

Fachhochschule Köln
Cologne University of Applied Sciences

DRAWING WITH DATA

Interactive Data Visualization for the Web
- Scott Murray -

Ein Workshop im Rahmen des Moduls „Visualistik“ im SS'15
von Uwe Müsse und Sheree May Saßmannshausen

INHALT

- Phase 1: Basics - Balkendiagramme
- Phase 2: SVG - Circles
- Phase 3: SVG - Balken + Labels
- Phase 4: SVG - Scatterplot

PHASE I

Balken erzeugen

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

```
d3.select("body")  
  .append("div")  
  .attr("class", "bar");
```

select(): Referenz auf das ausgewählte Element

append(): fügt ein Element an die zuvor ausgewählte Referenz

attr(): setzt ein HTML Attribut mit einem Wert für das zuvor ausgewählte Element

PHASE I

Balken erzeugen

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

```
d3.select("body")  
  .append("div")  
  .attr("class", "bar");
```



Abb. I: Balkendiagramm

PHASE I

Balken erzeugen

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

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

selectAll(): wählt alle Elemente aus, auch wenn sie noch nicht existieren

data(): lädt das dataset

enter(): erzeugt Platzhalter für die Selektion und die ausgewählten Daten

PHASE I

Anwendung des Arrays

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

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

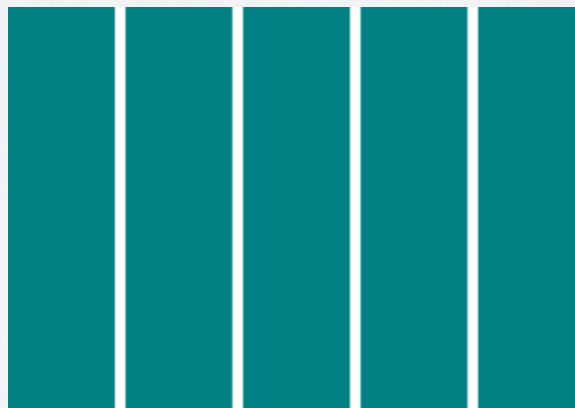


Abb. 2: Balkendiagramm mit 5 Balken

PHASE I

Anwendung des Arrays

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

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

style(key,value): gibt Attribute mit jeweiligem Wert bzgl. der Gestaltung an

PHASE I

Anpassung der Balkenhöhe

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

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



Abb. 3: aufsteigendes Balkendiagramm

PHASE I

Anpassung der Balkenhöhe

```
dataset = [];  
for (var i = 0; i < 10; i++){  
    var newNumber = Math.floor(Math.random() * MAX_VALUE);  
    dataset.push(newNumber);  
}
```

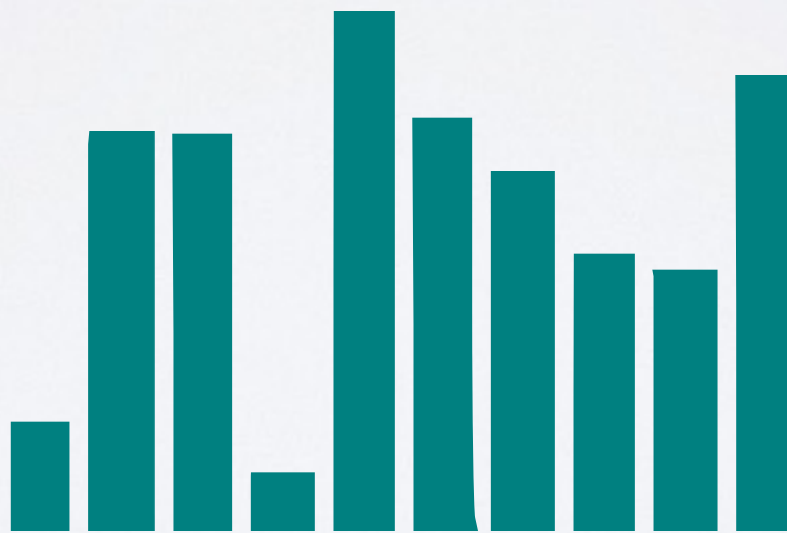
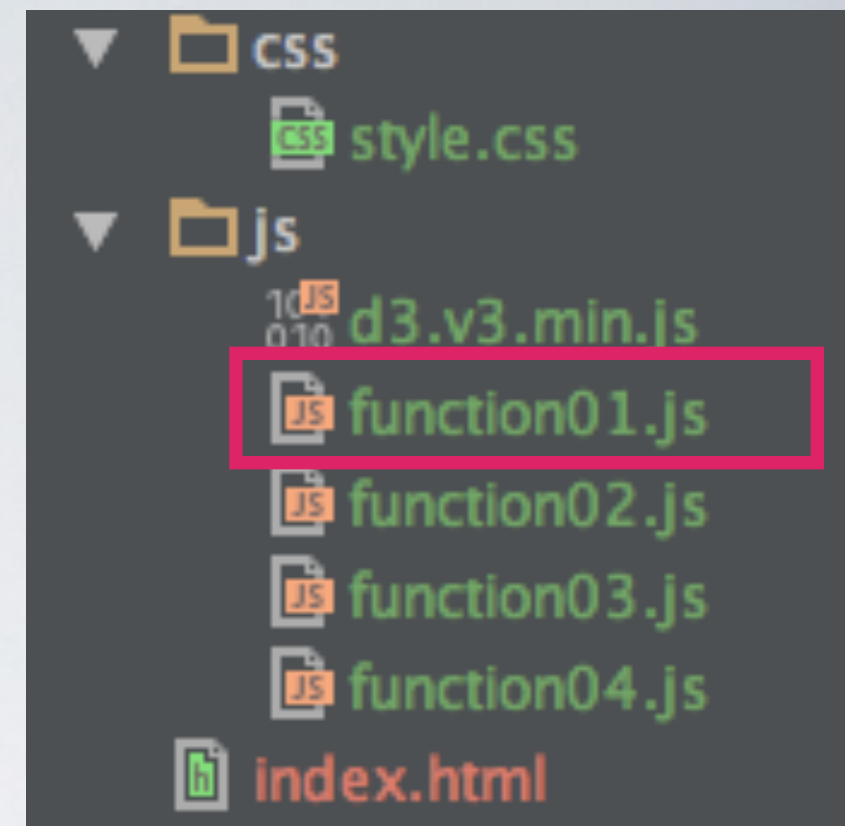


