1) I am extending the data visualization template, driven by a keen interest in data analytics and a quest to understand complex coding techniques. In addition to the template's original five charts, I've chosen to include two other extensions from Coursera: the Waffle Chart from week 11 and the Bubble Chart from week 13. In the future, I also aim to add a Stacked Area Chart, which is an original extension. For these, I will heavily rely on Object Orientation Programming, particularly encapsulation, closures, array manipulation and iteration, and nested loops for graphical generation (such as boxes and lines) and for manipulation of data from the tables.

   The Waffle Chart will be designed to display category-specific information for each day upon mouseOver and will utilise a random colour generator for the fill. I envision the intensive use of nested arrays and data manipulation.

   For the data for the Waffle Chart, I will be creating a CSV generator that will be able to auto populate survey results required for the chart.

   The Bubble Chart, on the other hand, will display the value and units for each bubble upon mouseOver. I plan to introduce a slider for the year selection, replacing the existing button group to enhance intuitiveness. Additionally, the texts within the bubble will be wrapped, demanding a lot of string manipulation, including breaking the original string into multiple components based on the bubble size.

   Lastly, the Stacked Area Chart is intended to demonstrate varied percentages across two time frames for efficient data comparison, much like a pie chart. It will need substantial data manipulation and will make use of vertexes to create shapes, which will require complex arrays to store shape coordinates based on table data.

   I anticipate the greatest challenge will come from implementing the Stacked Area Chart due to the lack of existing references and my limited experience with vertexes. However, I'm confident that with thorough and proper code planning, I can overcome these obstacles and draw the chart effectively.

2) I brainstormed the changes and updates that I can do for the 5 original visualisations as well as the 3 extensions, as well as sketching the graphic designs for each individual chart. For some charts, I have also drawn a code diagram that allows me to visualise the relationships between different variables and functions within the chart's constructor function. I have also updated the 'site map' that Coursera previously provided with the new extensions as well as the relationships between the different files to help with my understanding with the relationships.

Stage 1 For Midterms

Date No

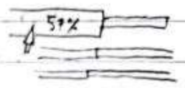
Coursera

Visualisations

- Stacked Bar Chart (Tech Diversity Gender)

  → Improvements:

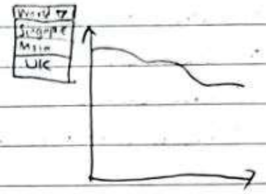
  - Display exact % on mouseover
  - Make it 3D, pop up upon mouseover.
  - Include more companies + scrollable?



- Line Graph (Pay Gap: 1997 - 2017)

  → Improvements:

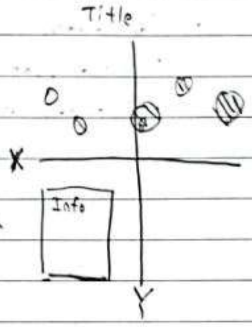
  - Make the graph curved.
  - More types of dataset + Selection (Maybe different countries?)



- Scatter Plot (Pay Gap by Job: 2017)

  → Improvements:

  - Label axes. /
  - Colour ellipses based on ~~p~~ position. /
  - Pop up upon mouseover + display more information ✗

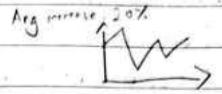


- ~~Th~~ Pie Chart (Tech Diversity)

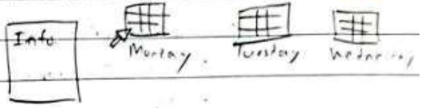
  → Improvements:

  - ~~Pop up upon mouseover~~ + display % /
  - 3D

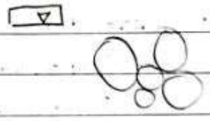
- Adjustable line graph + Temperature (Climate Change)
  - → Improvements
    - ○ Display average increase ✓
      (Changes depending on how zoomed in)



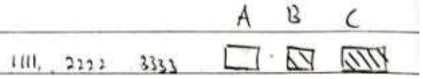
- Wattle charts (UK-Food Days)
  - → Survey → Result Generator (Code it out) ✓
  - → Improvements. ✓
    - ○ Label each day ✓
    - • Display information at side when mousover day. >



- Dynamic Presentation (UK Food. Type).
  - → Improvements.
    - ○ Vertical year selection / Drop down menu. X (Slider). ✓
    - ○ Text stays within the. the ellipse ✓ (Wrap Text)



- Complex waffle chart X (pivot)
  - → Multiple columns of data.
  - → Temperature based on data.



