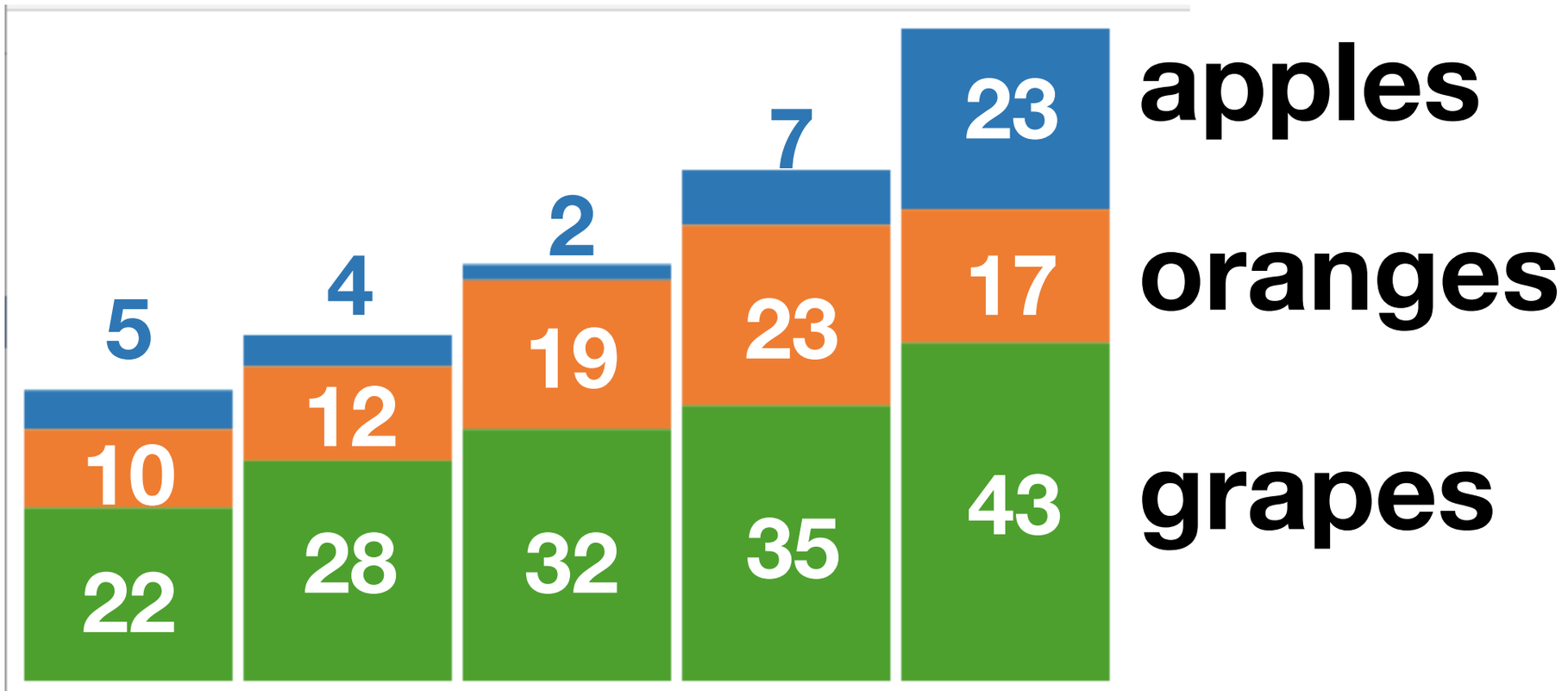
Layouts (Ch. 13)

