F21DV Lab 1 Report

Date:
Demonstrated To:
Repository: `https://github.com/linarietuma/linarietuma.github.io`

1 Introduction

This report summarises the reflections and results of exercises for Data Visualisations and Analytics (F21DV) course lab 1.

2 Part 2: D3 Setup

2.1 Exercise 1: What version number is displayed in the console output window?

D3 version number displayed is 7.3.0.

2.2 Exercise 2: Change other style properties of the paragraph tag.

Use .select("p") to select the first HTML element (e.g., div, p, #id, .class) of the specified type and .style("color", "red") to format the selected element where the first argument specifies CSS property and the second argument denotes its value.

2.3 Exercise 3: Write a loop which adds 10 'div' elements and sets the contents to the count value (i.e., 1, 2, 3, ...). Also the colour of the first 5 elements are red and the last 5 elements are green.

Both .style() and .attr() methods are used in combination with the .select()/ .selectAll() methods to change the properties of specified element/s. While .style() is used to change CSS styling, . attr() changes an attribute of the selected element (e.g. set class, id etc).

2.4 Exercise 4: 'selecting' and modifying your 'div' elements after you've created and added them.

When creating the elements, give a unique id to each which later allows to specifically target and change the desired elements.

2.5 Exercise 5: Exercise: Add to the 'chain syntax' version for the 'hello world' example above – so it also sets the 'color' of the text to green.

In D3, methods can be chained together to avoid storing intermediary variables that are passed between methods.

3 Part 3: Data

3.1 Exercise 6: Modify the example above so the 'otherdata' contains an additional variable called color (print this color value out in the 'text' method).

Use data() method to select data and iterate through them, allows accessing/ modifying values one by one. Add 'color' variable to each object, retrieve the value of the variable by calling d.color.

3.2 Exercise 7: Change the bounds check so the color is red for numbers between 50 and 100.

Assumed this includes 50 and 100.

4 Part 4: Data Binding

4.1 Exercise 8: Modify the above code, use the following data:

5 Part 5: Loading Data

5.1 Exercise 9: For the example above, to count how many of the names include 'Mr.' and 'Mrs' (or other). Also print out other details using other column header information, such as, how many passengers are 'male' and how many 'female'.

5.2 Exercise 10: Exercise: Write an update to the example above, so extra elements are added to the window to display information. For instance, display paragraphs for the total patients with heart failure between 1-30, 31-40, 41-60, 61-100. Process the data, store it in an array then pass that array to 'selectAll()', 'data()' as discussed in previous sections.

6 Part 6: SVG

6.1 Exercise 11: Exercise: Modify the code so the example draws a 'square shape' (4 lines) – each side of the square a different color.

6.2 Exercise 12: Build an SVG scene which is created from an external file. You need to create a csv with the information about the shapes. You should include columns in your csv file for the type of shape (circle, rectangle, ellipse, line), its dimensions and position, and color. Your program reads the data and creates and displays the shapes to the screen.

6.3 Exercise 13: Extend the example to include the 'enter' and 'exit' concepts. So that the svg elements are updated, created or removed based on the csv data from your csv file.

7 Part 7: Bar Chart

7.1 Exercise 14: Extend the simple bar chart example to display the heart failure data you processed in Part 5 (Part 5 - Loading Data) from the csv file. (i.e., age ranges for people with heart failure).

7.2 Exercise 15: Modify the simple bar chart to use color more (i.e., values over a certain threshold are displayed in 'red').