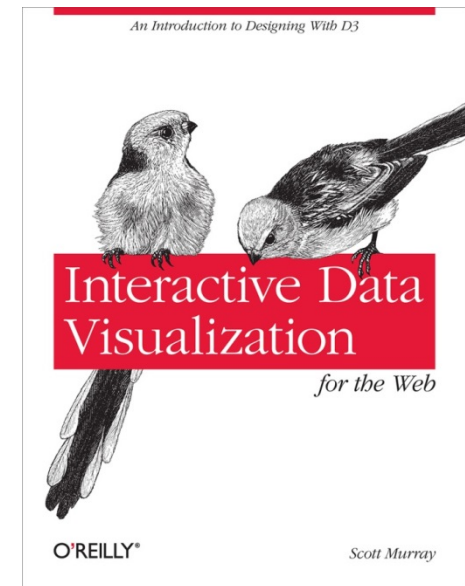


# Interactive Data Visualization for the Web

## Chapter 6. Drawing with Data:

> *It's time to start drawing with data.*

> <http://chimera.labs.oreilly.com/books/123000000345/ch06.html>



# Drawing divs

```
<div style="display: inline-block;  
    width: 20px;  
    height: 75px;  
    background-color: teal;"></div>
```

zeichnet:



## CSS class "bar"

In das CSS Stylesheet einfügen:

```
div.bar {  
  display: inline-block;  
  width: 20px;  
  height: 75px;  
  background-color: teal;  
}
```

Von jetzt an divs einfach formatieren:

```
<div class="bar"></div>
```

# Setting Attributes

```
<p class="caption">  
<select id="country">  

```

Diese HTML-Elemente enthalten 5 Attribute:

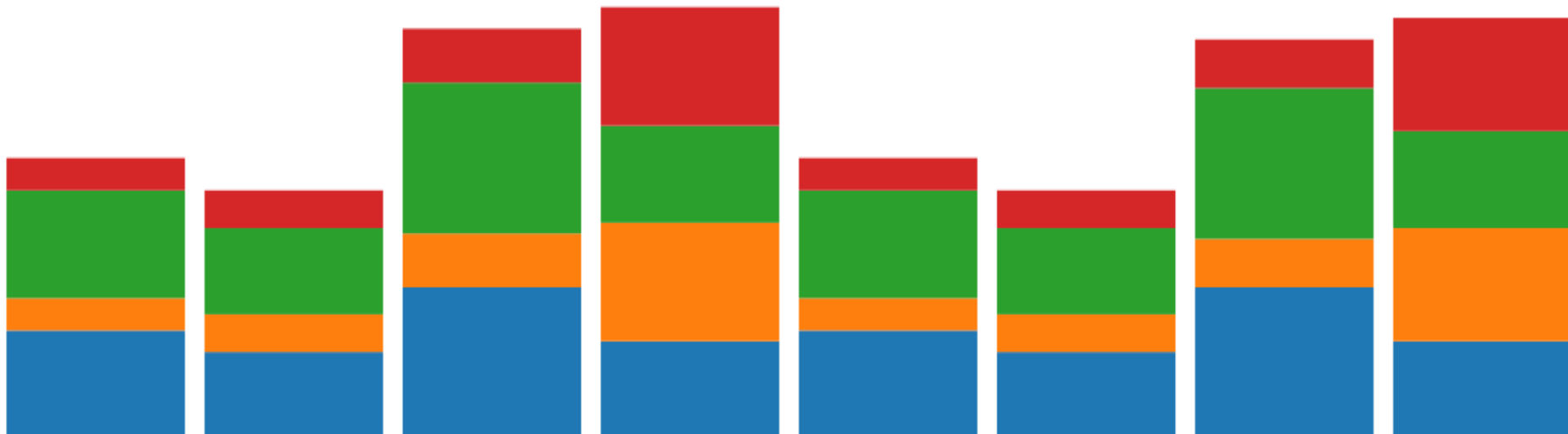
Attribute	Value
<code>class</code>	<code>caption</code>
<code>id</code>	<code>country</code>
<code>src</code>	<code>logo.png</code>
<code>width</code>	<code>100px</code>
<code>alt</code>	<code>Logo</code>

