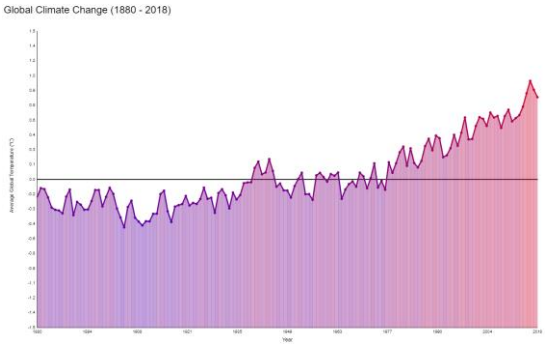
Project title:	Data Visualizer
Topic:	WK14: Extending Timeseries Charts

What progress have you made this topic?

The pay gap time series and climate change visuals were previously not extended as of the mid-term submission. This week, I focused on extending both these time series charts, leveraging on the multi-line timeseries chart constructor function that I had previously created for the COVID19 weekly visual. I created a new constructor that incorporates the elements of both the pay gap and climate change charts, making it a perfect use case of code reusability through modularization. I managed to refine the shading of area beneath the lines and added start/end date filters to the X axis.

What problems have you faced and were you able to solve them?

The most challenging part of this enhancement was getting the individual rectangles beneath the lines to line up nicely to give an illusion of a shaded area beneath the charts. After some brainstorming, I was able to create a seamless transition between one point to the next. This was done through the combination of a nested loop and some arithmetic calculations. By calculating the distance number of pixels between the x-axis of the previous and current points, and the increase/decrease in the Y axis, I was able to make small changes to the slope at each pixel, creating this illusion of a seamless shaded area chart.




```
//draw thin rectangles below line to give illusion of shading
for (var j=0; j<xDiff; j++){
  rect(
    //increment x by 1 pixel at a time
    this.mapTimeToWidth(previous.time)+j,
    //change height gradually from the previous to current value at each pixel
    this.mapYToHeight(previous.volume)+j*(yDiff/xDiff),
     //draw one pixel at a time
    1,
    // change the height of the rectangle such that it always end at bottom of chart
    this.mapYToHeight(this.minY)-(this.mapYToHeight(previous.volume)+j*(yDiff/xDiff))
  )
}
```

I also had difficulties animating the plot. Therefore, this task was postponed to the following weeks.

What are you planning to do over the next few weeks?

In the coming weeks, I will continue working on the pending extensions. In addition to that, as I am still facing issues animating the plots (as shown with the climate change template), I will also add this as a pending task for the coming weeks.

Are you on target to successfully complete your project? If you aren't on target, how will you address the issue?
Due to the complexities faced in enhancing and integrating the two visuals, the project timelines will be pushed back by 1 week (the bubble plot visualization was supposed to be completed this week as well). Nevertheless, I have created some time buffers for this project. Therefore, I will still be able to complete the required activities within the deadline.

Project title:	Data Visualizer
Topic:	Wk15: Extending Pay gap by Job Visual (Bubble Plot)

What progress have you made this topic?

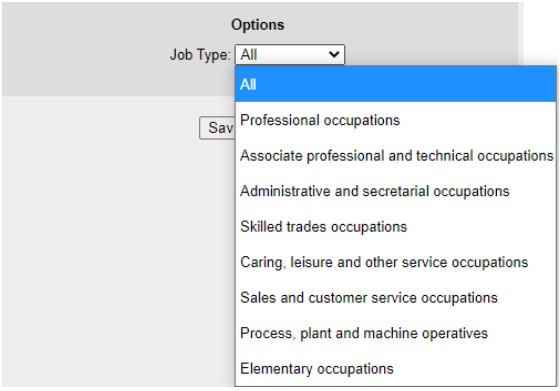
This week, I completed the bubble chart for the pay gap visualization. I labeled the axis of the charts to provide more context on what the positioning on the charts means, added color coding to differentiate male dominated vs female dominated jobs, and added a hover-over functionality so that users can get details of each data point when hovering over it. I also added a drop down to allow users to filter the data points based on the job type, a feature that was within the dataset but not utilized in the template.