Stage 2 For Finals.

Date No.

Improvements to be made.

- Stacked Bar Chart
  - → Display exact % and pop up upon mouseover
  - → Include a drop down selection.

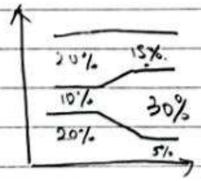
- Line Graph
  - → Multiple lines of data for different countries.

- Scatter plot
  - → Pop up Information box upon mouseover.

- Pie Chart
  - → Pop up upon mouseover

- Waffle chart
  - → Display information at side ~~whe~~ when mouseover days.

- Stacked area chart.
  - → Multiple [illegible] proportions.
  - → Compare 2 time period

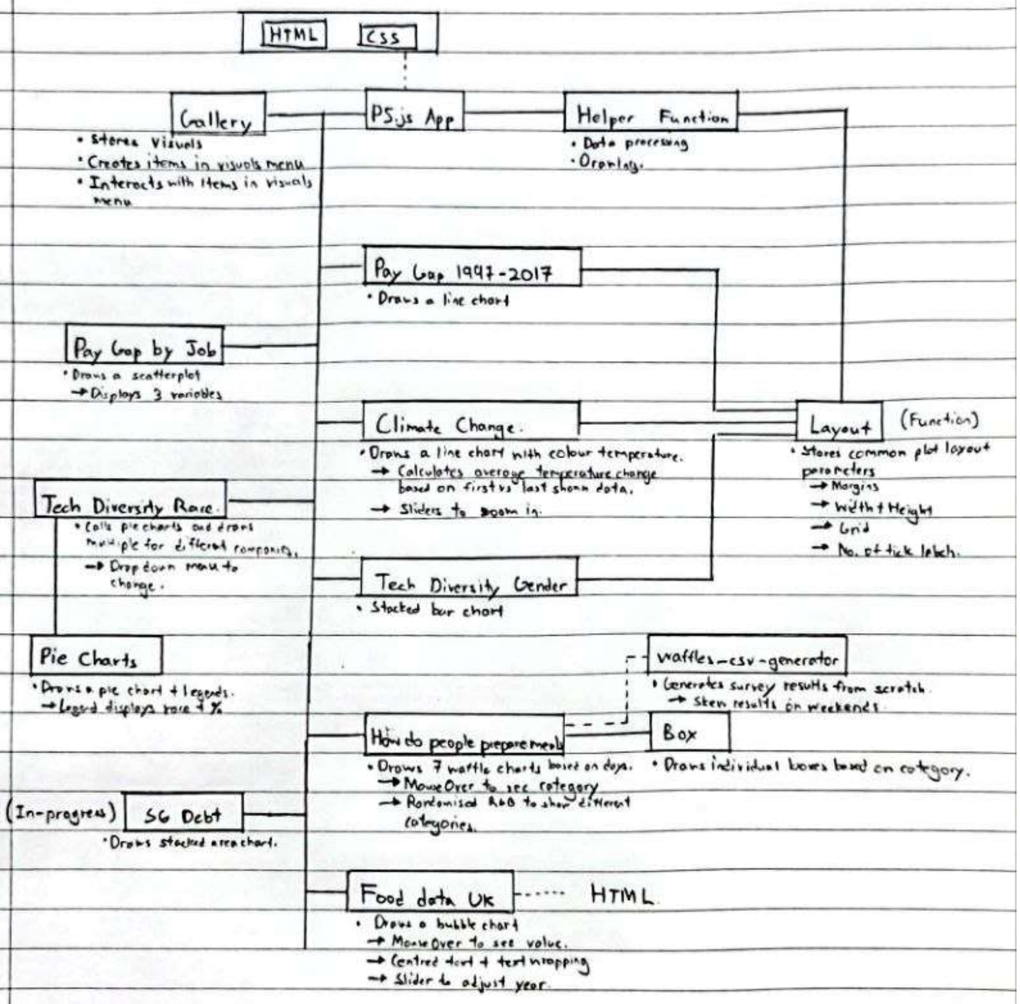


FALCON

Current Site map

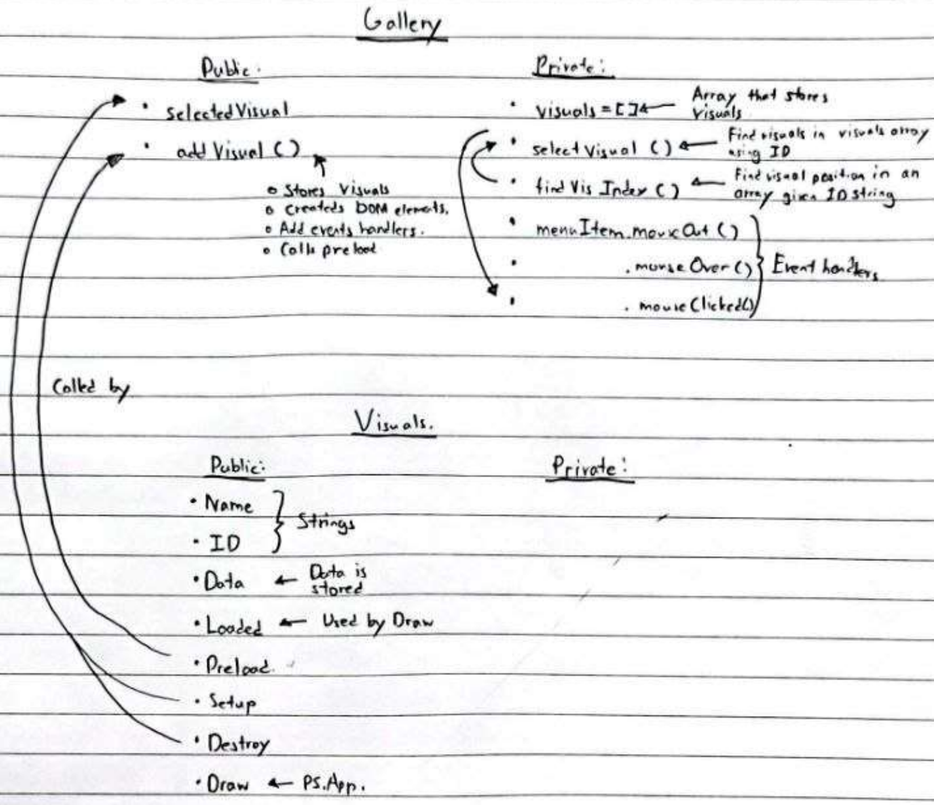
Date No

PS.JS PS.Dom


HTML
CSS
Gallery
• Stores Visuals
• Creates items in visuals menu
• Interacts with Items in visuals menu
PS.js App
Helper Function
• Data processing
• Drawing.
Pay Gap 1997-2017
• Draws a line chart
Pay Gap by Job
• Draws a scatterplot
→ Displays 3 variables
Climate Change.
• Draws a line chart with colour temperature.
→ Calculates average temperature change based on first vs last shown data.
→ Sliders to zoom in.
Layout
(Function)
• Stores common plot layout parameters
→ Margins
