ASHLEY JAMES DOWIE

sit 120 aSSEMENT 2: pRACITAL pORTFOLIO

Table of Contents

[Marking Justification 2](#_Toc78012048)

[Week 1 2](#_Toc78012049)

[Reflections 2](#_Toc78012050)

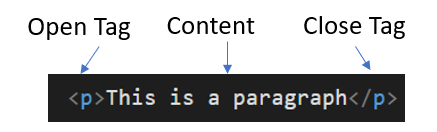
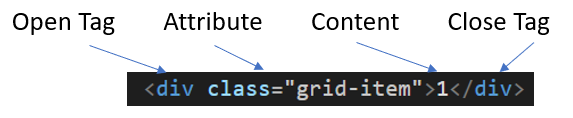
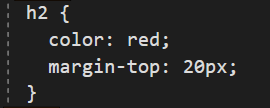
[Practical Tasks 2](#_Toc78012051)

# Marking Justification

# Week 1

## Reflections

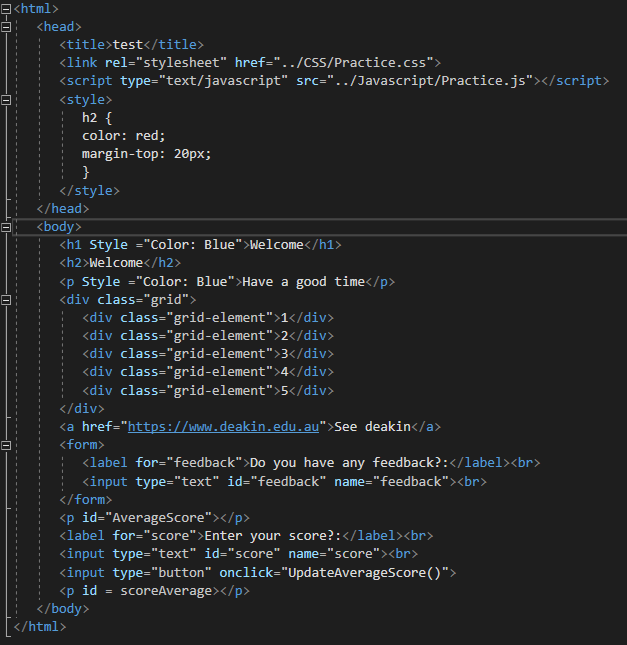
This week was the first week of the unit. As this week was the first week of the unit, we covered the unit structure, assessments and an outline of the content that will be covered. In addition to this we also were introduced to the fundamental building blocks of a web applications.

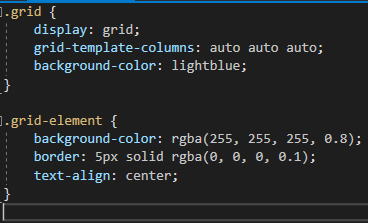
* HTML: Hyper Text Markup Language. HTML gives us the ability to outline the structure and content of our website. HTML is the main building block of any website as all websites require HTML to exist. A HTML document is made up entirely of elements
  + Elements: A HTML element makes up the content of a HTML page. Every element has a tag
    - Tag: A tag defines what type of element a HTML element is. Html tags can be structured in different ways depending on the tag. The below tags are both valid. Tags all begin with an opening tag such as <TAG>, some are then followed by content and then closed by a closing tag </TAG>
      * A paired tag  
         
      * An unpaired tag  
        
      * Tags can also contain further instructions within the opening tag known as attributes.  
        
* CSS: Cascading Style Sheets. CSS is used to modify appearance of websites and web applications. The structure of CSS is based on selectors and declarations.
  + Declarations are key values pairs which have a property and a value formatted as property: value. The property describes the describes the aspect to change such as colour, size, and many other things. The value assigns a value to this property.
  + Selectors contain declarations within them. The selector provides an instruction as to which elements the declarations should apply to A selector with inline CSS is not required as inline CSS is written as CSS declaration appearing within a HTML tag as an attribute as only applies to that single HTML element.
    - This is an example of inline CSS  
      
    - This is an example of a CSS selector The below selector would apply to any element that has the <h2> tag.  
      