What problems have you faced and were you able to solve them?

When I initially implemented the dropdown list for filtering the job types, I had was faced with an issue of not being able to see all the jobs on the chart at once, as exactly one of the job types was selected at once. To counter this, I added an "All" option to the list before looping through the unique job types. By using conditionals, I was then able to select rows of the data differently based on the selected option.

```
// Fill the options with all job types.
this.select.option("All"); // add an option to include all items
for (var i=1;i<jobtypes.length;i++){
  this.select.option(jobtypes[i]);
}
```

```
if(filter == "All"){
  var label=this.data.getColumn('job_subtype');
  var x = stringsToNumbers(this.data.getColumn('proportion_female'));
  var y = stringsToNumbers(this.data.getColumn('pay_gap'));
  var size = stringsToNumbers(this.data.getColumn('num_jobs'));
}
else{ // if a filter is being selected, only pass rows where filters match value
```



What are you planning to do over the next few weeks?

Now that all the template extensions have been modified, I will be enhancing the COVID19 extension that I have created in the coming weeks. I will also resolve the issue with animating the timeseries plots next week. Once that is done, I will be able to proceed with my stability and user testing activities.

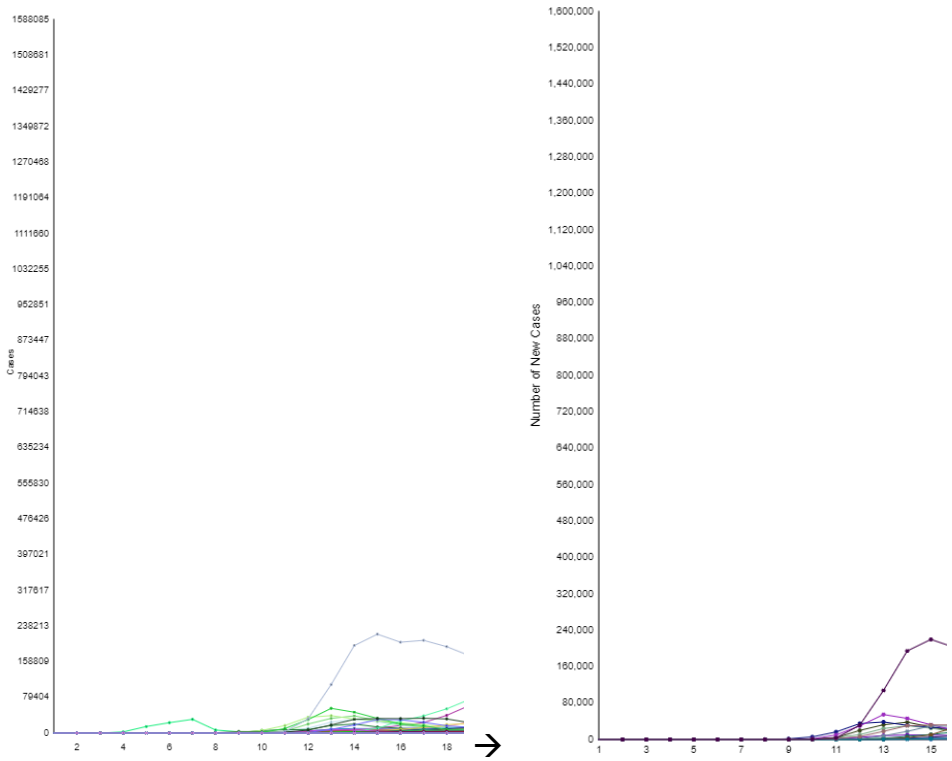
Are you on target to successfully complete your project? If you aren't on target, how will you address the issue?
Based on the updated timelines, I am currently on schedule of successful completion.

Project title:	Data Visualizer
Topic:	Wk16: Extending COVID19 visuals

What progress have you made this topic?

This week, I worked on enhancing the 2 existing COVID19 enhancements I have. For the geographic view chart, I found a map that was able to provide better contrast of the data points. I also removed the condition that only displayed top X country names. Instead, I am now displaying all country names with font sizes proportional to the number of cases.