→ Width + Height
→ Grid
→ No. of tick labels.
Tech Diversity Race.
• Calls pie charts and draws multiple for different companies.
→ Drop down menu to change.
Tech Diversity Gender
• Stacked bar chart
Pie Charts
• Draws a pie chart + legends.
→ Legend displays race + %
waffles-csv-generator
• Generates survey results from scratch.
→ Skew results on weekends
How do people prepare meals
Box
• Draws 7 waffle charts based on days.
→ MouseOver to see category
→ Randomised RGB to show different categories.
• Draws individual boxes based on category.
(In-progress)
SG Debt
• Draws stacked area chart.
Food data UK
HTML
• Draws a bubble chart
→ MouseOver to see value.
→ Centred text + text wrapping
→ Slider to adjust year.

Date No.

Gallery.js



Gallery

Public:

- SelectedVisual
- add Visual ( )
  - Stores Visuals
  - Createds DOM elements.
  - Add events handlers.
  - Calls preload

Private:

- visuals = [ ] ← Array that stores Visuals
- select Visual ( ) ← Find visuals in visuals array using ID
- find Vis Index ( ) ← Find visual position in an array given ID string
- menuItem.mouseOut ( )
- .mouse Over ( )
- . mouse Clicked()

} Event handlers

Called by

Visuals.