Abb. 4: Balkendiagramm mit zufälligen Werten

PHASE I



TODO

<http://bit.ly/1QjEIEI>

PHASE 2

SVG's erzeugen

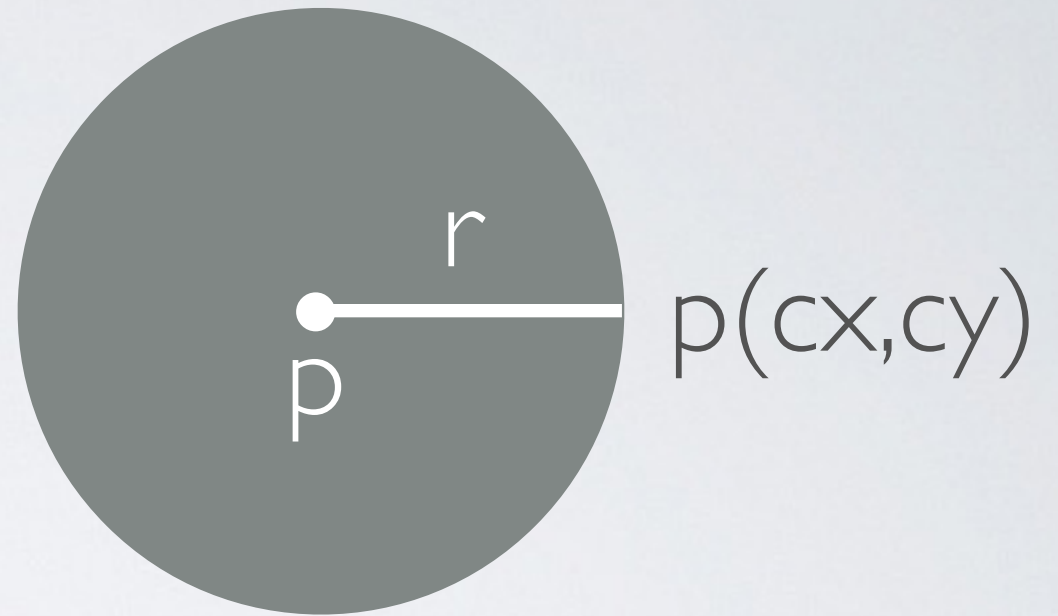
```
var svg = d3.select("#t2_p1")  
  .append("svg")  
  .attr("width", width)  
  .attr("height", height);
```

- Angabe der Breite (width)
- Angabe der Höhe (height)

PHASE 2

SVG's erzeugen

```
svg.append("circle")  
  .attr("cx", "50")  
  .attr("cy", "20")  
  .attr("r", radius);
```