* JavaScript: JavaScript is used to make web applications respond to user input. As JavaScript is a programming language, it is not as easy to succinctly define its syntax. JavaScript like most programming languages consists of objects, variables and functions. JavaScript is capable of adding, removing and modifying html elements in response to user input.

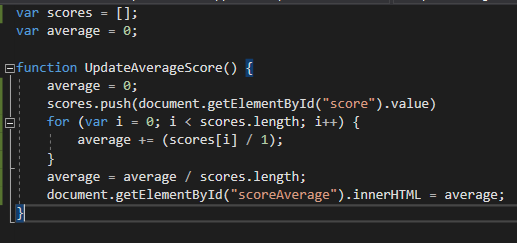
## Practical Tasks

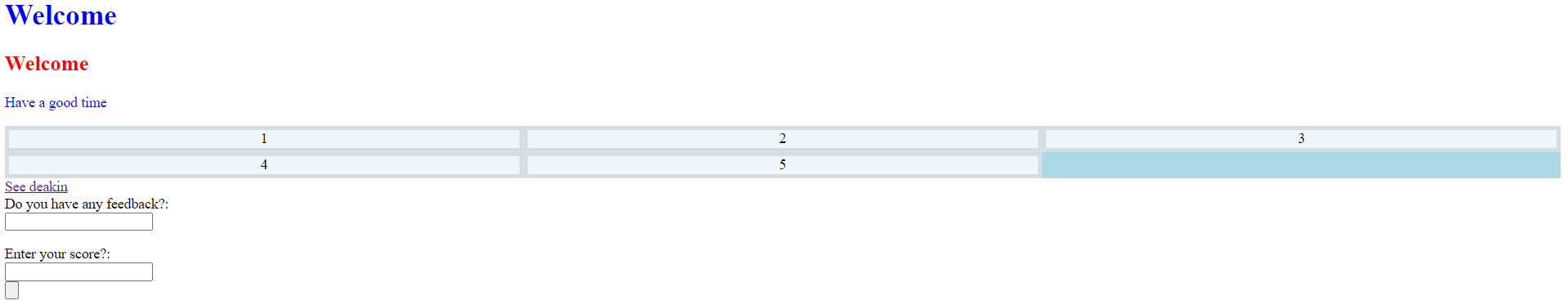
### Task 1-3

When undertaking this task, I was getting strange values when trying to calculate the student score average. I noticed I could fix this by changing scores[i] to scores [i]/1. I did not understand why this worked, only that it did.





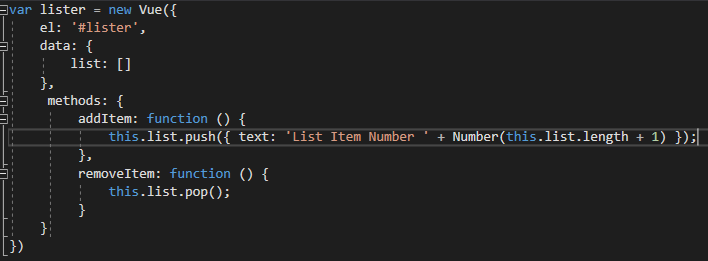




### Task 4

I was getting strange values when pushing new items to the list. Later I realised that the reason for this was because the values were acting as text addition rather than numerical addition. I resolved this by wrapping the values being added in Number() to convert them to a number after reading this documentation [Number - JavaScript | MDN (mozilla.org)](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Number) This helped me to realise what was happening with my previous issue that I resolved by changing score[i] to score[i]/1.