Public:

- Name
- ID

} Strings

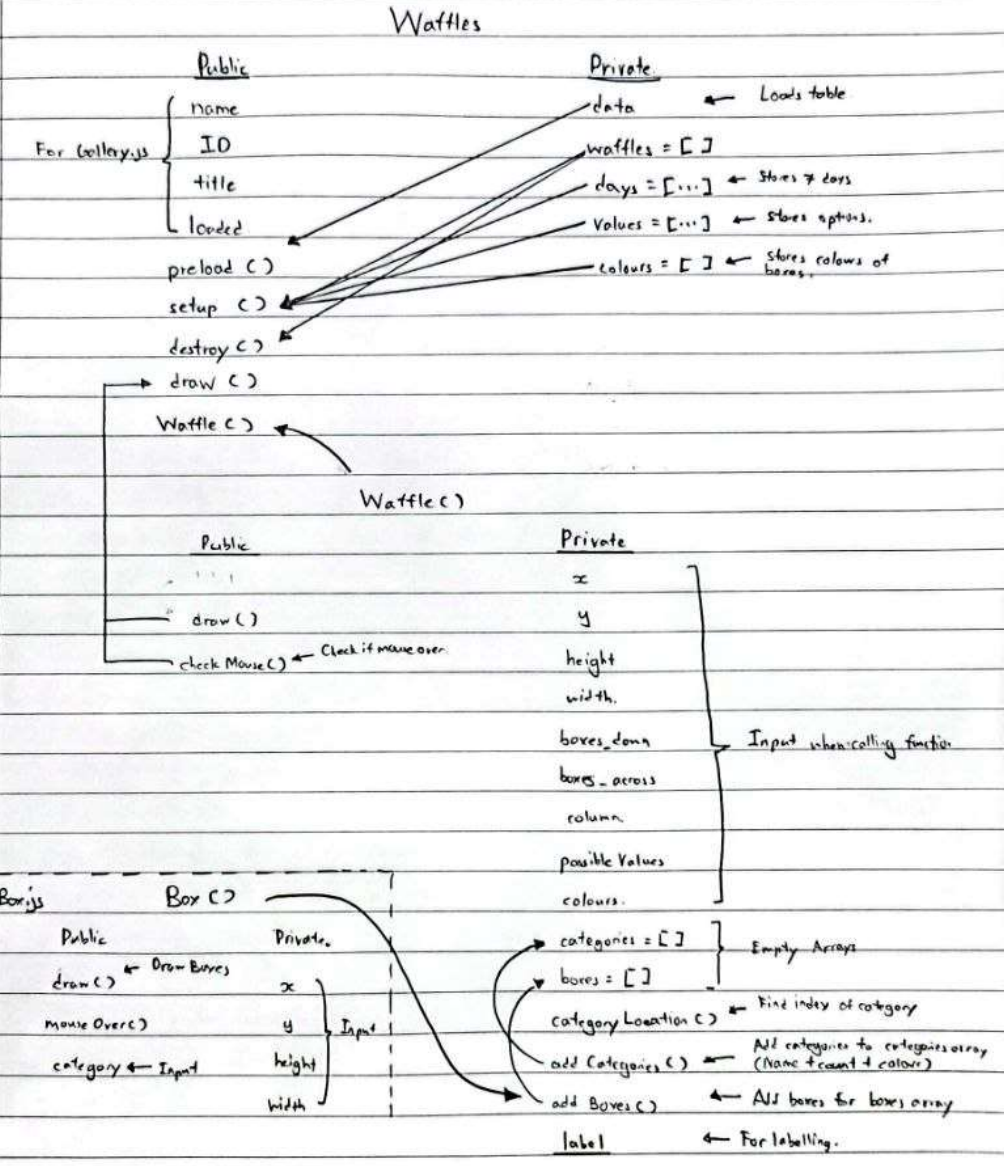
- Data ← Data is stored
- Loaded ← Used by Draw
- Preload.
- Setup
- Destroy
- Draw ← P5.App.

Private:

FALCON

Date No

how-do-people-prepare-meals.js (Waffle Chart)



Date No.

food-data-UK.js.

Food dataUK ( )

Public

name
ID
Title
loaded
} Gallery

preload ( )

setup ( )

destroy ( )

draw ( )

Bubble ( )

updates current Selected year → changeYear ( )

Private.

data ← To store Table data

bubbles = [ ] ← Store bubble objects

map Amt ← Map data values to size.

years = [ ] ← Store years included.

year slider ← creates sliders. ← Interacts with HTML.

minYear
maxYear.
} Set range of slider.