- **cx:** Translation des Mittelpunkts entlang der X-Achse
- **cy:** Translation des Mittelpunkts entlang der Y-Achse
- **r:** Radius

PHASE 2

Positionierung

```
svg.selectAll("circle")  
  .data(dataset)  
  .enter()  
  .append("circle")  
  .attr("cx", function(d, i){  
    return (i * 2 * (radius + 10)) + (radius + 10);  
  })  
  .attr("cy", height/2)  
  .attr("r", radius);
```

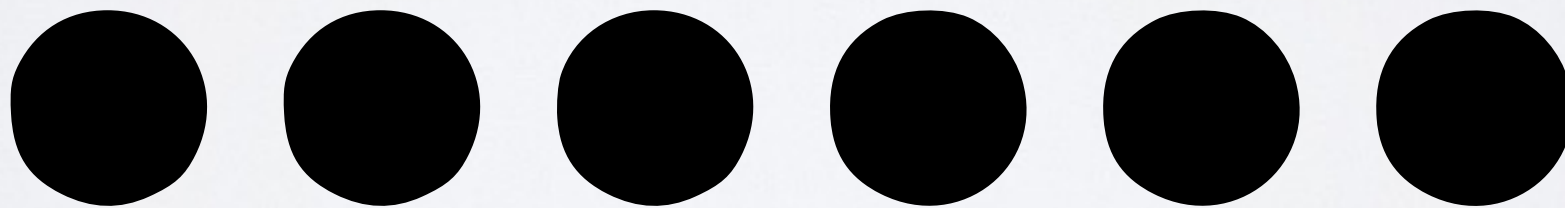


Abb. 5. aneinandergereihte Kreise

PHASE 2

Radius anpassen

```
svg.selectAll("circle")  
  .data(dataset)  
  .enter()  
  .append("circle")  
  .attr("cx", function(d, i){  
    return (i * 2 * (radius + 10)) + (radius + 10);  
  })  
  .attr("cy", height/2)  
  .attr("r", function(d){  
    return d + "px";  
  });
```

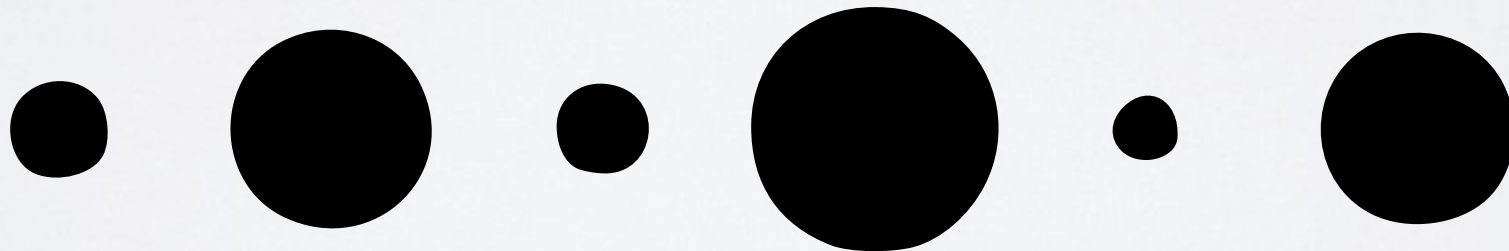
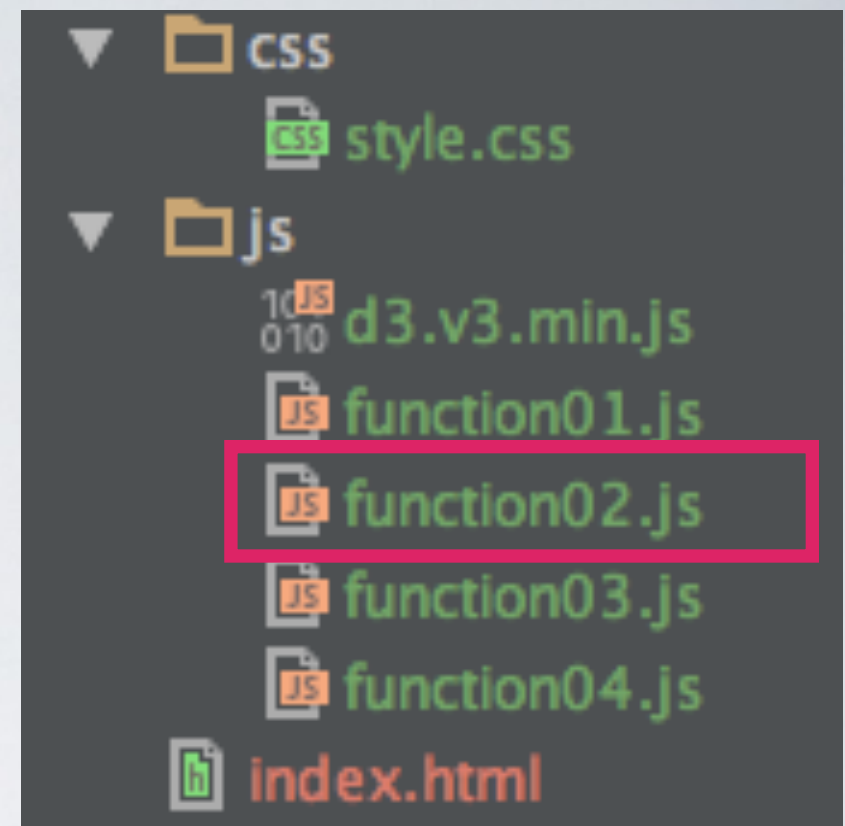


