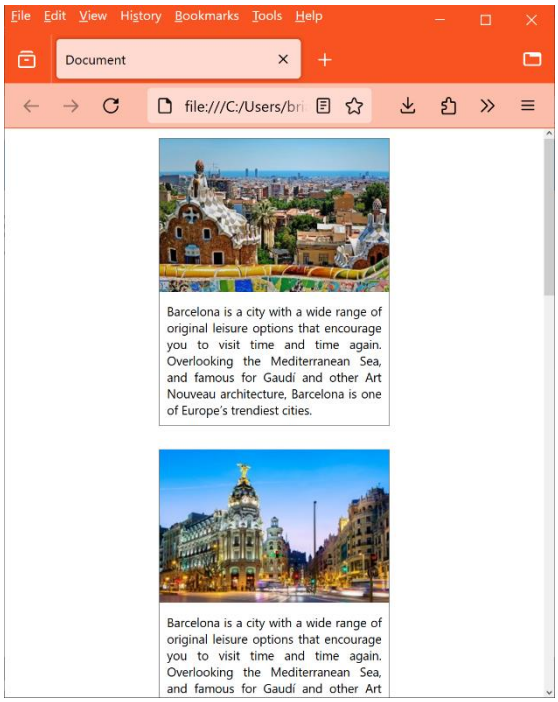
Author: Brian Webberley
Last updated: 2025.09.23

Interactive Web Design

Lab 3: Responsive CSS Grid

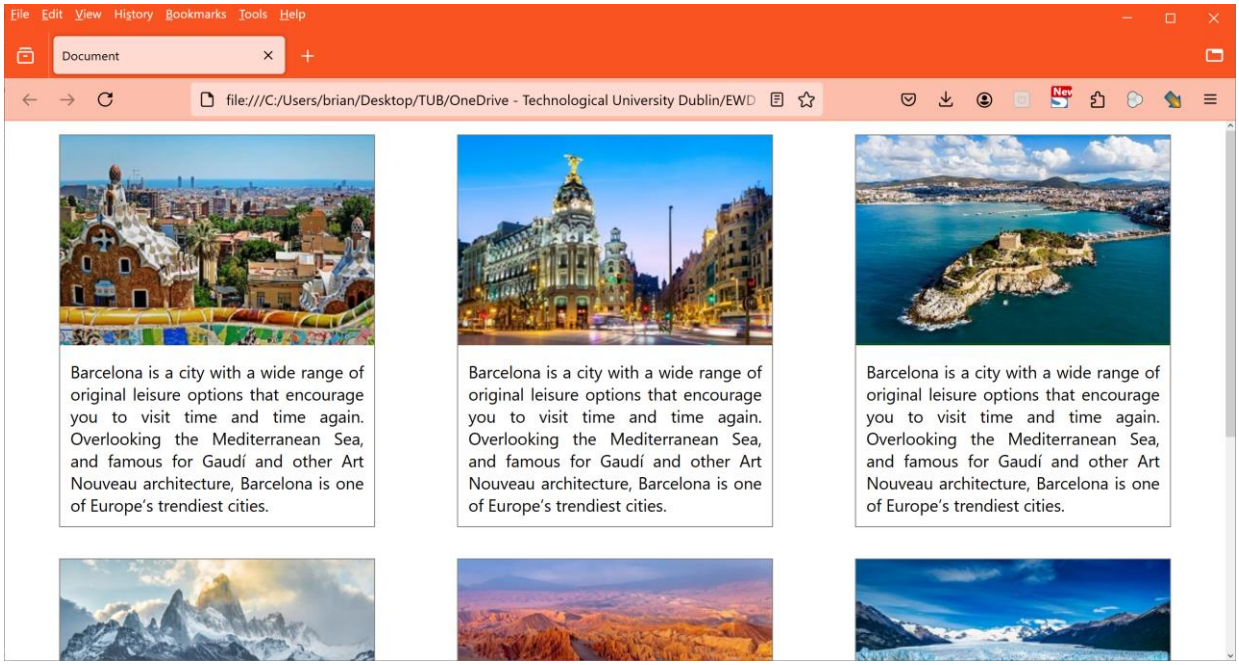
The CSS Grid Layout Module offers a grid-based layout system, with rows and columns, making it easier to design web pages without having to use floats and positioning. (We didn't cover floats and positioning for this reason).



