WEB REPORT 4

Group: SE-2429

Name: Daryn Musseyev

**Report – Part 1: Media Queries**

**Task 0: Responsive Typography**

In this task, a simple webpage was created with headings (h1, h2) and paragraphs (p).  
Media queries were used to make the text responsive on different devices:

* **Mobile (max-width: 600px):** smaller font sizes for compact screens.
* **Tablet (601px–1024px):** medium font sizes for better readability.
* **Desktop (min-width: 1025px):** larger font sizes for wide screens.

This ensures that text elements remain easy to read across all screen sizes.

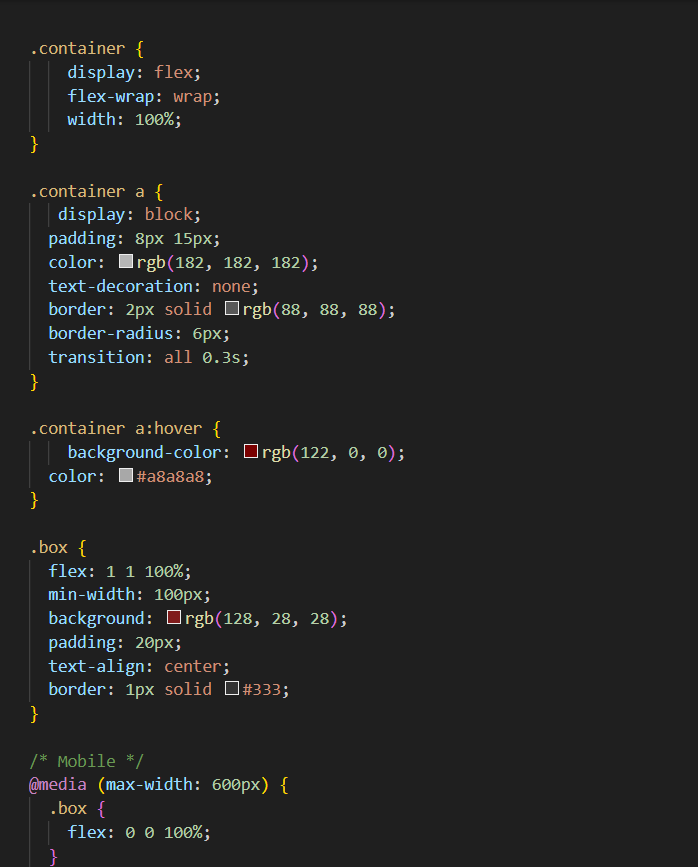


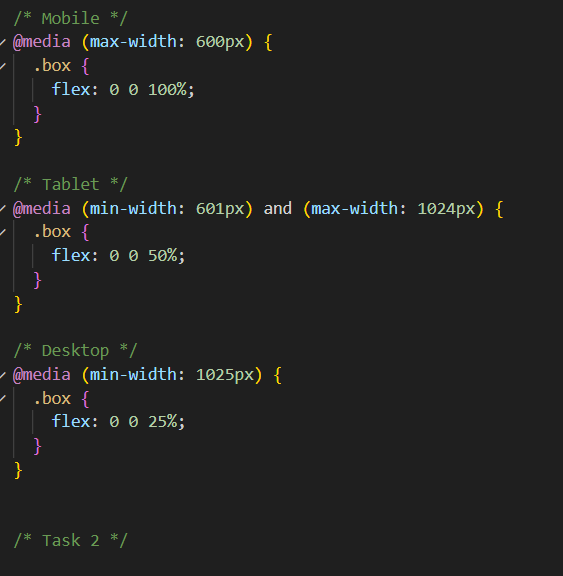
**Task 1: Responsive Layout with Media Queries**

In this task, a responsive layout was built using **only CSS media queries**, without Bootstrap.

The webpage contains **four boxes** (originally three were required) that adapt to the screen width:

* **On Desktop:** all boxes are shown side by side in one row.
* **On Tablet:** two boxes per row.
* **On Mobile:** one box per row (stacked vertically).





**Task 2: Responsive Layout Using Bootstrap Grid**

**Objective**

The goal of this task was to create a responsive webpage using the **Bootstrap Grid System** instead of plain CSS media queries.  
The layout should automatically adjust the number of boxes (links) displayed per row depending on the screen size.

**Implementation Details**

**HTML Structure**

* The page uses the **Bootstrap container** and **row/column** layout.
* Each column contains a .box with a link to a specific task (Task 1–4).
* The column classes define how many boxes appear per row:



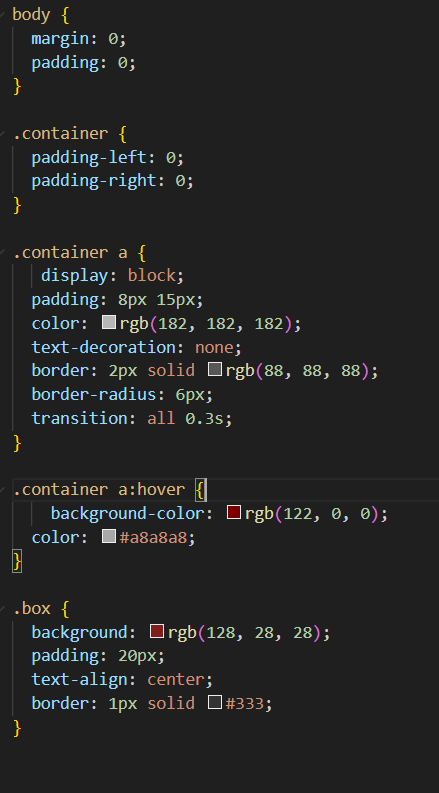
* + col-12 → one box per row on **mobile**.
  + col-md-6 → two boxes per row on **tablet**.
  + col-lg-3 → four boxes per row on **desktop**.
* The second version of the code also wraps the boxes inside a <nav class="navbar">, showing how Bootstrap components can be used together with the grid system.