Abb. 6. Kreise mit Radius der entsprechenden Werte des Arrays

PHASE 2



TODO

PHASE 3

Bar charts als SVG

```
svg_one.selectAll("rect")
  .data(dataset)
  .enter()
  .append("rect")
  .attr("x", function (d, i) {
    return (i * (width / dataset.length));
  })
  .attr("y", 0)
  .attr("width", function (d) {
    return (width / dataset.length) - barpadding;
  })
  .attr("height", function (d) {
    return d;
  });
```



Abb. 7. Balkendiagramm als SVG

PHASE 3

Anpassung an die Baseline

```
svg_two.selectAll("rect")
  .data(dataset)
  .enter()
  .append("rect")
  .attr("x", function (d, i) {
    return (i * (width / dataset.length));
  })
  .attr("y", function (d) {
    return height - d;
  })
  .attr("width", function (d) {
    return (width / dataset.length) - barpadding;
  })
  .attr("height", function (d) {
    return d;
  });
```

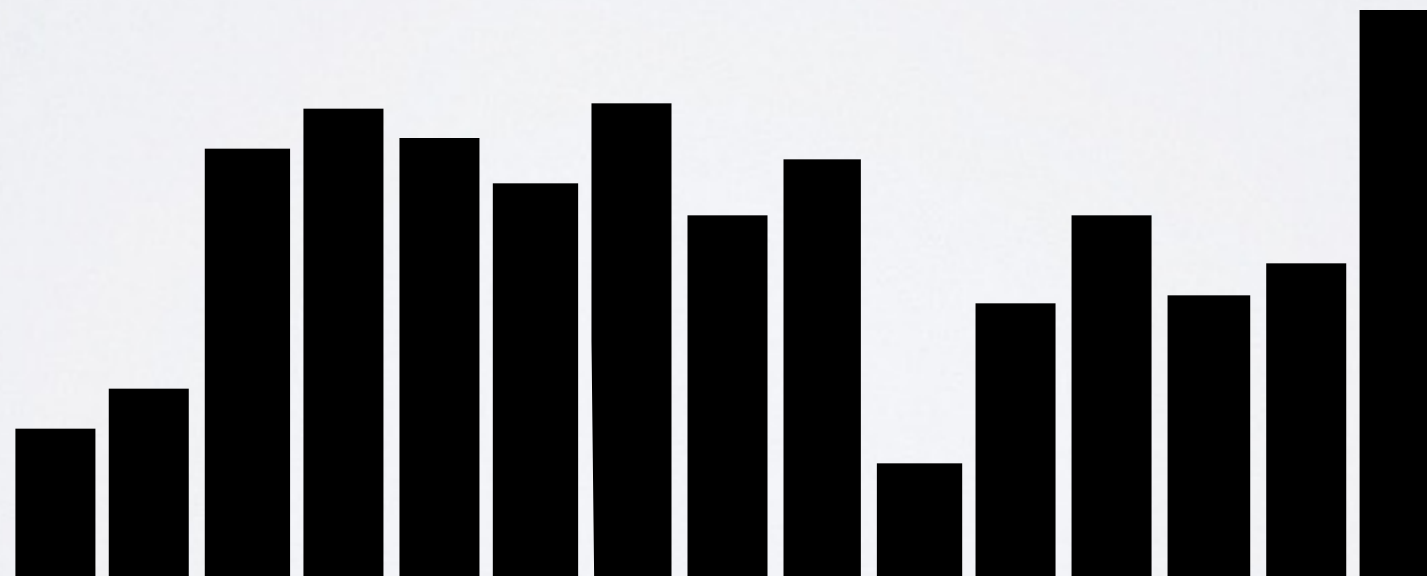


Abb. 8. Balkendiagramm mit Anpassung an die Baseline

PHASE 3

Farbverlauf

```
.append("rect")
.attr("x", function (d, i) {
    return (i * (width / dataset.length));
})
.attr("y", function (d) {
    return height - d;
})
.attr("width", function (d) {
    return (width / dataset.length) - barpadding;
})
.attr("height", function (d) {
    return d;
})
.style("fill", function (d) {
    return "rgb(0, 0, " + (d * 2) + ")";
});
```