For the weekly view chart, I modified the Y-axis, which previously displayed hard-to-read numbers, plotting them at every 10K interval instead, while maintaining the accurate positioning of data points relative to the axis. The numbers are also nicely formatted with commas.



Additionally, I was able to utilize the grid on/off functionality within the helper function by creating a dynamically changing button option and passing the state of this button as parameter within the constructor function. Finally, I also managed to successfully animate the time series plots – a task that was previously outstanding.

What problems have you faced and were you able to solve them?

One problem I encountered was with the toggle button. I was initially not sure how to modify the button's text and value. I tried the using binary 0/1 values to the button, but that did not work as

expected. Therefore, I changed the values to toggle between “On” and “Off”, using conditional statements to translate them into binary before passing it as a parameter to the chart template.

```
function ToggleGrid() {
  // assign DOM element to a variable
  var v= document.getElementById("grid-toggle").children[0];

  if (v.value=="On"){
    // If the current value is "On", toggle it off
    v.value = "Off";
    // Update the text on the button to indicate instruction to turn back on
    v.innerText= "Turn On";
  }
  else{
    // If the current value is "Off", toggle it on
    v.value = "On";
    // Update the text on the button to indicate instruction to turn back off
    v.innerText= "Turn Off"
  }
}
```

```
// set the gridOn value to true/false based on the current button value
var gridOn;
if(document.getElementById("grid-toggle").children[0].value=="On"){
  gridOn=true;
}
else{
  gridOn=false;
}
```

Another problem that I was previously facing was in animating the time series plots when the visual is loaded. In the template code, drawing was performed without modularization. This made it possible to update the year count within each year. However, due to the way my code was organized, the constructor function for plotting the chart is called at each frame. After multiple iterations and trial-and-testing, I found a way to animate the visuals, although in a slightly different way than what was presented in the template. I still initialized a frame count variable within the set-up method of the visualization constructor. However, the animation is achieved by modifying last date to be plotted on the chart at each frame count.

```
// Set min and max Weeks: assumes data is sorted by date.
var startTime = time[0];
// animate plot for each from count
//slows down plotting by 5 times
if(frameCount<=(time.length-1)*5){
  var endTime = time[Math.ceil(frameCount/5)];
}
else{
  var endTime = time[time.length-1];
}

this.timeseries= new TimeSeriesChart(
  x=time,
  y=pay_gap,
  title=title,
  xLabel=xAxisLabel,
  yLabel=yAxisLabel,
  gridOn=gridOn,
  startTime=startTime,
  endTime=endTime,
  minY=0,
  maxY=50,
  baseLine=0
);
this.timeseries.draw();

frameCount++;
```

What are you planning to do over the next few weeks?
As all extensions are now implemented, I will be performing stability & user tests over the next few weeks.
Are you on target to successfully complete your project? If you aren't on target, how will you address the issue?
I am still on track to completing the project successfully.

Project title:	Data Visualizer
Topic:	Wk17: Stability Testing

What progress have you made this topic?

This week, I did some rigorous stability testing using test cases developed for each visualization (see /documentation/stability-testing.xlsx). The tests were performed with the console open to identify any errors or warnings that may have been triggered at any stage of the process. This ensured full stability of the application.

What problems have you faced and were you able to solve them?

There were no major problems faced this week as the tutorials provided good guidelines on how to formulate and execute stability test cases. There were only 2 features that failed the stability test (test case 7 & 8).

For test case 7.5, no visual to be displayed at the minimum end year slider value. The issue was easily solved but changing the starting point of the end year slider.

```
this.endYrScaleDiv=createDiv("End Time: ").id("end-yr-scale");
this.endYrScaleDiv.style('padding','10px');
this.endYrScaleDiv.style('font-weight','normal');
this.endYrScaleDiv.parent('visual-options');
this.EndYrScale = createSlider(1,nYrs,nYrs,1); //default value = max years
this.EndYrScale.parent('end-yr-scale');
```

For test case 8.2, the line thickness of the axis of the "Pay gap: Gender inequality by Job Type" chart changed depending on the thickness of lines in the last selected visual. This was easily resolved by indicating the stroke weight within the function that draws the axis lines.

```
function drawBoxAxis(layout,topText,bottomText,leftText,rightText) {

  fill(0);
  stroke(0);
  strokeWeight(2);
```

No other issues were identified.