Bubble ( )

Public

size
target-size
pos
direction
name
units
color
data
current data
} Variables.

draw ( )

Update bubble position → update ( )

Set current data value. → setData ( )

Used to display values. → mouseOver ( )

Calculate number of lines needed → calcLines ( )

wrapText ( )

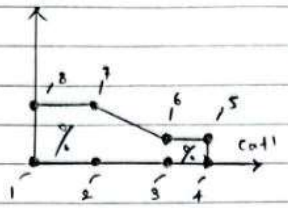
} Wraps text

Private

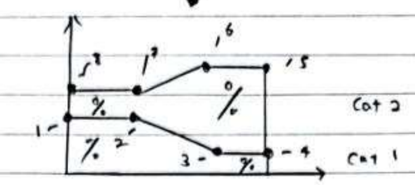
Date No.

sg-dbl

Dist between 1&8 dependent on %
Dist between 5&4 dependent on %



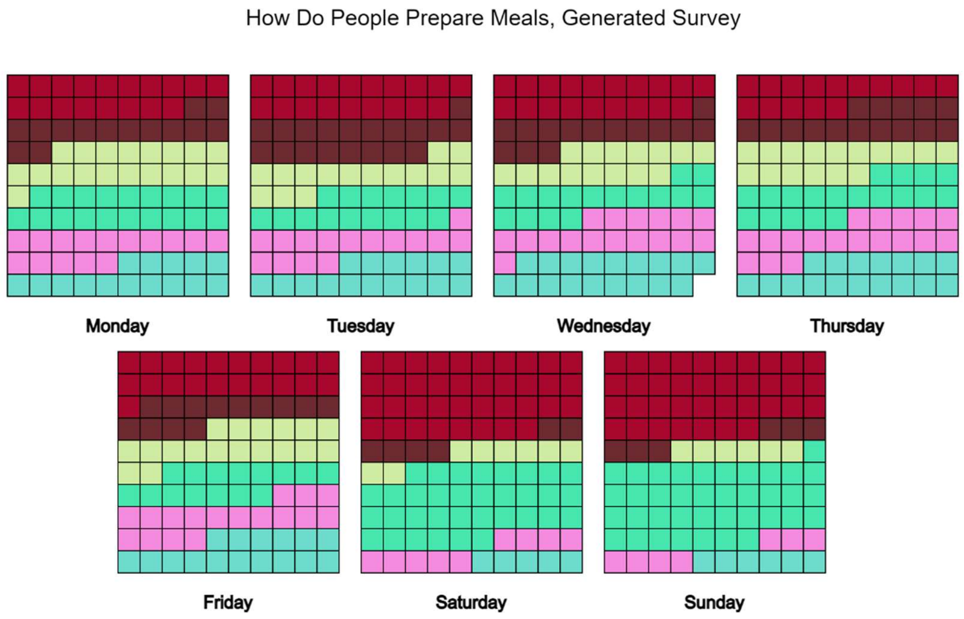
Category 1



Category 2

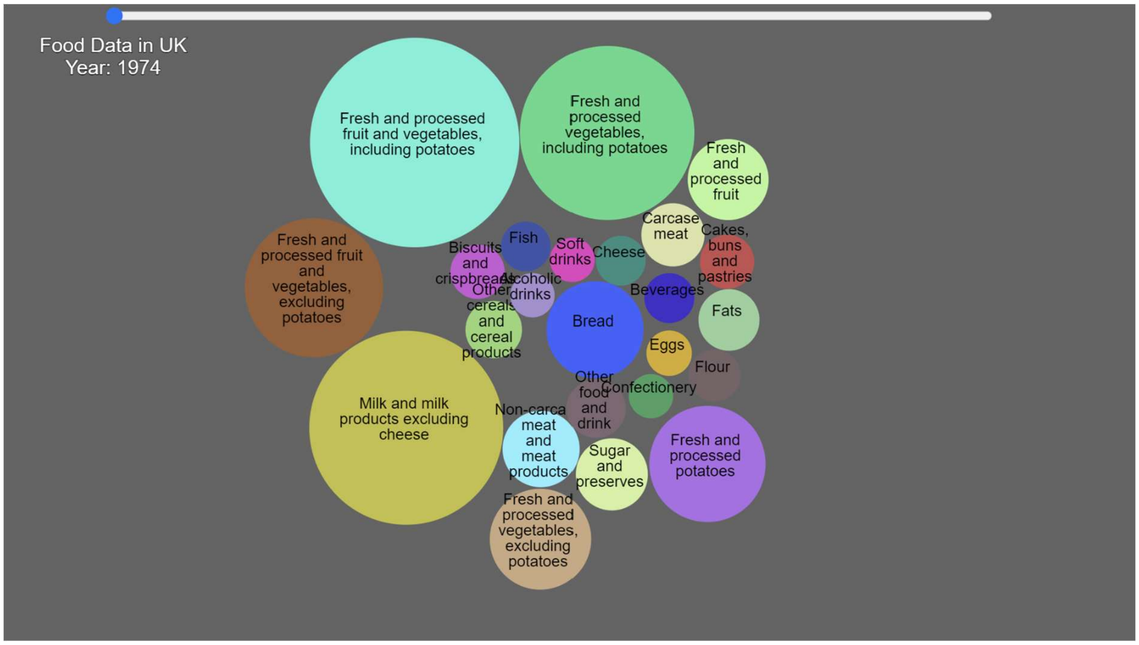
```
for (i=length, i--)   Cat 1 8 = Cat 2 1        for (i=0, i++)
                      Cat 1 7 = Cat 2 2
                      Cat 1 6 = Cat 2 3
                      Cat 1 5 = Cat 2 4

                      Repeat for all categories
```