Dem Attribut "class" den Wert "bar" zuweisen:

```
.attr("class", "bar")
```

# Agenda

1. Einen Balken mit div zeichnen
2. **Mehrere Balken zeichnen**
3. Flexibel Daten hinzufügen





# Using D3.js API Reference

Den Befehl `selection.attr()` in der API Referenz nachschlagen:

# `selection.attr(name[, value])`

If *value* is specified, sets the attribute with the specified name to the specified value on all selected elements. If *value* is a constant, then all elements are given the same attribute value; otherwise, if *value* is a function, then the function is evaluated for each selected element (in order), being passed the current datum `d` and the current index `i`, with the `this` context as the current DOM element. The function's return value is then used to set each element's attribute. A null value will remove the specified attribute.

If *value* is not specified, returns the value of the specified attribute for the first non-null element in the selection. This is generally useful only if you know that the selection contains exactly one element.

The specified *name* may have a namespace prefix, such as `xlink:href`, to specify an "href" attribute in the XLink namespace. By default, D3 supports `svg`, `xhtml`, `xlink`, `xml`, and `xmlns` namespaces. Additional namespaces can be registered by adding to `d3.ns.prefix`.

*name* can also be an Object of *name* and *value* attributes.

Link: <https://github.com/mbostock/d3/wiki/Selections#attr>

# D3.js Bar Example

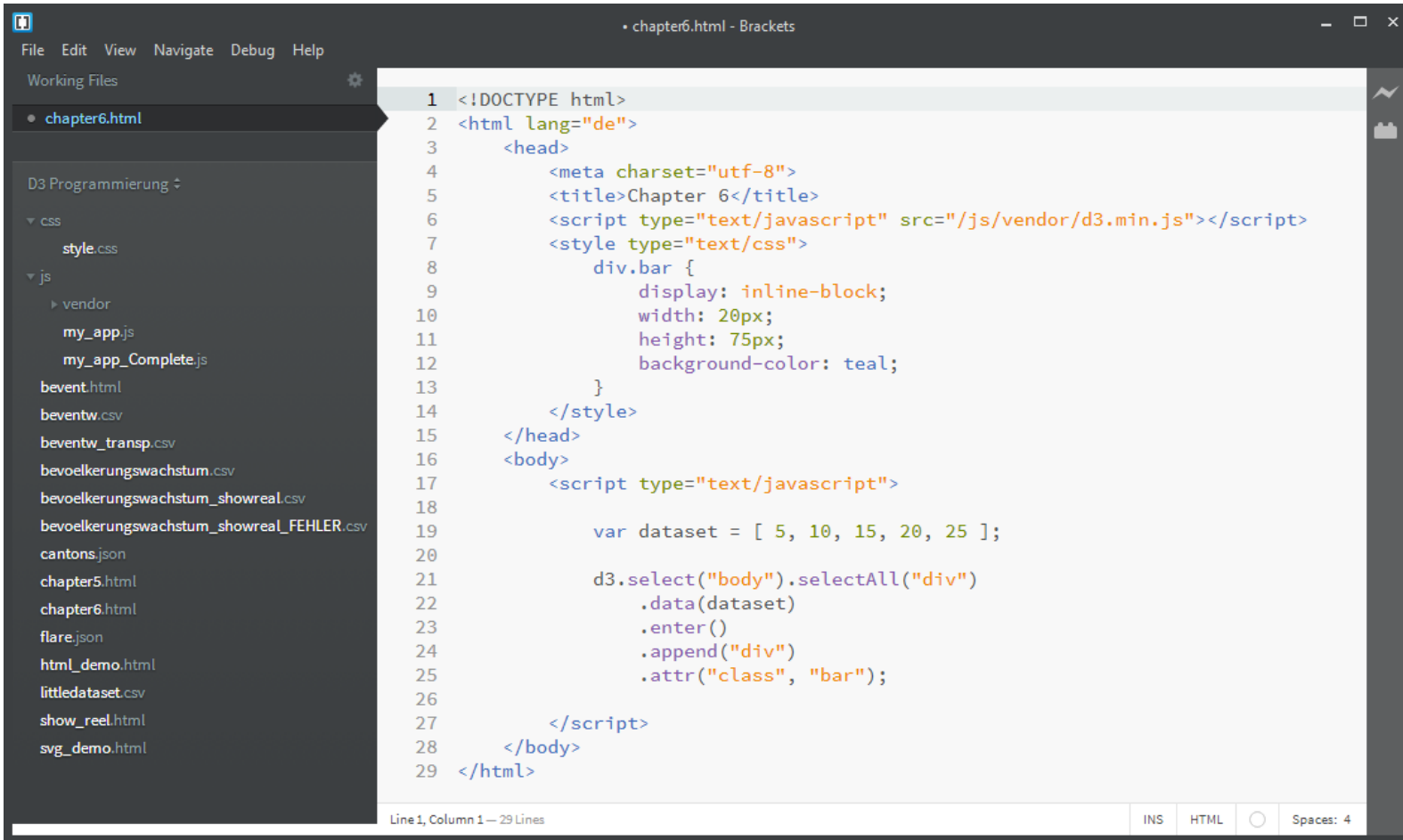
```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Chapter 6</title>
    <script type="text/javascript" src="/js/vendor/d3.min.js"></script>
    <style type="text/css">
      div.bar {
        display: inline-block;
        width: 20px;
        height: 75px;
        background-color: teal;
      }
    </style>
  </head>
  <body>
    <script type="text/javascript">

      var dataset = [ 5, 10, 15, 20, 25 ];

      d3.select("body").selectAll("div")
        .data(dataset)
        .enter()
        .append("div")
        .attr("class", "bar");

    </script>
  </body>
</html>
```