**CSS Styling**

A separate file (style2.css) is used for custom styling:

* **Reset margins and paddings** for consistent layout:
* body { margin: 0; padding: 0; }
* **Box design:**  
  Each .box has a dark red background, border, and centered text.
* **Link styling:**  
  The <a> elements have borders, rounded corners, and a hover effect that changes background and text color for better interactivity.
* **Container adjustment:**  
  Default Bootstrap container paddings were removed to align boxes neatly.



**Task 4: Responsive Portfolio Page (Bootstrap + Media Queries)**

**Objective**

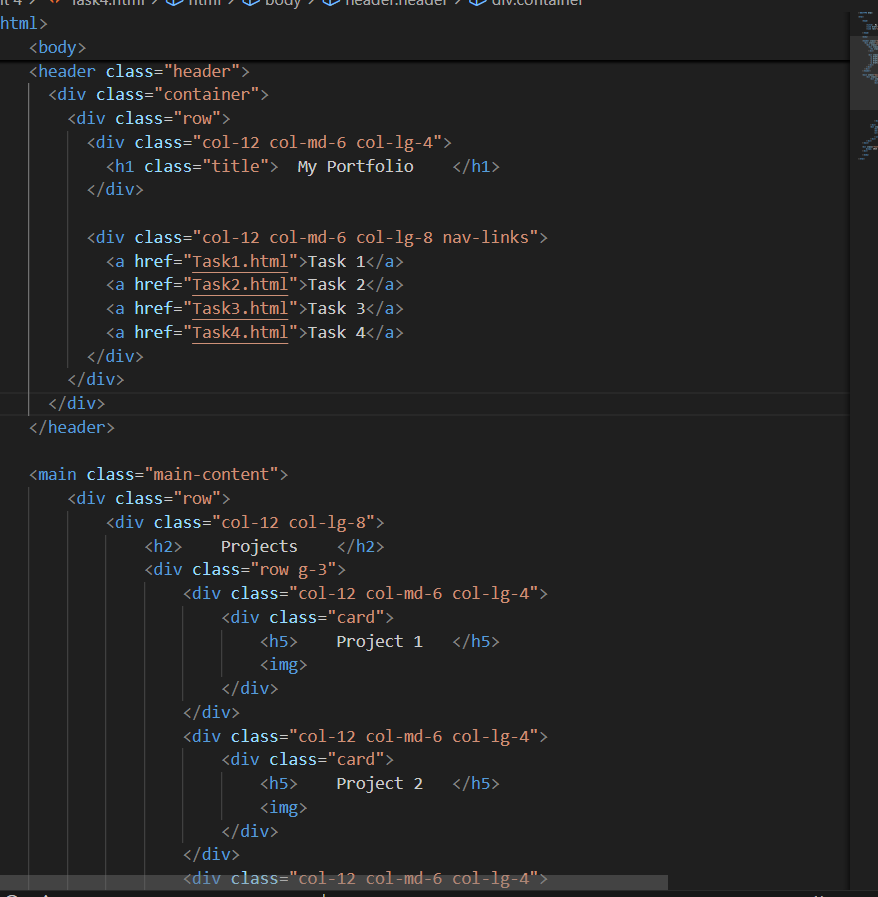
The goal of this task was to create a **responsive personal portfolio page** using both **Bootstrap’s grid system** and **custom CSS media queries**. The page should remain readable and visually appealing across mobile, tablet, and desktop devices.

**Implementation Details**

**1. Structure (HTML)**

The HTML file follows a clear three-section layout:

* **Header** – Contains the portfolio title and navigation links (Task 1–4).  
  Bootstrap’s grid system divides the header into two main columns:
  + Left (col-12 col-md-6 col-lg-4) for the title.
  + Right (col-12 col-md-6 col-lg-8) for navigation links.
* **Main Section** – Divided into two main parts:
  + **Left side (col-12 col-lg-8):** Displays “Projects” with responsive Bootstrap cards arranged in a grid (row g-3).
  + **Right side (col-12 col-lg-4):** Sidebar with “About Me” and contact information.
* **Footer** – Fixed to the bottom of the viewport, ensuring it remains visible at all times.



**2. Styling (CSS)**

The style4.css file applies both Bootstrap-friendly and custom styling:

* **Header:**  
  A dark red background with white navigation links. Links change color and background on hover.
* **Cards:**  
  Black cards with smooth hover lift effect (transform: translateY(-5px)), creating a dynamic presentation for projects.
* **Sidebar:**  
  Distinct black background with padding and rounded corners, clearly separated by a left border.
* **Footer:**  
  Fixed at the bottom using:
* position: fixed;
* bottom: 0;
* left: 0;
* width: 100%;



Styled with red tones to match the header and maintain visual balance.

* **Responsive Design:**  
  Custom media queries adjust font sizes, padding, and layout behavior for:
  + **Mobile (max-width: 600px)** – centered text and smaller navigation.
  + **Tablet (601px–1024px)** – medium font sizes and improved spacing.
  + **Desktop (1025px and above)** – full layout with balanced proportions.