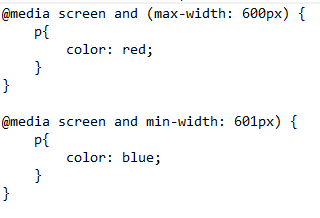




# Week 2

## Reflection

This week we learnt about what responsive websites are and how to create them. Responsive websites are websites which can respond to the size and capabilities of the device being used. A responsive website will respond/ change, so they display optimally on different screen sizes and shapes. Below is a list of ways to make a website more responsive:

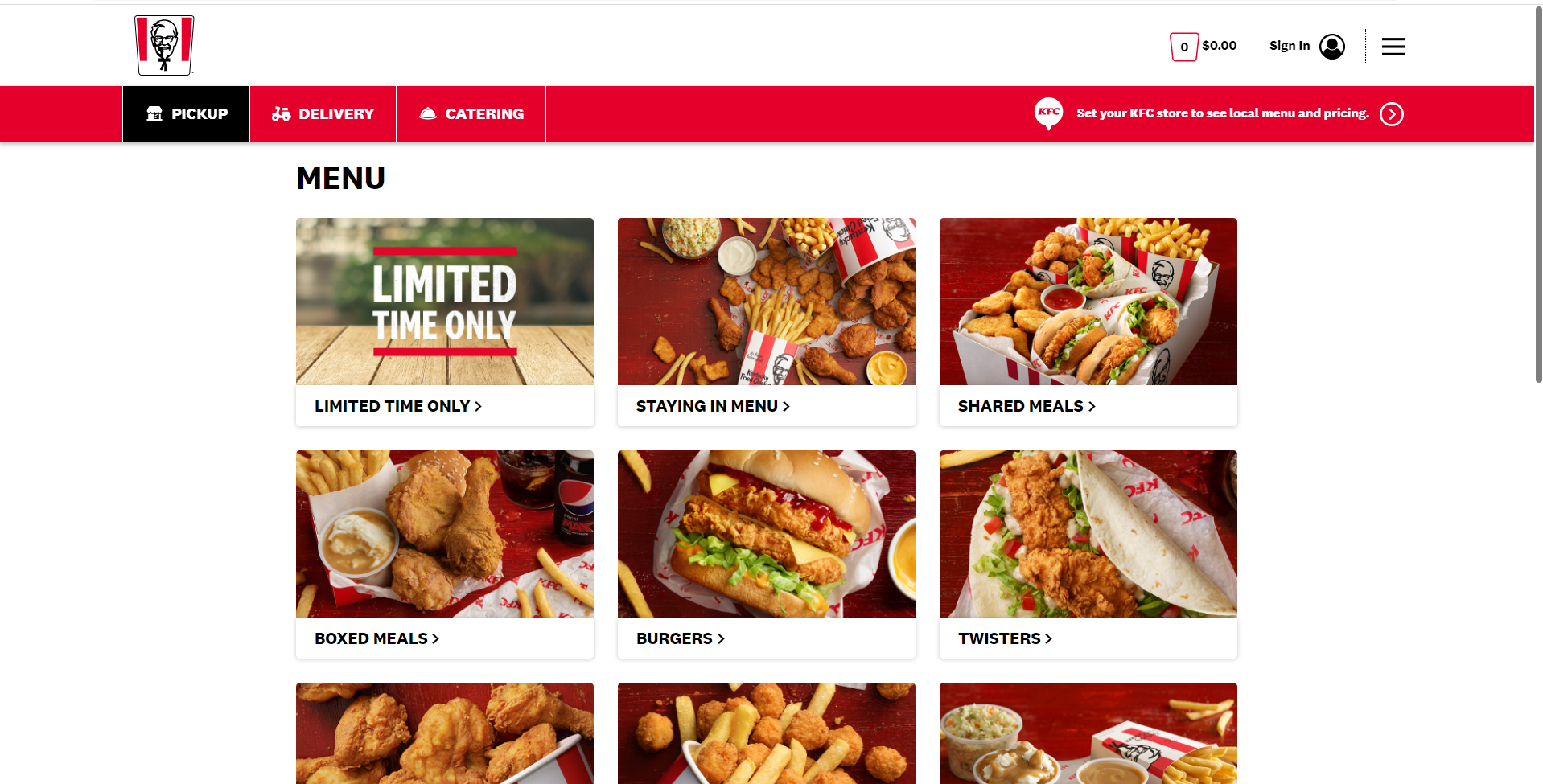
* Use the viewport tag in the html document. Using the viewport tag till allow the content of the website to be displayed in a way that fits the specific devices screen. The viewport tag is written as <meta name="viewport" content="width=device-width, initial-scale=1"> and should be included in the HTML head.
* Do not set content sizes as absolute values such as pixels, instead use percentage so that the data will scale with changes in screen size. This is handled within CSS
* Use breakpoints to determine how the display of the page should change at certain size ranges. Media queries are written in CSS, they are similar to if else statements. A particular screen size in pixels (normally width) is set as the point that if the screen size is above this amount, certain CSS declarations will be user, whereas if the screen size is below this point, other CSS declarations will be used. The following is an example of a media tag that will display paragraphs as blue if the screen is wider than 600 pixels, otherwise paragraphs will be red.   
  