# D3.js Bar Example



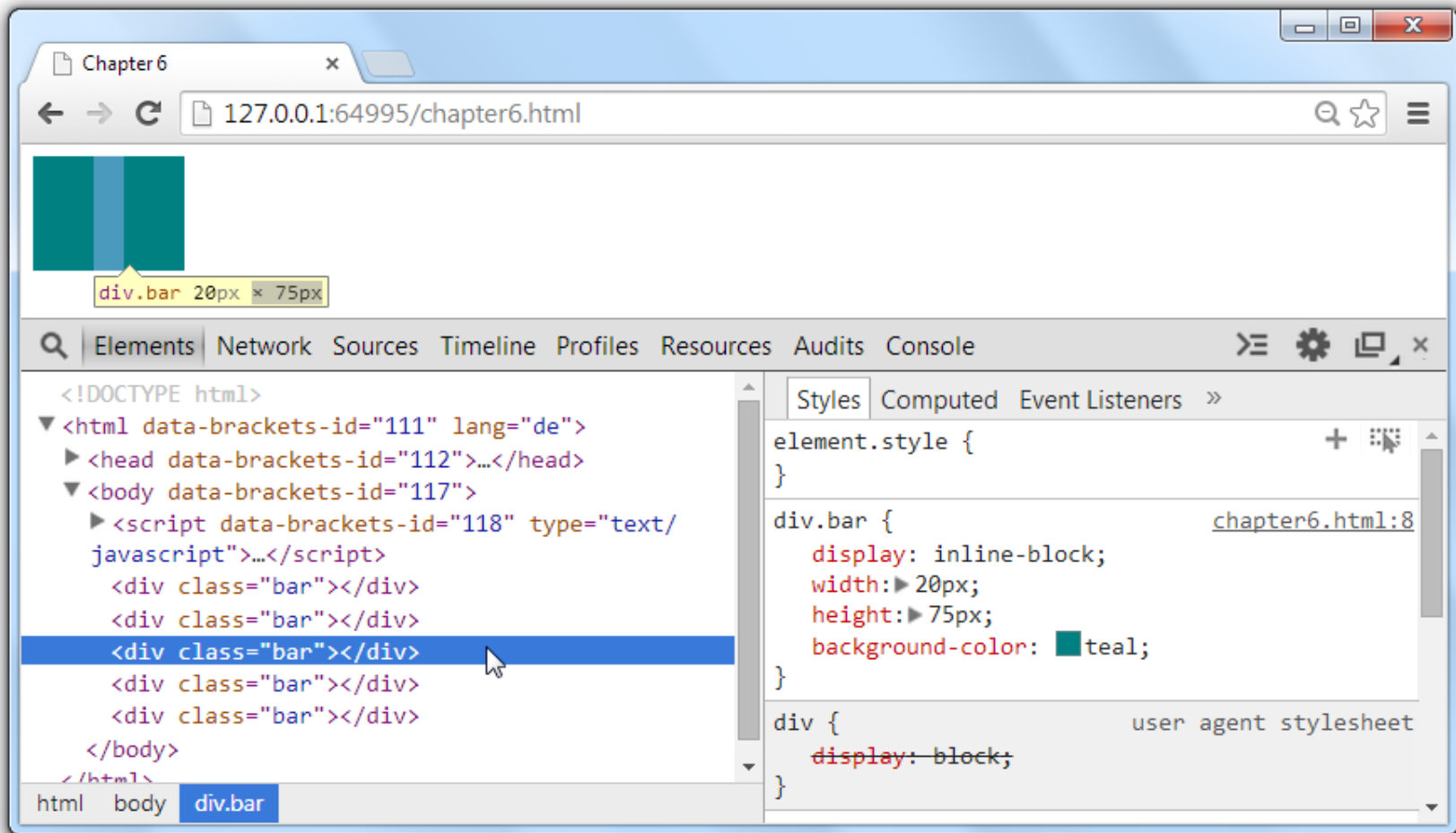
The screenshot shows a code editor window titled "chapter6.html - Brackets". The editor displays HTML and JavaScript code for a D3.js bar chart. The code includes a head section with meta tags, a title, and a script tag for D3.js. The body section contains a script that defines a dataset and uses D3.js to create a bar chart with teal bars.

```
1 <!DOCTYPE html>
2 <html lang="de">
3   <head>
4     <meta charset="utf-8">
5     <title>Chapter 6</title>
6     <script type="text/javascript" src="/js/vendor/d3.min.js"></script>
7     <style type="text/css">
8       div.bar {
9         display: inline-block;
10        width: 20px;
11        height: 75px;
12        background-color: teal;
13      }
14    </style>
15  </head>
16  <body>
17    <script type="text/javascript">
18      var dataset = [ 5, 10, 15, 20, 25 ];
19
20      d3.select("body").selectAll("div")
21        .data(dataset)
22        .enter()
23        .append("div")
24        .attr("class", "bar");
25
26    </script>
27  </body>
28 </html>
```

Line 1, Column 1 — 29 Lines

INS HTML  Spaces: 4

# D3.js Bar Example



The screenshot shows a web browser window with a single teal bar chart. The browser's developer tools are open, displaying the HTML structure and the CSS styles for the selected bar.

**HTML Structure:**

```
<!DOCTYPE html>
<html data-brackets-id="111" lang="de">
  <head data-brackets-id="112">...</head>
  <body data-brackets-id="117">
    <script data-brackets-id="118" type="text/javascript">...</script>
    <div class="bar"></div>
    <div class="bar"></div>
    <div class="bar"></div>
    <div class="bar"></div>
    <div class="bar"></div>
  </body>
</html>
```

**CSS Styles:**

```
element.style {
}

div.bar {
  display: inline-block;
  width: 20px;
  height: 75px;
  background-color: teal;
}

div {
  display: block;
}
```

# D3.js Bar Example

Balken mit `div` aus dem Array generieren:

```
var dataset = [ 5, 10, 15, 20, 25 ];  
  
d3.select("body").selectAll("div")  
  .data(dataset)  
  .enter()  
  .append("div")  
  .attr("class", "bar");
```

# Setting Styles

Die Höhe eines Balkens kann mit `height` festgelegt werden:

```
<div style="height: 75px;"></div>
```