What are you planning to do over the next few weeks?

Next week, I will be conducting user tests with 2 individuals, observing their interactions with the application, and getting feedback for areas for UI improvements.

Are you on target to successfully complete your project? If you aren't on target, how will you address the issue?

I am on track to completing the project successfully.

Project title:	Data Visualizer
Topic:	Wk18: User Testing

What progress have you made this topic?

This week, user testing was conducted with 2 individuals of different demographics. The results of the test can be found in /documentation/user-testing.xlsx.

One thing I realized from user testing was that users had to be prompted to hover over certain visuals for interactivity. Therefore, I added a user guide button and documentation to provide users with the ability to get the most functionality out of the visuals.

For the geographic visual, the fonts for smaller number of cases were harder to read. Therefore, the minimum font size was increased.

There were a couple of feedback that I got on the Weekly COVID19 chart.

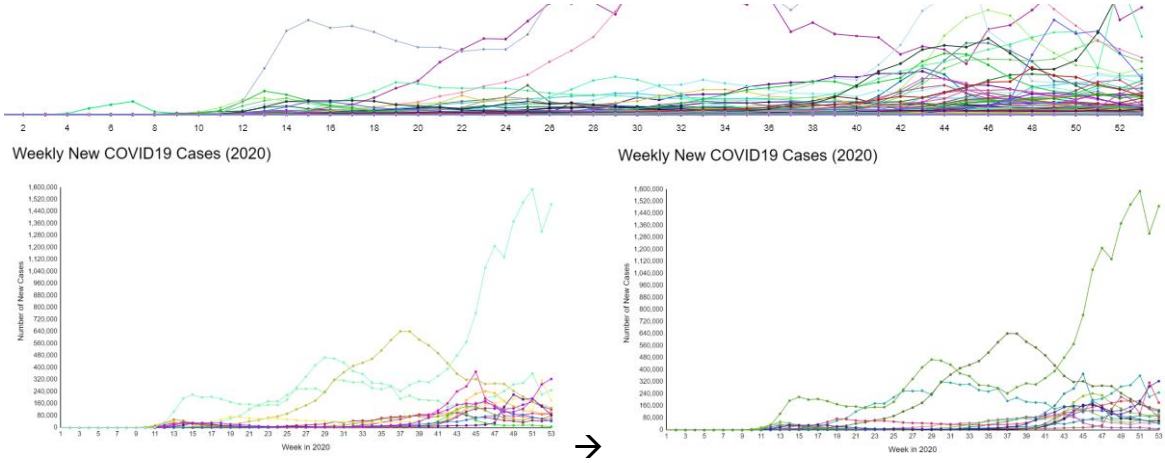
1. Some lines of the randomly generated numbers were too light. The lack of contrast made the graph difficult to read. Therefore, I changed the maximum for each RBG value to 220 instead of 255.

```
//create array of colours of lines
this.itemcolour=[];
for (var i = 1; i < this.data.getColumnCount(); i++) {
  //assign a random number to  each country in the dataset
  //ensure darker colour shades by setting limit of 200 instead of 255
  this.itemcolour.push(color(random(0,220),random(0,220),random(0,220)));
}
```

