Lab 3 (2/3)

Submit your team number

Question *Submitted Feb 3rd 2023 at 2:14:12 pm*

Please enter your team number.

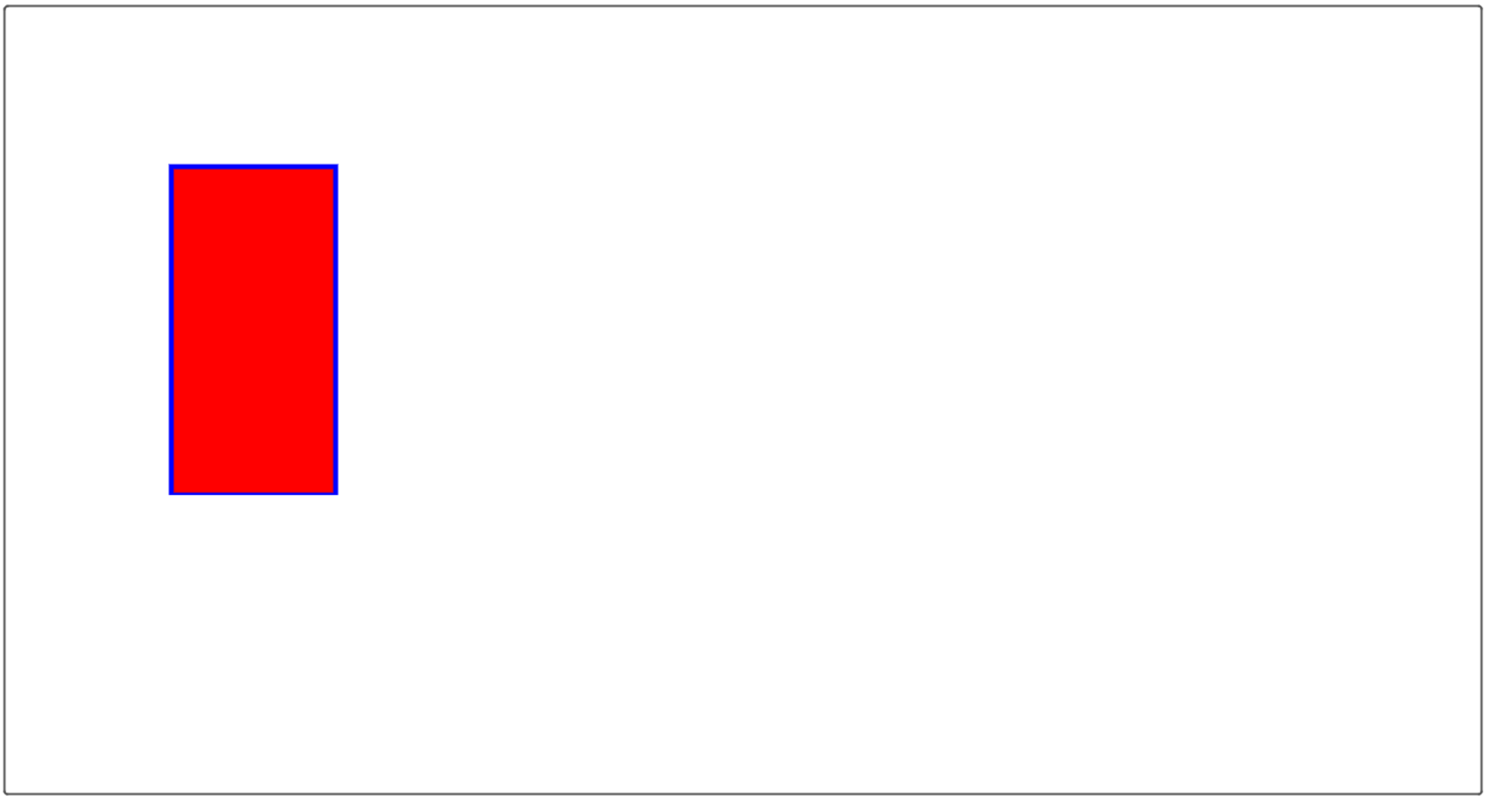
50

1. Using Inspect tool (and/or other methods) to debug

Copy paste the following code in an HTML file.

```
<!DOCTYPE html>
<html>


<head>
    <script src="https://d3js.org/d3.v7.min.js"></script>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <title>Debug</title>
</head>


<body>
    <div id = "svgcontainer">
    </div>
    <script type="text/javascript">
        var width = 300;
        var height = 300;
        var svg = d3.select("svgcontainer")
            .append("svg").attr("width", width)
             attr("height", height);
        svg.append("rect");
            .attr("x1",100 )
            .attr("y1", 100)
            .attr("w", 100)
            .attr("h", 200)
            .style("color", "red")
            .stye("stroke", "blue")
            .style("stroke-width", "3");


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