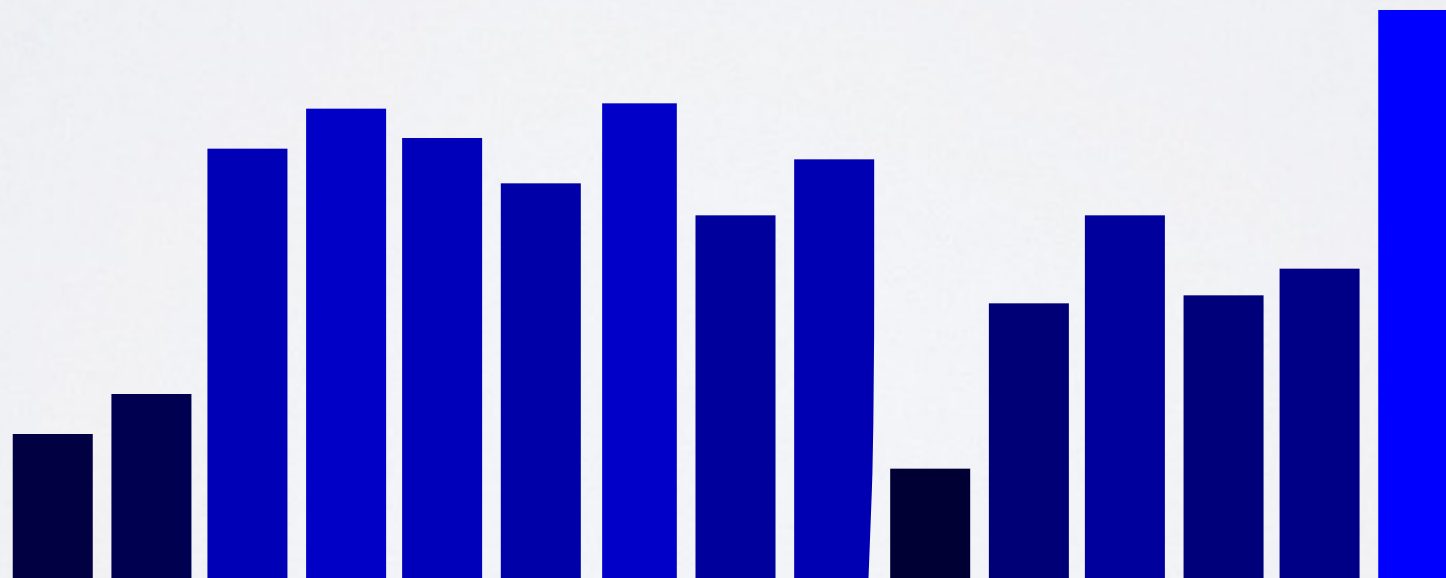


Abb. 9. Balkendiagramm mit Farbverlauf

PHASE 3

Labels

```
svg_four.selectAll("text")
  .data(dataset)
  .enter()
  .append("text")
  .text(function (d) {
    return d;
  })
  .attr("x", function (d, i) {
    return (i * (width / dataset.length) + (barWidth / 2));
  })
  .attr("y", function (d) {
    return height - d + 14;
  })
```