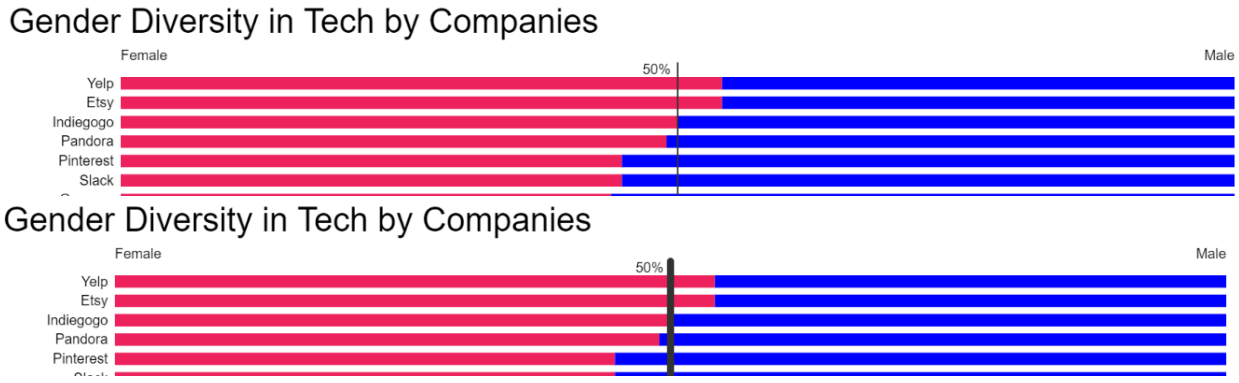
2. The initial graph had also a lot of clutter towards the bottom of the chart, which was information that could not be used for any insights. Therefore, a condition was included in the code to only include the country if there was at least 1 week where new cases recorded were above 100,000.

```
var country=data.columns[i]

// get the ith column of data (each category is one column)
var col=stringsToNumbers(data.getColumn(i));

//only draw line if the category has a value exceeding the data limit
if(max(col)>datalimit){

  for (var j=0; j < timestamps +1; j++){
    // Create an object to store data for the current Week.
    var current = {
      // store the current time step and volume in an object
      'time': startTime+ j,
      'volume': data.getNum(j,i)
    }
```

3. There was also feedback that the area over which the hover functionality worked was too small. Therefore, I increased the allowable region for which the hover functionality would work.



For the "Tech Diversity: Gender composition by company" visualization, the reference line was relatively thin, making it difficult for the user to see. Therefore, thickness of this line was increased.



Other than these, users had a relatively seamless interaction with the application – the filters within the side bar were especially intuitive. There were some really useful feedbacks that can help to boost the user's overall experience, but these changes can be further enhancements to improve the application further in the future.

What problems have you faced and were you able to solve them?

There were no major issues encountered with user testing. The stability test cases were performed once again after the changes were made to ensure nothing broke.

What are you planning to do over the next few weeks?

Now that tests have been completed, I will be enhancing the quality of my codes through folder restructuring, rigorous commenting, and modularizing functions wherever possible in the coming weeks.

Are you on target to successfully complete your project? If you aren't on target, how will you address the issue?
I am on track to completing the project successfully.

Project title:	Data Visualizer
Topic:	Wk19/20: Code Optimization

What progress have you made this topic?

I reorganized the structure of files and folders for the project. There is now a data folder, with each analysis type in its own folder. There is also a stylesheet folder containing the CSS stylesheet for the project. And finally, there is a script folder for all JavaScript files. The JavaScript files are categorized into 3 folders:

1. **chart-templates**, containing the replicable scripts that can be used to build a visualization.
2. **visualisations**, containing all visualisation script, each of which calling a template constructor function stored in the chart-templates folder.
3. **helpers**, containing all other JavaScript functions that are used across the constructors to process data and visuals.

I reduced long codes by implementing functions as separate codes in the "helpers" folder, modularizing as much as possible. Some helper functions include those to check if a mouse is hovered over a point, add titles to charts, and draw chart axes.

I have also worked to ensure that variables are locally scoped whenever possible.

Finally, I went through all codes to ensure they were well commented.

What problems have you faced and were you able to solve them?

One difficulty I faced with modulization is the limitation when it comes to customizing charts and functions. For example, I initially wanted the mouse hover functionality for the weekly COVID19 chart to increase the diameter of the ellipse whenever a mouse is over it. However, as the function is being used on gender equality bubble chart as well, such a feature would cause confusion to users hovering over the bubble plots. While this issue was not resolved in this project, it is possible to tackle it in future projects by passing a parameters into the modularized function indicated a flag of whether the size of the ellipse should be increased upon hover.

What are you planning to do over the next few weeks?

As the application is now stable with clean codes, I will be working on my reports in the coming week and ensuring the assignment is submitted on time.

Are you on target to successfully complete your project? If you aren't on target, how will you address the issue?

Yes, on track to successful submission.