Your task is to debug the code to get the following rectangle on the browser. If you are using Google Chrome, the Chrome DevTools (you can open it by right-click > inspect on Chrome) will be very helpful. Your HTML page should look similar to this.



⚠ Notice: The rectangle is not supposed to be placed exactly on the top left corner.

Question *Submitted Feb 3rd 2023 at 2:17:15 pm*

Copy/paste your HTML code.

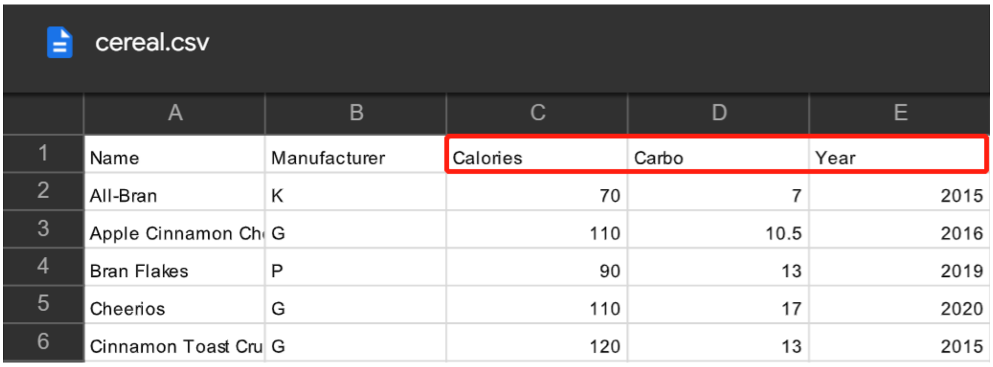
```
<!DOCTYPE html>
<html>
<head>
 <script src="https://d3js.org/d3.v7.min.js"></script>
 <meta charset="utf-8">
 <meta name="viewport" content="width=device-width, initial-scale=1">
 <title>Debug</title>
</head>
<body>
 <div id = "svgcontainer">
 </div>
 <script type="text/javascript">

 var width = 300;

 var height = 300;

 var svg = d3.select("#svgcontainer")

 .append("svg")

 .attr("width", width)

 .attr("height", height);

 svg.append("rect")

 .attr("x",100 )

 .attr("y", 100)

 .attr("width", 100)

 .attr("height", 200)

 .attr("fill", "red")

 .attr("stroke", "blue")

 .attr("stroke-width", "3");