Objective:

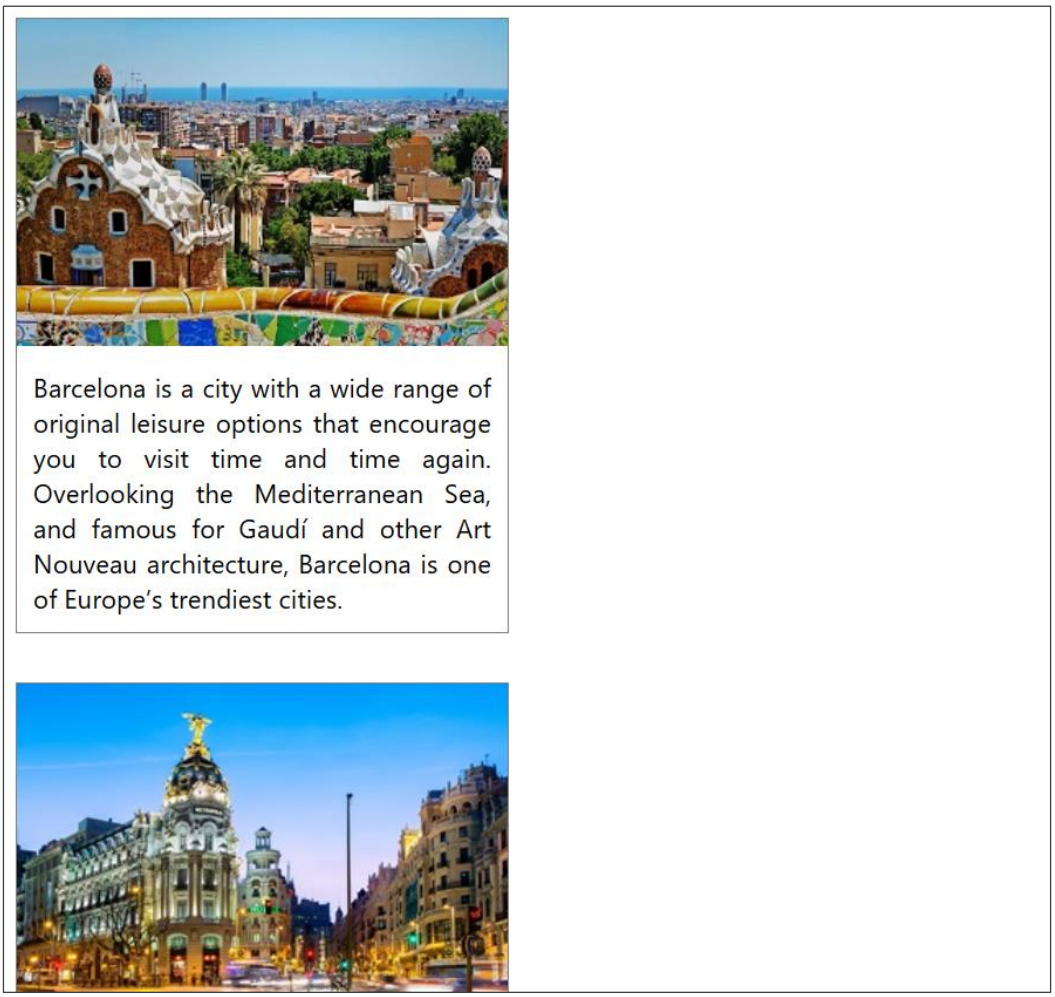
In lab 2 we looked at a creating a 'gallery card'. In this lab we will create a responsive image gallery of six gallery cards, that has two layouts.

The first layout, suitable for a mobile device, will look like the webpage to the right. The second layout, suitable for larger screens will look like the below web page.



Step 1: Create 5 more gallery cards just like in lab 2.

Step 2: Add a 30px margin below each gallery card to space them apart so that they look more like the below screenshot:



Step 3: Put the gallery cards into a 'CSS Grid'

Step 3.1: HTML - Create a Grid Container

```
<section class="grid-container">

    <figure>

        <img src="https://cdn.britannica.com/82/94382-050-20CF23DB/Great-Wall-of-China-Beijing.jpg"
            alt="Placeholder kitteh">
        <figcaption>The Great Wall of China</figcaption>

    </figure>



    <figure>
```

In the above example the containing element is a section. The elements contained within it are the figures, which hold the images. (Side note: the figure tags are, in turn, the containers for the images.)

Step 4: Create the Default, Mobile Layout

In your CSS file, create the default layout for the grid using the below code:

```
.grid-container {
    display: grid;
    grid-template-columns: auto;
    background-color: red;
    justify-items: center;

}
```

Explanation:

`display: grid;` - this turns the container in your HTML file (i.e., *<section>* with a class of *.grid-container*) into a grid

`grid-template-columns: auto;` - this sets the amount of columns. In this example, it's one column.
`grid-template-columns: auto auto;` = 2 columns
`grid-template-columns: auto auto auto;` = 3 columns, and so on.

`background-color: red;` - This is just used to separate the background of the grid from the background of the page. It helps you to distinguish between the two should you wish to style one or the other, you can see where your changes are having an effect. It's just temporary.

`justify-items: center;`

This will center your grid items.

