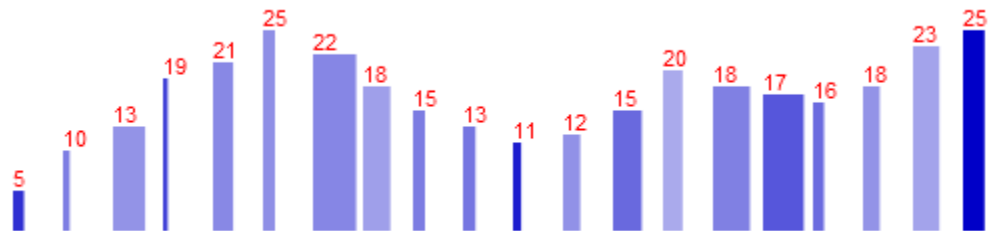# D3 Visualization Exercises

Exercise 1: Visualization Programming using d3

- **Install d3** on your computer.
- **Download PDF** of **d3 Textbook**:
  http://comminfo.rutgers.edu/~aspoerri/Teaching/InfoVisOnline/Resources/d3/Interactive_Data_Visualization_for_the_Web.pdf
  Read to all the chapters up to the end of **Chapter 6** (page 112)
- Use the following file as the starting point for your Ex1: https://github.com/alignedleft/d3-book/blob/master/chapter_06/21_making_a_bar_chart_aligned.html
- Make following modifications:
  - Use **data set with three data variables**:
    var dataset = [[5, 23, 75], [10, 14, 34], [13, 67, 23], [19, 10, 65], [21, 42, 29], [25, 25, 25], [22, 90, 30], [18, 57, 17], [15, 25, 35], [13, 26, 39], [11, 17, 85], [12, 36, 24], [15, 60, 45], [20, 41, 11], [18, 77, 33], [17, 85, 55], [16, 23, 44], [18, 35, 23], [23, 55, 15], [25, 45, 100]];
  - **First data variable** is used for **height** of bar.
  - **Second data variable** is used to scale **width** of bar (need to scale how the width is computed in the example code by the 2nd variable divided by 100). In short: the *width is proportional to value of the 2nd variable*, which you can assume is a value between 1 and 100.
  - **Third variable** is used to control **fill-opacity** of bar, where the fill-opacity can not be less than 0.25 and the 3rd variable scales 0.75 by being divided by 100).
  - The **color** is rgb (0, 0, 200).
  - The **text label** has to be shown in **red** and needs to be placed at the **top** of the bar, be **left-aligned** and **fully visible** (to achieve full visibility, add an **area** to the chart **at the top** that has a height of **20 pixels**).
  - The final chart needs to look like this:

- **Submit**: URL of Ex1 web page that displays the bar chart as specified.

Exercise 2: Visualization Programming using d3

- Resources > Tools > d3 contains a PDF of **d3 Textbook**: read the **Chapters 7 - 10**.
- Use your Ex1 as a starting point for Ex2:
  - Use *data set with three data variables*:
    var dataset = [[5, 23, 75], [10, 14, 34], [13, 67, 23], [19, 10, 65], [21, 42, 29], [25, 25, 25], [22, 90, 30], [18, 57, 17], [15, 25, 35], [13, 26, 39], [11, 17, 85], [12, 36, 24], [15, 60, 45], [20, 41, 11], [18, 77, 33], [17, 85, 55], [16, 23, 44], [18, 35, 23], [23, 55, 15], [25, 45, 100]];
  - As in Ex1, *first data variable* is used for *height* of bar.
    Second data variable is used to scale *width* of bar.
    Third variable is used to control *fill-opacity* of bar.
  - Use **width = 500** and **height = 250** for the bar chart.
- The following files mentioned in the textbook will be useful :
  - Files in Chapter 9 and Chapter 10
- Make following modifications and enhancements to Ex1:
  - Add ordinal **xScale** and linear **yScale** (similar to bar chart example in Chapter 9, but keep in mind that we have multidimensional dataset) and update how bar chart is created.
    - Make sure to use approach described on page 117 in Chapter 7 for computing max for 1st variable.
  - Add **hover effect for bar** so that fill color is changed to **red** using **CSS** rule (as covered in Chapter 10 example).
  - Add **sort capabilities** when clicking on bar so that user can sequentially sort by the **1st, 2nd and 3rd variable, respectively** (hint: use Modulus (division remainder) operator) and make sure that **text label** shows value of variable currently used to sort the bar chart.
    - Chapter 10 has an example where user can click on a bar to sort the bar chart.
    - When using **ascending(a, b)**, keep in mind that a and b are *arrays* and you need to specify which variable to use in the sort operation.
    - To be able to "cycle through" the different variables when successively clicking on one of the bars use the **Modulus** (division remainder) operator in JavaScript to specify which variable to sort by.
  - Add **scaling capabilities** so that height of the bars is scaled by 1/3, 2/3 and 1, respectively, by successively clicking on a paragraph above the chart.
    - Chapter 9 has example where clicking on a paragraph changes the bar chart.
    - **Modulus** (division remainder) operator in JavaScript could be useful to employ.
  - The **text label** has to be shown in *red* and needs to be placed at the *top* of the bar, be *left-aligned* and *fully visible* as well as **maintain its position with respect to the top of a bar** as the bar chart is sorted and/or scaled .
    - Chapter 10 contains example that show how to update position of text labels.
  - Add **transition effects** when the user performs a **sorting and/or scaling** operation.
    - Use a **delay function** as specified on **page 153** in Chapter 9 for all transitions.
  - The final chart needs to **look and behave** as shown in this screencast (if you can't hear the audio right away, please *reload* the screencast page in the browser).
    http://comminfo.rutgers.edu/~aspoerri/Teaching/InfoVis_ITI/d3/Ex2_d3/Ex2_d3.html

- **Submit**: URL of Ex2 web page that displays the bar chart as specified.