 </script>

</body>

</html>
```

2. Write your callback function to convert all numeric columns into float/int

Now that you have an understanding how to debug, let's get started with our analysis of cereal.csv in lab3.html. **Download** the two files and make sure they are in the same folder.

First we will load cereal.csv to lab3.html, using d3.csv(). The second argument of d3.csv() is a callback function. The callback function allows us to do any data processing work, while loading the csv file (convenient!).

Lab3.html already defined an empty callback function `rowConverter()`. This function takes a row of the csv file as an input, which is represented as an array of Strings, and returns an object which has a list of key-values. Notice that the Calories, Carbo, and Year have numerical data yet they are going to be represented as Strings when d3.csv() is called; so `rowConverter()` should convert these String values to either integer or float. To do so, you can use parseInt() and parseFloat() functions.



cereal.csv

	A	B	C	D	E
1	Name	Manufacturer	Calories	Carbo	Year
2	All-Bran	K	70	7	2015
3	Apple Cinnamon Ch	G	110	10.5	2016
4	Bran Flakes	P	90	13	2019
5	Cheerios	G	110	17	2020
6	Cinnamon Toast Cru	G	120	13	2015

Complete `rowConverter()` on lab3.html so that the Name and Manufacturer keys hold String values while the Calories, Carbo, and Year keys hold integer/float values. Following example will help you construct your `rowConverter()`.

If we have a CSV file like:

```
cities.csv:

city,state,population,land area
seattle,WA,652405,83.9
new york,NY,8405837,302.6
boston,MA,645966,48.3
kansas city,MO,467007,315.0
```

We can write a callback function `rowConverter` like this.

```
var rowConverter= function(d){
    return {
        city : d.city,        // key: city, value: d.city
        state : d.state,        // key: state, value: d.state
        population : parseInt(d.population), // key: population, value: int of d.population
        land_area : parseFloat(d["land area"]) // key: land_area, value: float of d["land area"]
  }
}
```

Question *Submitted Feb 3rd 2023 at 2:18:40 pm*

Have you written your own **rowConverter** function which can convert all numeric columns of cereal.csv into int/float?

● Yes, we are done!

○ No we are not done

3. Analyze the data using D3 library methods

You can read a csv file, say cereal.csv, using d3.csv(). To do so, you need to load and parse data using `rowConverter()` as a callback function.

For example, if we want to load and parse a csv file "cities.csv" with our `rowConverter()` and output the data to the console, we can use the following code:

```
d3.csv("cities.csv", rowConverter).then(function(data) {
  console.log(data);

  // this is where you add the code to do the tasks below



});
```

Question 1 *Submitted Feb 3rd 2023 at 5:27:51 pm*

Once you loaded cereal.csv, complete the following tasks:

1. Group the cereals by manufacturers
2. Get the sum of carbo, per manufacturer, per year
3. Find the sum of calories for manufacturer “K” in 2020

`Console.log` your data at every step.

Copy/paste the relevant part of your code.

```
console.log(data)

 //Group the cereals by manufacturers

 let manuGrouped = {};

 for (let cereal of data){

 if(!manuGrouped[cereal.Manufacturer]){

 manuGrouped[cereal.Manufacturer] = [JSON.stringify(cereal.Name)];

 }

 manuGrouped[cereal.Manufacturer].push(JSON.stringify(cereal.Name));
 }

 console.log(manuGrouped)

 for (var manu in manuGrouped) {

 manuGrouped[manu] = [...new Set(manuGrouped[manu])];

 }

 console.log(manuGrouped);

 //Get the sum of carbo per manufacturer per year

 let carbManuYear = {};

 for (let cereal of data){

 if(!carbManuYear[cereal.Year]){

 let carbManu = {};

 carbManu[cereal.Manufacturer]=cereal.Carbo;

 carbManuYear[cereal.Year] = carbManu;

 }

 if(!carbManuYear[cereal.Year][cereal.Manufacturer]){

 let carbManu = carbManuYear[cereal.Year];

 carbManu[cereal.Manufacturer] = cereal.Carbo;

 carbManuYear[cereal.Year] = carbManu;

 }

 let carbManu = carbManuYear[cereal.Year];

 carbManu[cereal.Manufacturer]+=cereal.Carbo;

 carbManuYear[cereal.Year]= carbManu;

 };

 console.log(carbManuYear);