I will approximate the total number of codes written to be 60% of the progress for the entire ITP2 project, with majority of the difficult extensions (Waffles and Bubbles) being mostly completed. For the waffle chart, I have updated it with the random colour generator that will update with each iteration of the categories, as well as coding out a CSV file generator to generate the survey results required for the waffle chart, where skewed data is generated for the weekends.

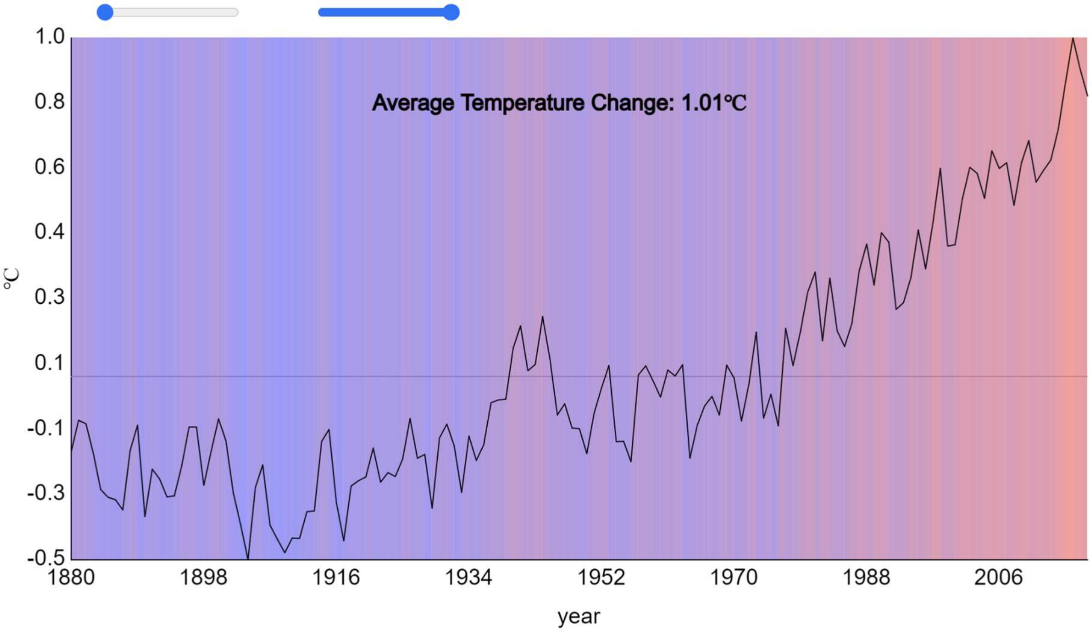


I plan to incorporate a pop up information box for the Waffle chart, which will contain the rest of the information present in the CSV file.

For the Bubble chart, I have incorporated a text wrapping to the bubble, such that the text stays within the ellipse as well as adapting to any changes in size to the ellipses. I have also incorporated a slider in the place of the year-selection buttons to make the visualisation more organised and intuitive, Lastly, I have incorporated a mouseOver function to display the values of each bubble along with their units.



For the provided Line graph with colour temperature, I have incorporated an "average temperature change" calculation to the chart that adjusts according to years being displayed, and will automatically update when the slider is shifted.