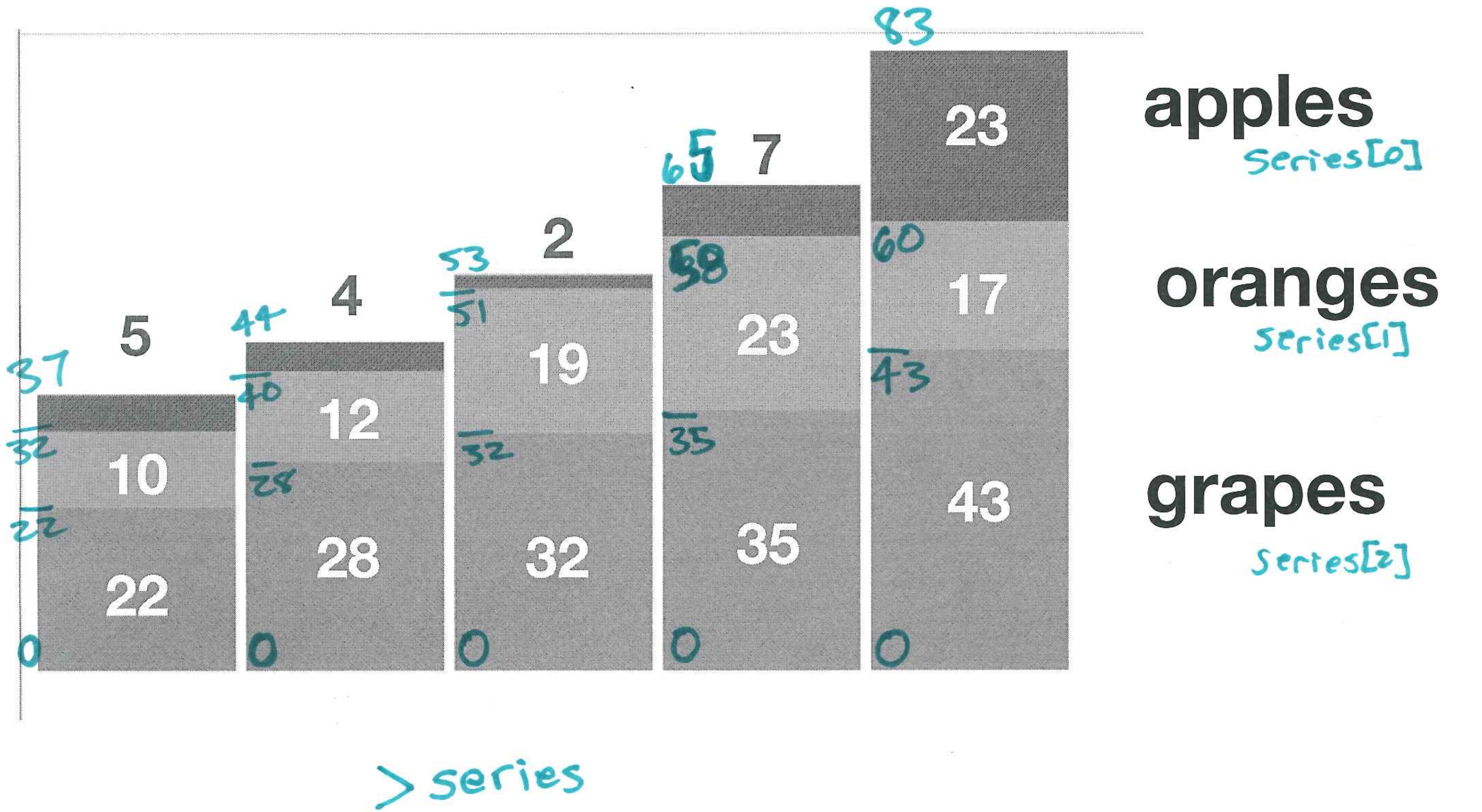
reorganize your data into a more convenient form for the type of graph you want to draw

d3.stack()

d3.stack()

```
var dataset = [
    { apples: 5, oranges: 10, grapes: 22 },
    { apples: 4, oranges, 12, grapes: 28 },
    { apples: 2, oranges: 19, grapes: 32 },
    { apples: 7, oranges, 23, grapes: 35 },
    { apples: 23, oranges, 17, grapes: 43 }
];
```