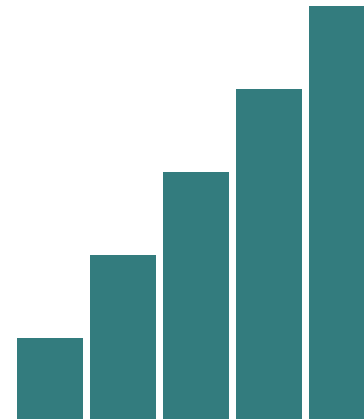
Im D3.js Code den Befehl `.style("height"...)` anfügen:

```
d3.select("body").selectAll("div")
  .data(dataset)
  .enter()
  .append("div")
  .attr("class", "bar")
  .style("height", function(d) {return d + "px";});
```

# Adding Space

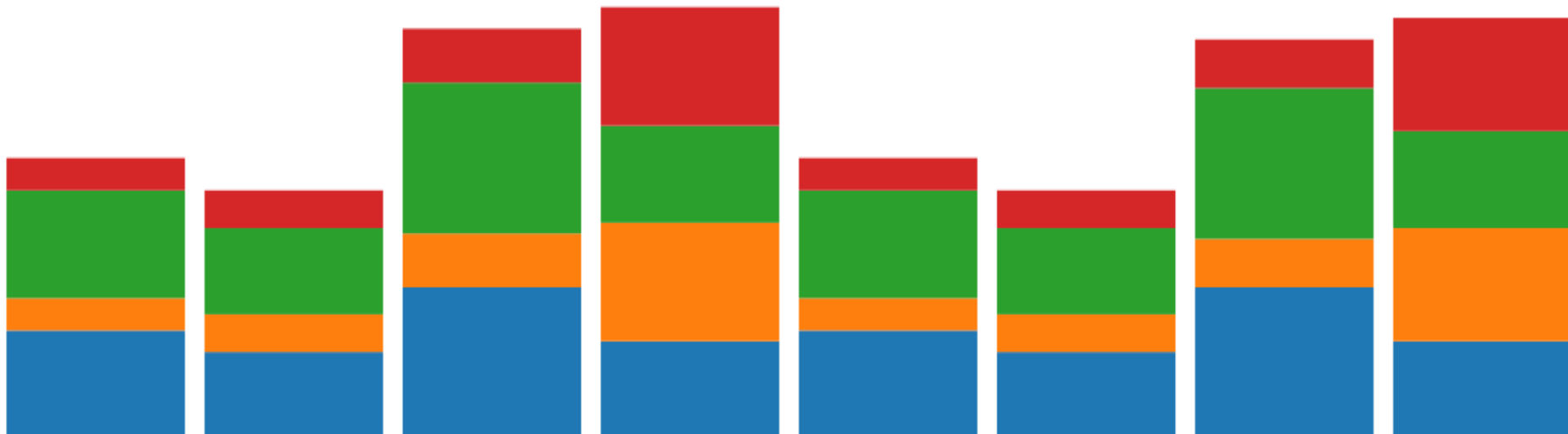
```
div.bar {  
  display: inline-block;  
  width: 20px;  
  height: 75px;  
  background-color: teal;  
  margin-right: 2px;  
}
```

zeichnet:



# Agenda

1. Einen Balken mit div zeichnen
2. Mehrere Balken zeichnen
3. **Flexibel Daten hinzufügen**



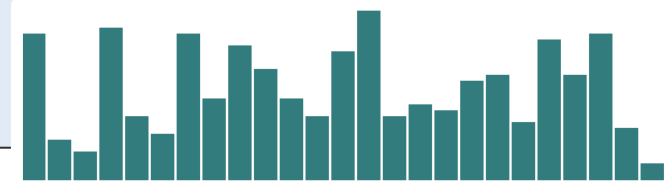


# Flexibilität von .data()

Mehr Daten hinzufügen und Höhe der Balken verstärken:

```
var dataset = [ 25, 7, 5, 26, 11, 8, 25, 14, 23, 19,
                14, 11, 22, 29, 11, 13, 12, 17, 18, 10,
                24, 18, 25, 9, 3 ];

d3.select("body").selectAll("div")
  .data(dataset)
  .enter()
  .append("div")
  .attr("class", "bar")
  .style("height", function(d) {
    var barHeight = d * 50;
    return barHeight + "px";
  });
```