In addition to what responsive website design is and how to achieve it, we also learnt about user stories UX and UI.

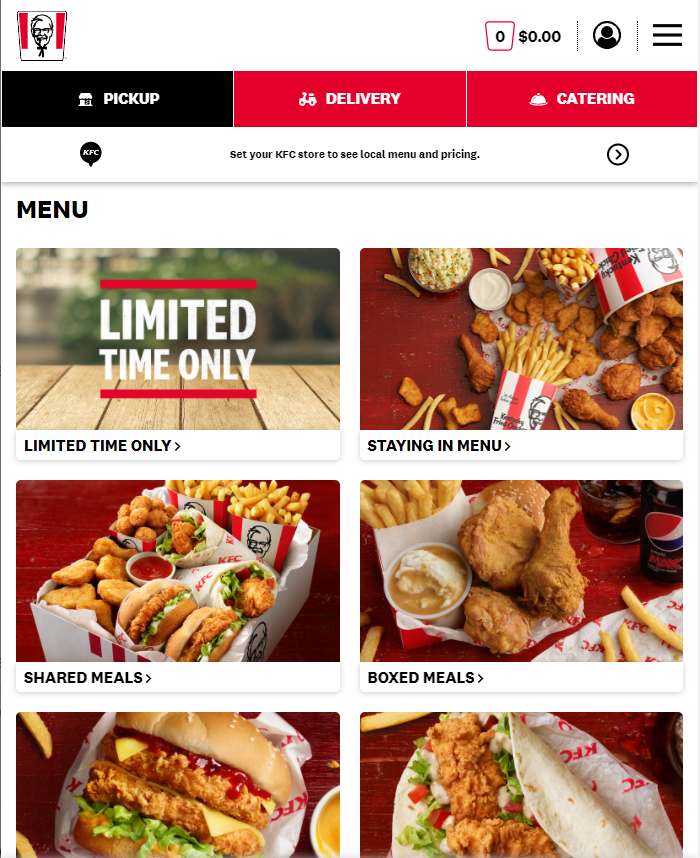
* User stories: User stories (also known as personas) are a description of an imaginary person who uses will use the website. User stories describe how the user likes to interact with websites and who their expectations, needs and desires of the website are. Stories of different types of users can be combined to get an understanding of user requirements and how to make the websites easier to use for these types of users.
* UX: User experience design is about the feelings of humans when interacting within an organisation/ system. In theory user experience design is not only applicable to applications.
* UI: User interface design is about the look and feel of a product that a user interacts with. User interface design is applicable to software applications.