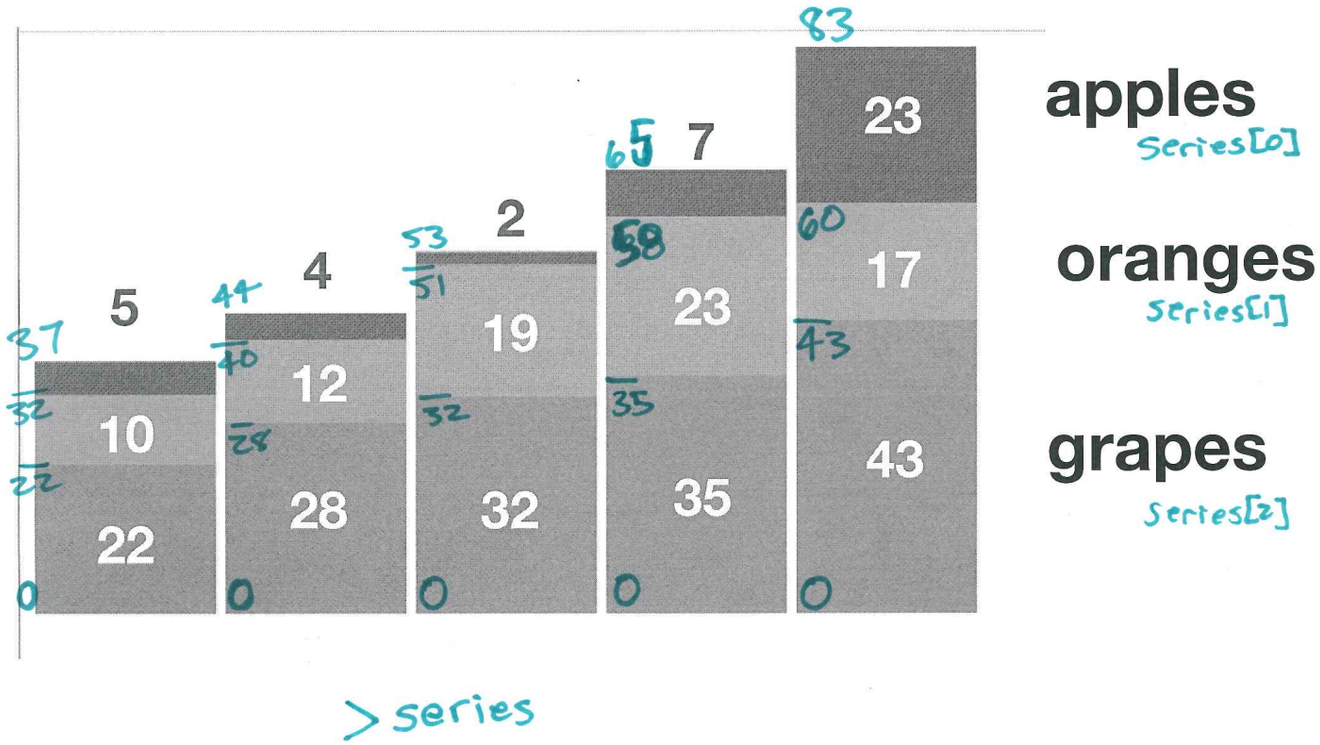
5
4
2
7
23
10
12
19
23
17
22
28
32
35
43
apples
oranges
grapes
series[0]
series[1]
series[2]
37
32
22
0
44
40
28
53
51
32
65
58
35
83
60
43
> series