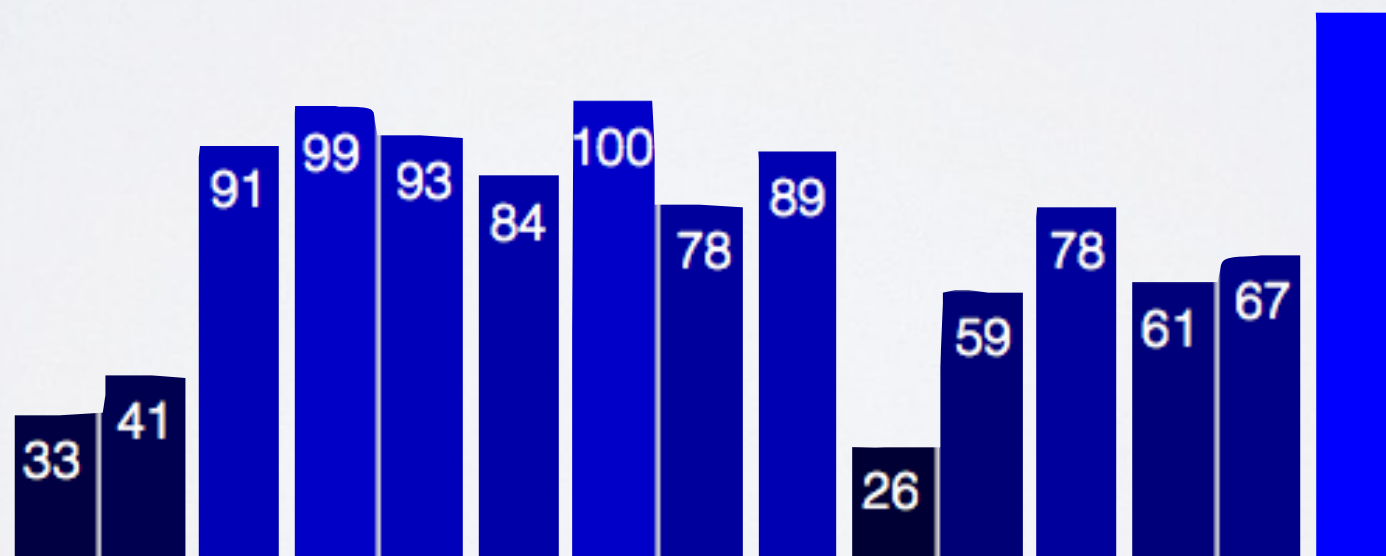


Abb. 10. Balkendiagramm mit Labels

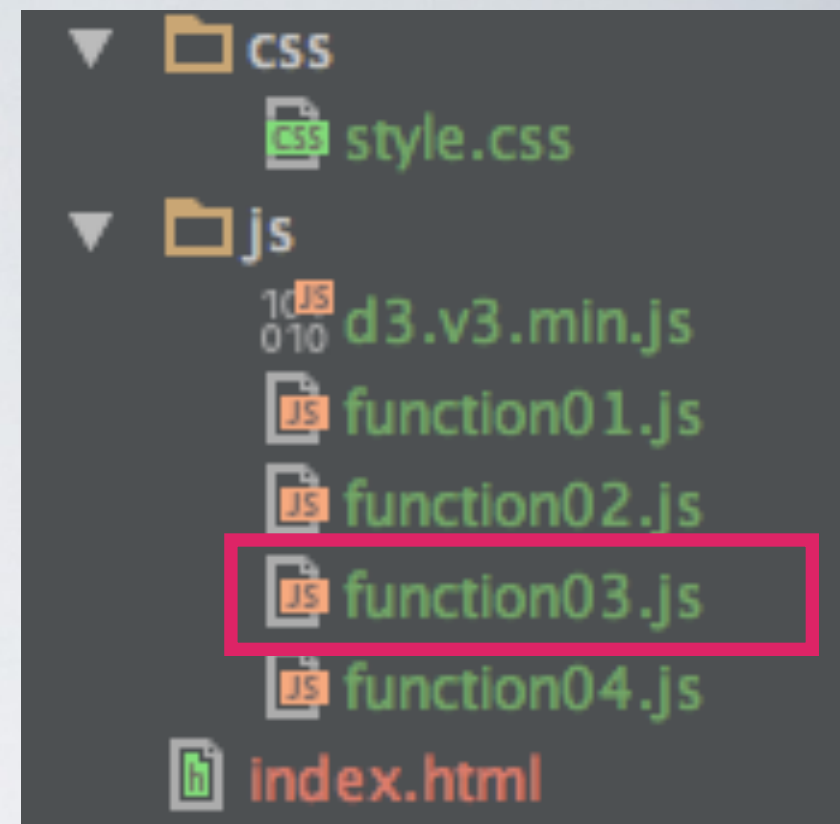
PHASE 3

Bar charts als SVG

- Schreibweise der Attribute als Array

```
svg.selectAll("rect")
  .data(dataset)
  .enter()
  .append("rect")
  .attr({
    x: function(d, i) { return i * (w / dataset.length); },
    y: function(d) { return h - (d * 4); },
    width: w / dataset.length - barPadding,
    height: function(d) { return d * 4; },
    fill: function(d) { return "rgb(0, 0, " + (d * 10) + ")"; }
  });
```