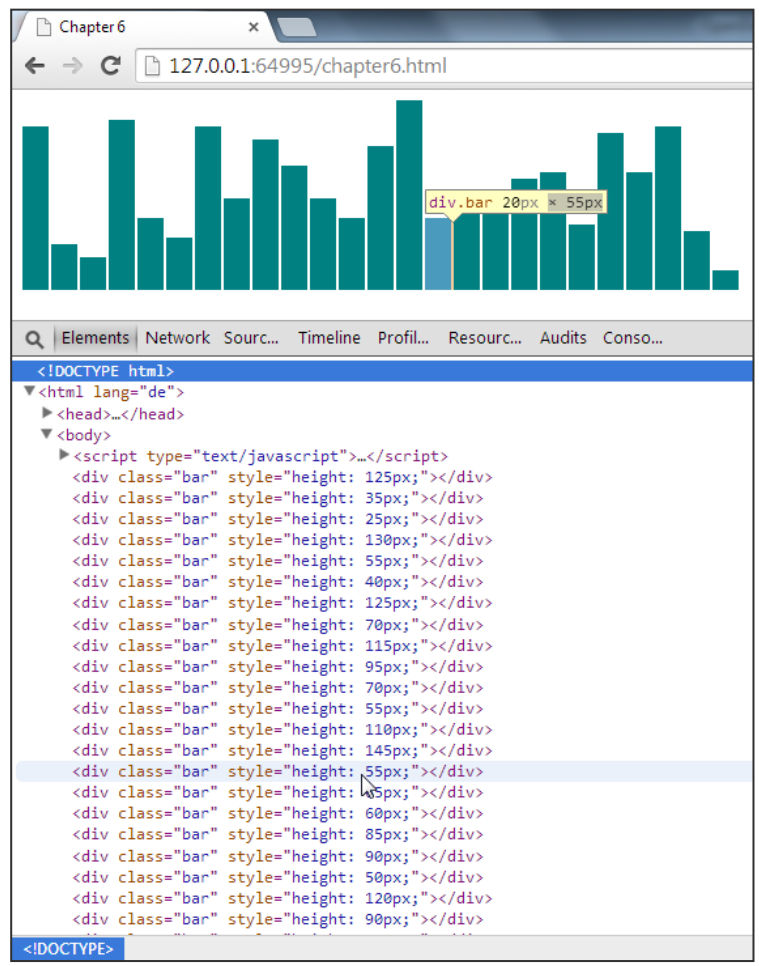
# Flexibilität von .data()

```

chapter6.html - Brackets
File Edit View Navigate Debug Help

1 <!DOCTYPE html>
2 <html lang="de">
3   <head>
4     <meta charset="utf-8">
5     <title>Chapter 6</title>
6     <script type="text/javascript" src="/js/vendor/d3.min.js"></script>
7     <style type="text/css">
8       div.bar {
9         display: inline-block;
10        width: 20px;
11        height: 75px;
12        background-color: teal;
13        margin-right: 2px;
14      }
15    </style>
16  </head>
17  <body>
18    <script type="text/javascript">
19
20      var dataset = [ 25, 7, 5, 26, 11, 8, 25, 14, 23, 19,
21                    14, 11, 22, 29, 11, 13, 12, 17, 18, 10,
22                    24, 18, 25, 9, 3 ];
23
24      d3.select("body").selectAll("div")
25        .data(dataset)
26        .enter()
27        .append("div")
28        .attr("class", "bar")
29        .style("height", function(d) {
30          var barHeight = d * 5;
31          return barHeight + "px";
32        });
33
34    </script>
35  </body>
36 </html>

```



# Zufallsgenerator Math.random()

Zufallszahlen generieren:

```
var dataset = [];  
for (var i = 0; i < 25; i++) {  
  var newNumber = Math.random() * 30;  
  dataset.push(newNumber);  
}
```