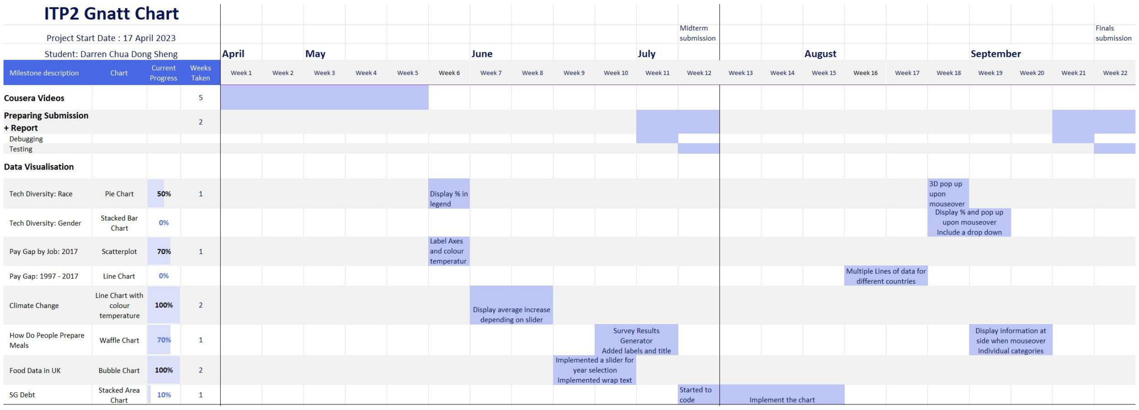


Despite starting on the stacked area chart, it is currently incomplete and does not work, I plan to finish the Stacked area chart, where a comparison between the percentages of different categories between 2 different time periods will occur.

3) I believe that I can split rest of the time into multiple segments, where I have dedicated a certain number of weeks to each individual chart. The amount of time I dedicated to a specific chart will depend on the complexity of the chart and my prediction of the amount of time that will be spent on the task.

   For the next 8 weeks after the midterms, up until the start of September, I will be focusing on coding and finishing the planned improvements to the visualisations, dividing the tasks into bite sized chunks for each week. For the last 2 weeks before the final submission I will spend the time cleaning up the codes, debugging any errors within the files as well as testing the app, all these while finishing up the submission requirements for the project itself.

   Reference to the Gnatt Chart can be made to see how I split the time between the individual visualisation within the 8 weeks previously mentioned.



ITP2 Gnatt Chart

Project Start Date : 17 April 2023

Student: Darren Chua Dong Sheng

				April		May				June				July
Milestone description	Chart	Current Progress	Weeks Taken	Week 1	Week 2	Week 3	Week 4	Week 5	Week 6	Week 7	Week 8	Week 9	Week 10	Week 11
Cousera Videos			5	▨	▨	▨	▨	▨						
Preparing Submission + Report			2											▨
Debugging														▨
Testing														
Data Visualisation														
Tech Diversity: Race	Pie Chart	50%	1						Display % in legend					
Tech Diversity: Gender	Stacked Bar Chart	0%												
Pay Gap by Job: 2017	Scatterplot	70%	1						Label Axes and colour temperatur					
Pay Gap: 1997 - 2017	Line Chart	0%												
Climate Change	Line Chart with colour temperature	100%	2							Display average increase depending on slider	▨			
How Do People Prepare Meals	Waffle Chart	70%	1										Survey Results Generator Added labels and title	▨
Food Data in UK	Bubble Chart	100%	2									Implemented a slider for year selection Implemented wrap text	▨	
SG Debt	Stacked Area Chart	10%	1											

	Midterm submission										Finals submission
				August				September			
Milestone description	Week 12	Week 13	Week 14	Week 15	Week 16	Week 17	Week 18	Week 19	Week 20	Week 21	Week 22
Cousera Videos											
Preparing Submission + Report	▨									▨	▨
Debugging										▨	
Testing	▨										▨
Data Visualisation											
Tech Diversity: Race							3D pop up upon mouseover				
Tech Diversity: Gender							Display % and pop up upon mouseover Include a drop down	▨			
Pay Gap by Job: 2017											
Pay Gap: 1997 - 2017					Multiple Lines of data for different countries	▨					
Climate Change											
How Do People Prepare Meals								Display information at side when mouseover individual categories	▨		
Food Data in UK											
SG Debt	Started to code	Implement the chart	▨	▨							

4) External sources:

For stacked area chart inspiration:

- chang, A. (no date) Alvinschang/P5-stacked-area-chart: Using the p5.js library for the processing language, this lets you simply alter the data file to create the chart., GitHub. Available at: https://github.com/alvinschang/p5-stacked-area-chart (Accessed: 17 July 2023).