## Practical Tasks

### Task 1

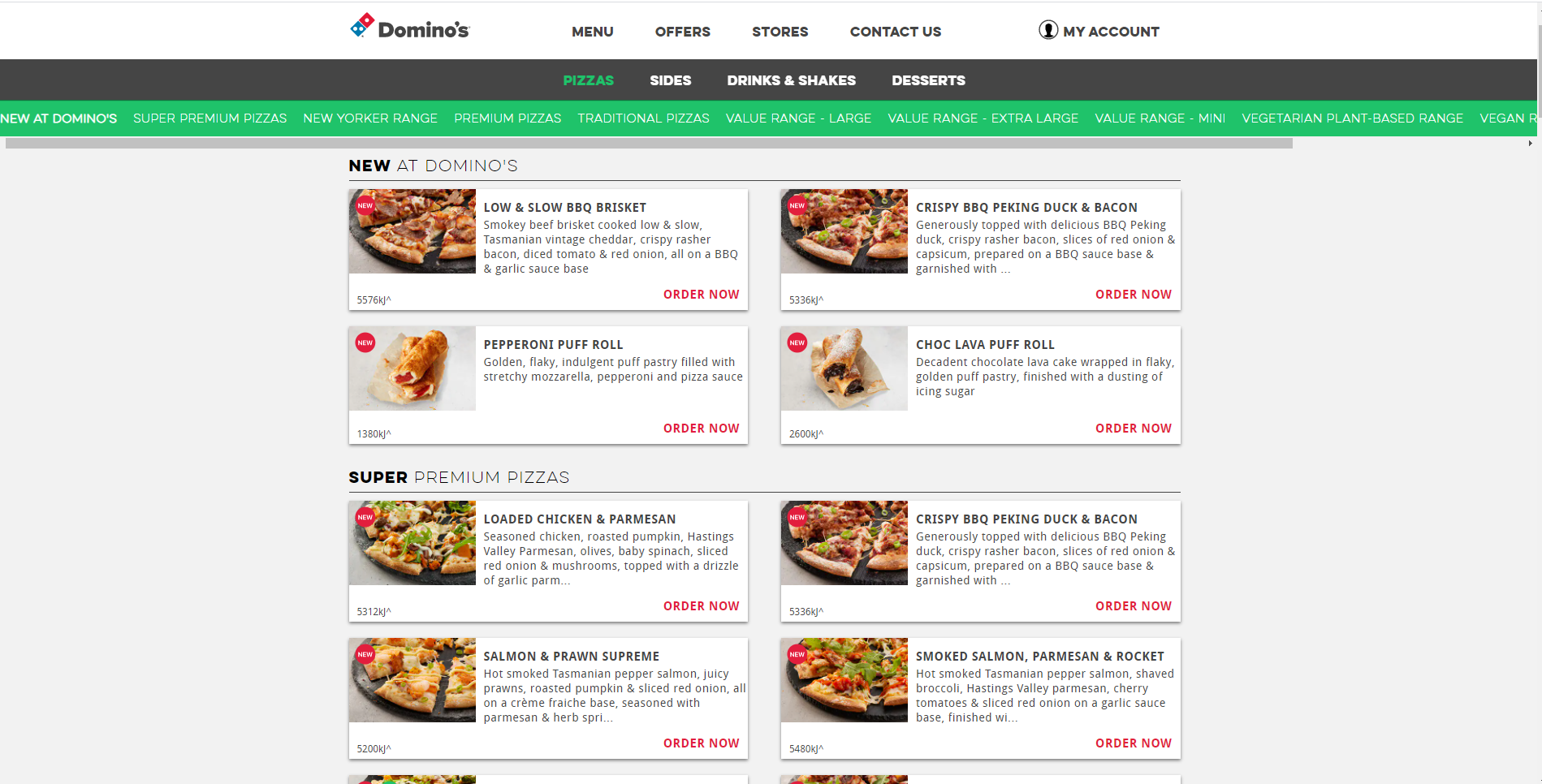
KFC’s website is responsive, particularly its menu. Different menu items are represented by an image. Images respond to image size by reducing image size and the number of images displayed per column as screen size is reduced. Below is the KFC screen at full width:  