d3.stack()

```
var dataset = [
    { apples: 5, oranges: 10, grapes: 22 },
    { apples: 4, oranges, 12, grapes: 28 },
    { apples: 2, oranges: 19, grapes: 32 },
    { apples: 7, oranges, 23, grapes: 35 },
    { apples: 23, oranges, 17, grapes: 43 }
];

var stack = d3.stack()
    .keys( [ "apples", "oranges", "grapes" ])
    .order(d3.stackOrderDescending);

var series = stack(dataset);
```




```
> series
```

```
▼ (3) [Array(5), Array(5), Array(5)]
  ▼ 0: Array(5)
    ▶ 0: (2) [32, 37, data: {…}]
    ▶ 1: (2) [40, 44, data: {…}]
    ▶ 2: (2) [51, 53, data: {…}]
    ▶ 3: (2) [58, 65, data: {…}]
    ▶ 4: (2) [60, 83, data: {…}]
      index: 2
      key: "apples"
      length: 5
    ▶ __proto__: Array(0)
  ▼ 1: Array(5)
    ▶ 0: (2) [22, 32, data: {…}]
    ▶ 1: (2) [28, 40, data: {…}]
    ▶ 2: (2) [32, 51, data: {…}]
    ▶ 3: (2) [35, 58, data: {…}]
    ▶ 4: (2) [43, 60, data: {…}]
      index: 1
      key: "oranges"
      length: 5
    ▶ __proto__: Array(0)
  ▼ 2: Array(5)
    ▶ 0: (2) [0, 22, data: {…}]
    ▶ 1: (2) [0, 28, data: {…}]
    ▶ 2: (2) [0, 32, data: {…}]
    ▶ 3: (2) [0, 35, data: {…}]
    ▶ 4: (2) [0, 43, data: {…}]
      index: 0
      key: "grapes"
      length: 5
    ▶ __proto__: Array(0)
```

Now what?

Step 1:
Append a group for each array
(that is, each fruit)

```
var groups = svg.selectAll("g")
    .data(series)        <-- Bind the data
    .enter()                 one series per group
    .append("g")
    .style("fill",(d,i) => colors(i));
```

One fruit per group

<g> **<g>** **<g>**

```
> series
◂ ▼ (3) [Array(5), Array(5), Array(5)]
    ▼ 0: Array(5)
      ▶ 0: (2) [32, 37, data: {…}]
      ▶ 1: (2) [40, 44, data: {…}]
      ▶ 2: (2) [51, 53, data: {…}]
      ▶ 3: (2) [58, 65, data: {…}]
      ▶ 4: (2) [60, 83, data: {…}]
        index: 2
        key: "apples"
        length: 5
      ▶ __proto__: Array(0)
```

```
▼ 1: Array(5)
  ▶ 0: (2) [22, 32, data: {…}]
  ▶ 1: (2) [28, 40, data: {…}]
  ▶ 2: (2) [32, 51, data: {…}]
  ▶ 3: (2) [35, 58, data: {…}]
  ▶ 4: (2) [43, 60, data: {…}]
    index: 1
    key: "oranges"
    length: 5
  ▶ __proto__: Array(0)
```

```
▼ 2: Array(5)
  ▶ 0: (2) [0, 22, data: {…}]
  ▶ 1: (2) [0, 28, data: {…}]
  ▶ 2: (2) [0, 32, data: {…}]
  ▶ 3: (2) [0, 35, data: {…}]
  ▶ 4: (2) [0, 43, data: {…}]
    index: 0
    key: "grapes"
    length: 5
  ▶ __proto__: Array(0)
```

> d3.select("g").data()

```
◂ ▼ [Array(5)] i
    ▼ 0: Array(5)
      ▶ 0: (2) [32, 37, data: {…}]
      ▶ 1: (2) [40, 44, data: {…}]
      ▶ 2: (2) [51, 53, data: {…}]
      ▶ 3: (2) [58, 65, data: {…}]
      ▶ 4: (2) [60, 83, data: {…}]
```

Step 2: Draw the rects

```
var rects = groups.selectAll("rect")
    .data(d => d)
    .enter()
    .append("rect")
    .attr("y", d => yScale(d[1]))
    .attr("height", d =>
        yScale(d[0]) - yScale(d[1]))
    .attr(
    ...
    );
});
```

<-- Bind values to rectangles

IDVW2, p. 270

Example

```
var rects = groups.selectAll("rect")
    .data(d => d)
    .enter()
    .append("rect")
    .attr("y", d => yScale(d[1]))
    .attr("height", d =>
         yScale(d[0]) - yScale(d[1]))
    .attr(
    ...
    );
});
```

d[0] = 32
d[1] = 37

yScale(d[0]) = 272 px
yScale(d[1]) = 252 px

```
var yScale = d3.scaleLinear()
  .domain([0, 100])
  .range([400, 0]);
```