PHASE 3



TODO

PHASE 4

Scatterplot

```
var dataset = [  
  [5, 20],  
  [480, 90],  
  [250, 50],  
  [100, 80],  
  [330, 95],  
  [...]  
];
```

```
svg.selectAll("circle")  
  .data(dataset)  
  .enter()  
  .append("circle")  
  .attr("cx", function(d){  
    return d[0];  
  })  
  .attr("cy", function(d){  
    return d[1];  
  })  
  .attr("r", radius);
```

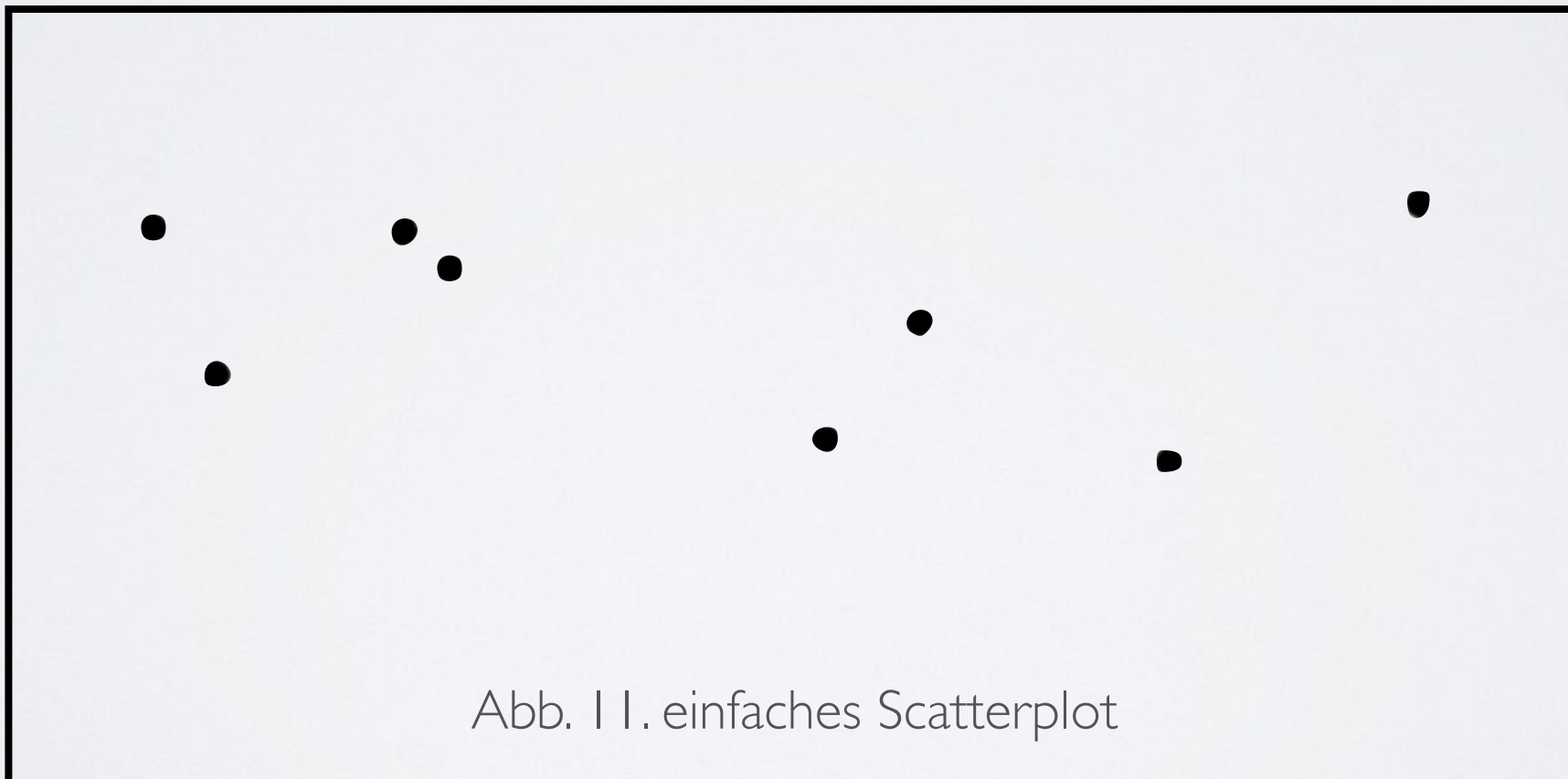


Abb. 11. einfaches Scatterplot

PHASE 4

Scatterplot - Labels

```
svg.selectAll("text")  
  .data(dataset)  
  .enter()  
  .append("text")  
  .text(function(d) {  
    return d[0] + "," + d[1];  
  })  
  .attr("x", function(d) {  
    return d[0] + 20;  
  })  
  .attr("y", function(d) {  
    return d[1];  
  })
```