For code references:

- **P5.JS**

  p5.js (no date) home | p5.js. Available at: https://p5js.org/ (Accessed: 17 July 2023).

- **W3Schools**

  Javascript string indexof() (no da te) JavaScript String indexOf() Method. Available at: https://www.w3schools.com/jsref/jsref_indexof.asp (Accessed: 17 July 2023).

- **Mozilla Developer Network**

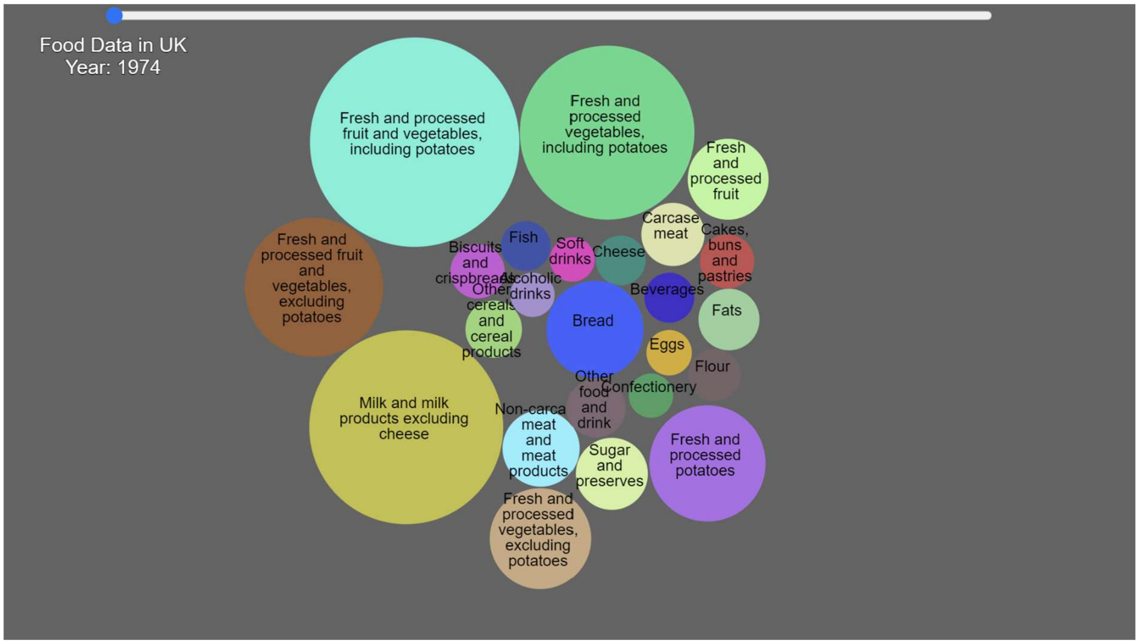
  MozDevNet (no date) Datatransfer: Setdata() method - web apis: MDN, Web APIs | MDN. Available at: https://developer.mozilla.org/en-US/docs/Web/API/DataTransfer/setData (Accessed: 17 July 2023).

- **Coursera**

  Coursera (no date) Coursera. Available at: https://www.coursera.org/degrees/bachelor-of-science-computer-science-london/home (Accessed: 17 July 2023).

7) I believe that the implementation of the slider, the value and units showing upon mouse over as well as text wrapping in the bubbles chart is exceptional and original. Combining all the buttons into one will go a long way in making the chart less overwhelming and was not previously suggested, therefore I believe that it is original.

While showing the value and units required extensive data and array manipulation, and the text wrapping required the breaking of strings into multiple components, as well as lots of calculations for the number of lines. I believe these are exceptional.


Food Data in UK
Year: 1974
Fresh and processed fruit and vegetables, including potatoes
Fresh and processed vegetables, including potatoes
Fresh and processed fruit
Carcase meat
Cakes, buns and pastries
Fresh and processed fruit and vegetables, excluding potatoes
Fish
Biscuits and crispbreads
Soft drinks
Cheese
Alcoholic drinks
Other cereals and cereal products
Beverages
Bread
Fats
Eggs
Flour
Other food and drink
Confectionery
Milk and milk products excluding cheese
Non-carca meat and meat products
Sugar and preserves
Fresh and processed potatoes
Fresh and processed vegetables, excluding potatoes