See more here: https://www.w3schools.com/cssref/css_pr_justify-items.php

Step 5: Make the Images Responsive on screens larger than a mobile

We will create a media query that is activated when the viewport size is at least 768 pixels wide. Inside this media query, create a rule that makes each gallery card grid item align so that your 6 gallery cards are align in two rows, with each row having 4 gallery cards. i.e.,

□□□
□□□

The below code tells the CSS code inside the media query to change based on the parameters the media query specifies (i.e. when the viewport width reaches 768 pixels wide). The CSS code that will change any HTML elements inside the class of ***.grid-container*.**

```
@media only screen and (min-width: 768px) {

    .grid-container {
        grid-template-columns: auto auto auto;
    }

}
```

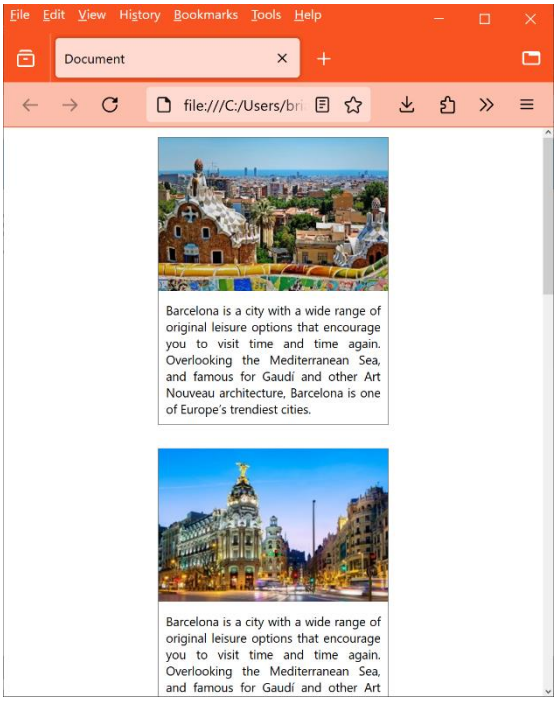
The CSS code is changing from the default:

```
    grid-template-columns: auto;
```

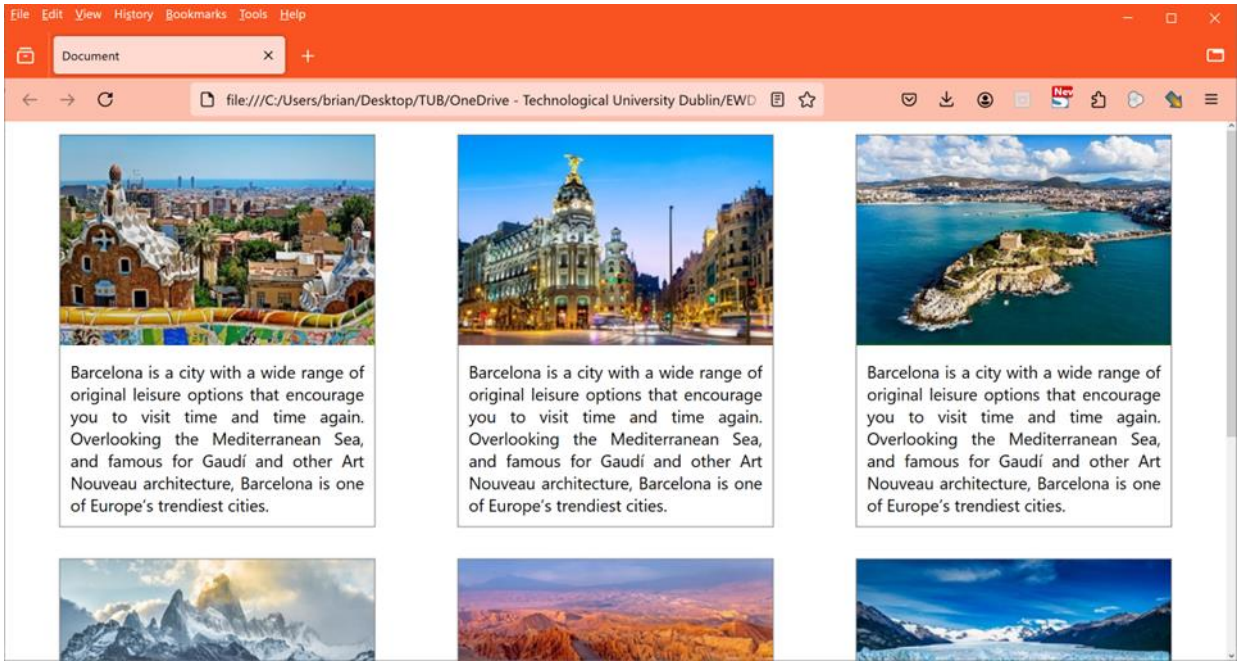
to:

```
    grid-template-columns: auto auto auto;
```

After you save your files, your webpage should look like this when your browser is narrow:



As you widen your browser window, your webpage should switch to a 3x2 layout as shown below.



Step 6: Make the Images Responsive on Laptops / Monitors

You could use the same method to create a media query so that when the viewport width reaches:

576 pixels wide, the webpage will display a grid of 2 images across.

992 pixels wide, the webpage will display a grid of 4 images across.

Checklist / Marking Rubic

Task #	Does the webpage have...	Viewport pixel width	Marks	Complete?
1	6 images vertically stacked	< 768	2.5	
3	2 x 3 image grid	>= 768	2.5	
4	3 x 2 image grid	>= 576	2.5	
5	4 images across, 1 image underneath	>= 992	2.5	
		Total:	/10	