



COMP213 - ASSIGNMENT #4

Instructions

In this assignment, you will create a layout template based on the provided wireframes that you will use to develop your final project. You will employ a **mobile-first** methodology where the styling of the page is based on a mobile layout that adapts to wider screens as well.

View the attached .pdf containing the wireframe diagrams of the mobile view, tablet view, and desktop view that provides a layout of structural and placeholder elements.

Use the attached logo and image as placeholders.

Name your template HTML file **assignment4.html** and your template CSS file **style.css** and upload them along with all dependencies for your project to a folder named 'Assignment4' in your folder on the studentweb server.

Link from your index.html homepage to your 'assignment4.html' file in the Assignment 4 folder.

Requirements

HTML

- Create a <header> with:
 - an image (use the logo placeholder here) and
 - main heading with the words "Main Heading"
 - don't forget the alt tag as well as the width and height for the logo image (**120px width, 100px height**)
- Create a <nav> with:
 - an ordered list and
 - four list items containing links (use href="#")
- Create a <main> area with:
 - two <h2> headings containing sample text
 - two <p> tags containing sample text
- Create an <aside> area with:

- an image (use the image placeholder here). You may wish to give this an image an ID for easier referencing
 - don't forget the alt tag values as well as the width and height attributes for the placeholder image (**300px width, 390px height**)
- Create a <footer> with:
 - some placeholder text for copyright information
- Add the proper <meta> tag to set the width of the viewport to be the width of the browser device screen.
- Link a stylesheet to your webpage.

CSS - Mobile View (0-679px vw)

*Hint – while you are working on your project, you may set a border around your elements to help you see them, then remove before you submit your assignment:

```
head, nav, main, aside, footer {
border: 1px solid red;
}
```

- Set all elements to use the border-box to calculate the width of the element
 - remove all default margins and paddings from all elements
- Set the body to have a dark grey background colour (#3b3b3b). We will see it in the desktop view. Set the structural elements to have a white background color:
 - header, nav, main, aside, footer {


```
background-color:#fffff;
```
- Set the body to have a min-width of 500px to prevent elements from re-sizing below a reasonable level
- Set the header as a flexbox to easily set the logo and text to be vertically aligned. Add a gap of 1em to your header to separate the logo and text
- Target the ol element in your nav section and remove the list styling (bullets) from the ordered list. Center the list items and add a grey background colour behind the links. Use the **descendant element selector** to only select the ordered list element that are nested in the nav element.
- Target the list element in your nav section. Add 10px of padding above and below the list items to provide visual breathing room for the menu items. Add a border on the bottom of the list elements to distinguish them vertically. **Use the descendant element selector** to only select the list item elements that are nested in the nav element.
- Use a **pseudo-selector** to target the first li tag within the nav element to give it a top-border to complete the border in the nav section

- `nav li:first-child{`
 `border-top:1px solid black;`
 `}`
- Add padding to the main section to give the text breathing room:
 - `main {`
 `padding:5%;`
 `}`
- Center the image within the aside element (`text-align:center;`).
- Set the image in the aside section to be 300px wide for the mobile view (height set to auto). Provide a margin of 25px on all sides of the image to give it visual breathing room.
- Add a top border for the footer area and top and bottom padding of 1em. Center the text in the footer area

CSS - Tablet View (680px-1023px)

- Set a `@media` rule for a minimum screen width of 680px for the tablet view
- Set the body element to display as a grid.
 - Create a grid that has 2 columns and 4 rows. The columns should be 1fr in width. Use **grid-template-areas** to create named grid areas as follows:
 - `"header header"`
 - `"nav nav"`
 - `"main aside"`
 - `"footer footer"`
 - You do not need to set `grid-template-rows` (leave them as auto)
- Set the header, nav, main, aside, and footer elements to use the corresponding named grid areas according to the wireframe.
- Set the `ol` element within the nav element (use descendant selector) to be a flexbox, and display the list items in a horizontal row with even spacing around the list items.
- Remove the border from the list items since they are not vertical anymore
- Set the image in the aside section to be 200px wide for the tablet view.
- Set the aside section to be display as a flexbox with a flex direction of column. Align the image so its vertically and horizontally centered in the aside section.

CSS - Desktop View (1024px+)

CSS

- Set a `@media` rule for a minimum screen width of 1024px for the tablet view
- Adjust the body's grid-template areas to have 3 columns and 3 rows according to the wireframe.

- Set the width of the grid columns to be 150px, 2fr, 1fr
 - Remember to set the width of the columns using the grid-template-columns declaration.
 - You do not need to set grid-template-rows (leave them as auto height)
- Use **grid-template-areas** to create named grid areas as follows:
 - “header header header”
 - “nav main aside”
 - “footer footer footer”
- Set the image in the aside section to 300px wide for the desktop view
- Limit the body to 80% width and provide auto margins
- Set the ol tag within the nav element to display as a block
- Set the list items within the nav element to its default block display to make the list items vertical again (use descendant selector to only select list items within the nav element).

Marking Rubric (10 Marks)

HTML

- alt, height, and width attributes exist for both the logo image in the header and placeholder image in the aside sections **(1)**
- correctly added the meta tag to re-size the width of the browser according to the device's width**(1)**

CSS

Mobile View

- Header is set to a flexbox, and the image and main heading text is vertically aligned **(1)**
- Correctly used descendant selector to select li and ol belonging to nav**(1)**

Tablet View

- Correctly set @media rule for a minimum screen size of 680px **(1)**
- Correctly set the aside section as a flexbox and vertically and horizontally aligned the image placeholder in the aside section **(1)**
- Correctly created a 2 column grid with 4 rows according to the wireframe **(1)**

Desktop View

- Correctly set @media rule for minimum screen size of 1024px **(1)**
- Correctly created a 3 column grid with 3 rows according to the wireframe **(1)**

Publishing

- Uploaded all files into a folder named 'Assignment4' and provided a link from index.html to the assignment4.html file **(1)**