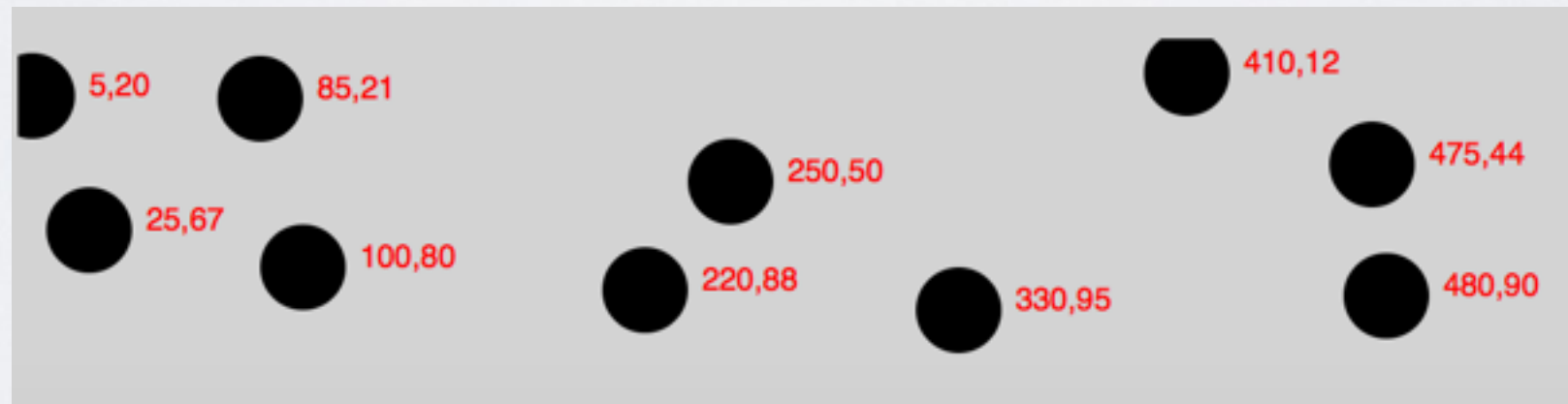
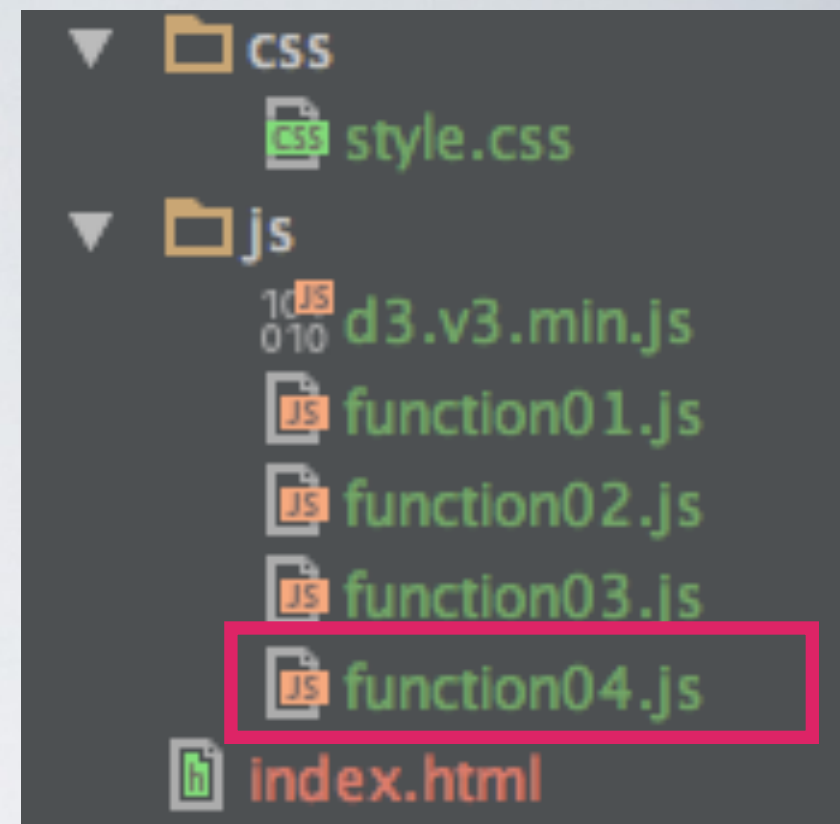


Abb. 12. Scatterplot mit Labels

PHASE 4



TODO

HANDOUTS

<http://bit.ly/1FOuGbZ>