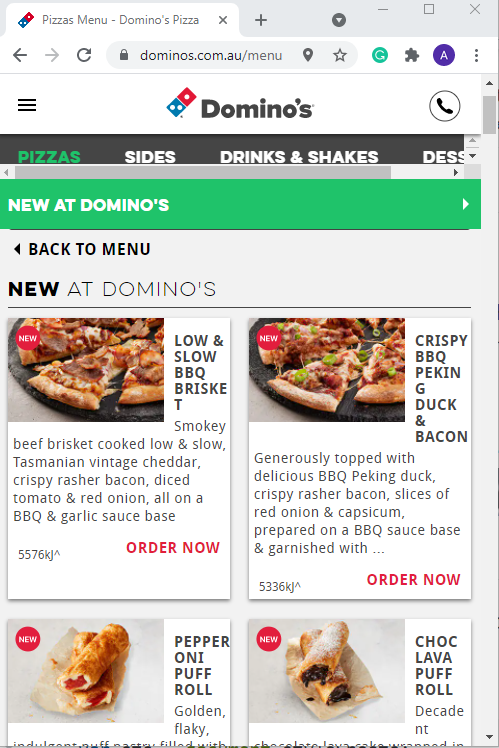
And now here it is at about half width.



As you adjust the screen size you can see where the media controller break points are occurring in the CSS code. KFC achieved this by most likely using flex grids with different break points.

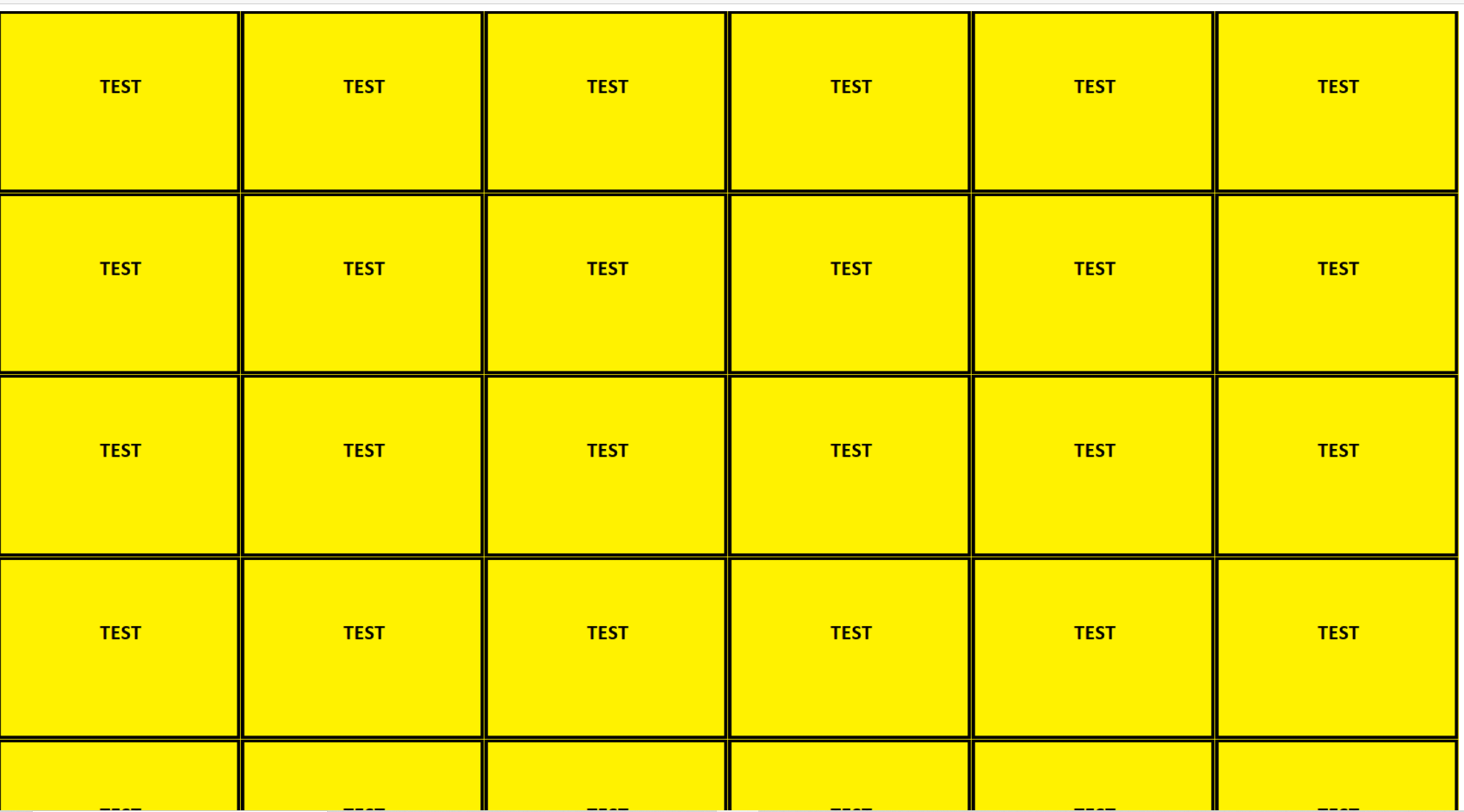
As for the dominos website, the number of columns is not adjusted as the image resolution shrinks. instead, items are stretched vertically. The dominos website can achieve this without stretching the images, as the images take up a small portion of each items content. This means the images themselves to not need to be resized. Most of each item is text, which can be easily spread down vertically.