 // Find the sum of calories for manufacturer “K” in 2020?

 let manuCal = {};
```

```
for (let cereal of data){
if (cereal.Year==2020){
if(!manuCal[cereal.Manufacturer]){
manuCal[cereal.Manufacturer] = cereal.Calories;
}
manuCal[cereal.Manufacturer]+= cereal.Calories;
}};
console.log(manuCal);
console.log(manuCal["K"])
var manuGroupedTable = document.getElementById("manuGrouped");
var html = "<tr><th>Manufacturer</th><th>Cereals</th></tr>";
for (var manu in manuGrouped) {
html += "<tr><td>" + manu + "</td><td>" + manuGrouped[manu] + "</td></tr>";
}
manuGroupedTable.innerHTML = html;
var carbManuYearTable = document.getElementById("carbManuYear");
var html = "<tr><th>Year</th><th></th></tr>";
for (var year in carbManuYear) {
var table = "<table class=center><tr><th></th><th>Total Calories by Manufacturer</th></tr></table>";
for (var manu in carbManuYear[year]){
table+="<tr><td>" + manu + "</td><td><b>"+ carbManuYear[year][manu]+"</b></td></tr>";
};
html += "<tr><td><b>" + year + "</b></td><td>"+table+"</td></tr>";
}
carbManuYearTable.innerHTML = html;
document.getElementById("K2020").innerHTML = "Manufacturer K Total Calories: "+"
```

```
<b>"+manuCal["K"]+"</b>";

 });
```

Question 2 *Submitted Feb 3rd 2023 at 5:27:54 pm*

Are you done with the activity and ready to submit?

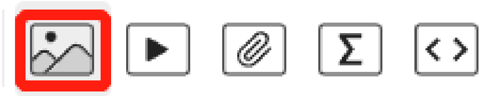
● Yes, we are done!

○ No we are not done

Upload Your Files

Question 1 *Submitted Feb 3rd 2023 at 5:30:33 pm*

Upload the screenshot of your resulting webpage (Your console information). You will need to click the "clip" button to upload a file into the Answer box.

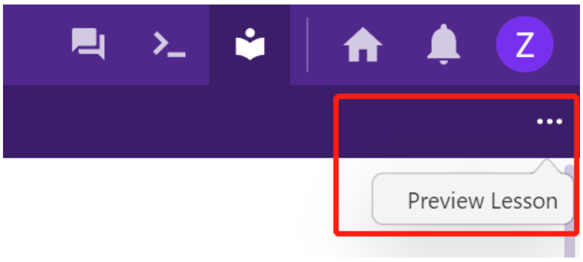


lab3.png

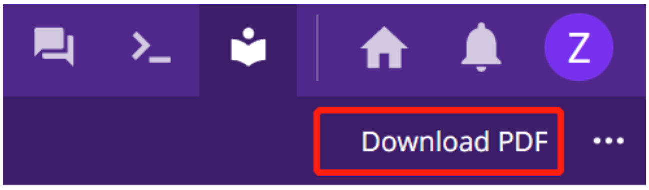
Question 2 *Submitted Feb 3rd 2023 at 5:30:37 pm*

You need to download the PDF of lecture exercise 3 and upload it with other files to the Gradescope. Follow the instructions on how to download PDF file:

1. Click on the ellipsis button and the Preview Lesson.



2. After that, click on the Download PDF button.



● PDF downloaded!

○ Haven't done yet!

Question 3 *Submitted Feb 3rd 2023 at 5:30:47 pm*

Upload the following files to Gradescope. You need to make **a group submission,adding all present members in your team**, so that the present members get the participation credit.

Files to upload:

- lab3.html
- PDF you downloaded as Q2

◉ Our team uploaded the the files on gradescope!

○ Oops, our team did not upload the files on gradescope!

Feedback

Question

Was the activity today clear? If not, please share how the course can improve it. Your comments will help us design future lab content (and also future students).

No response