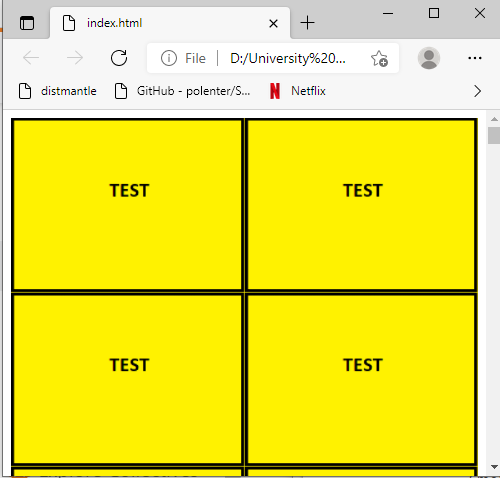


### Task 2

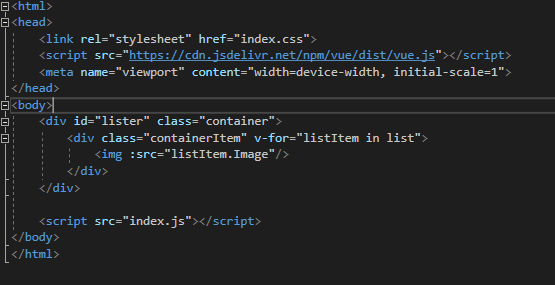
For task 2. I wanted to try and replicate the approach the KFC had taken with their website. I did this by using flex grids and breakpoints with the media controller.

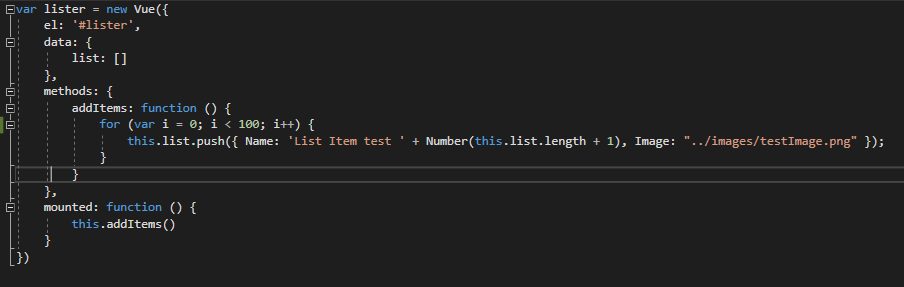
My page looks like this at full screen size:

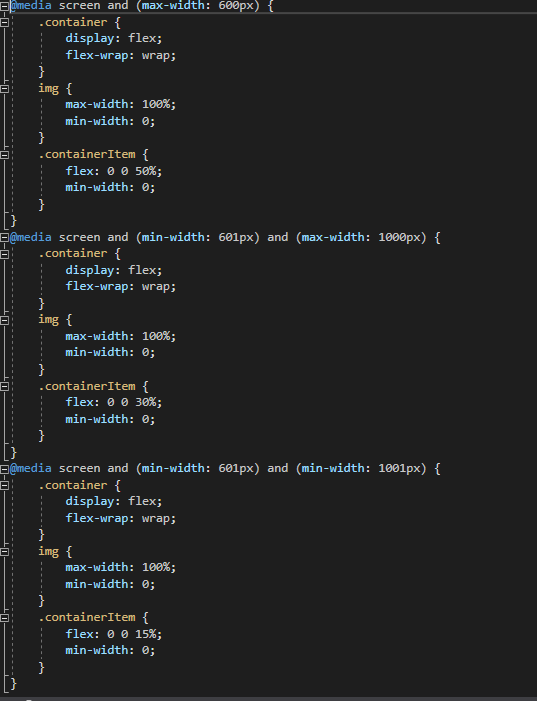
And this at minimum screen size:



I achieved this will the following code:



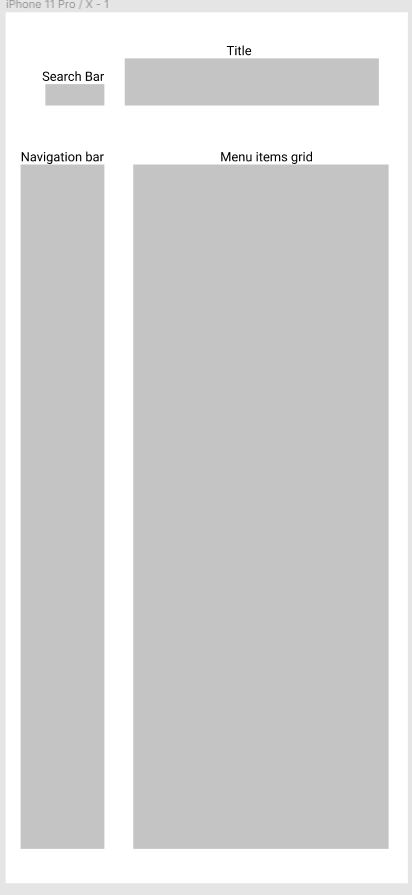




### Task 3

As John is a manual labourer, he gets quite hungry after he finishes work. John often does overtime. As a result of this when he finishes work it is quite late, and he wants food on the go. Unfortunately, the lines at many of the takeaways are quite long around the time that John finishes work. Because of this, it would be better for John to be able to order ahead so his food is ready to pick up as soon as he gets to the takeaway. Also, as John is coming from work, he won’t have access to a computer, only his mobile phone. John doesn’t have the best eyesight and he has little patience for hard to use applications, so the any application he uses needs to be easy to see despite being on the smaller screen of a mobile device.

I used Figma to design the below prototype for a food ordering app. This app displays things horizontally for the most part so that users are not required to scrolls horizontally through the navigation bar or menu. I tried to make the app’s layout similar to what you would see in any other typical mobile app as a user like John would not want to spend time having to adjust to a application that breaks standard conventions.



### Task 4

I added some audio functionality to the website. When the user selects any of the menu items, an audio clips now will play. I achieved this by using the following code.

