**WEB REPORT 5**

**Group: SE-2429**

**Name: Daryn Musseyev**

**Part 1. Basic HTML Forms**

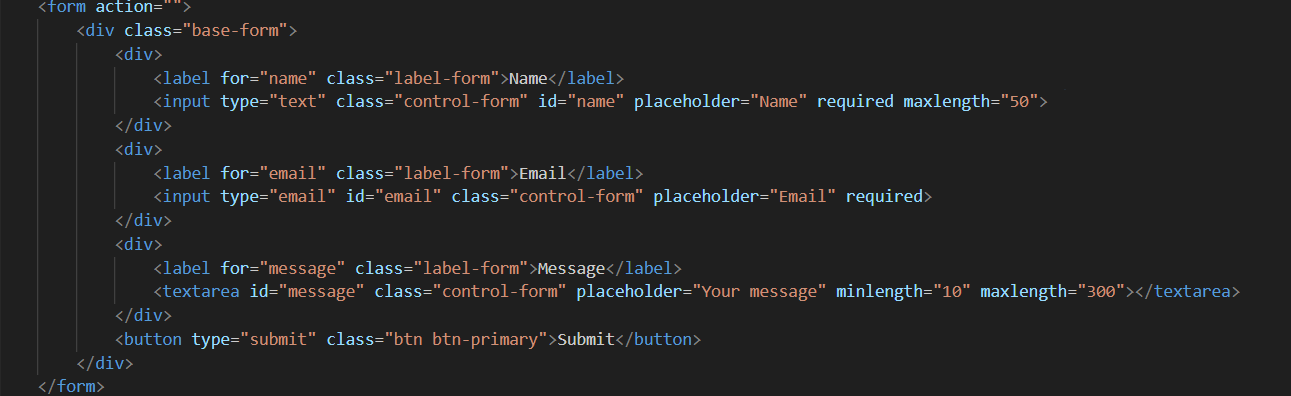
**Task 0. Simple Contact Form**

**Objective:**  
The goal of this task was to create a simple contact form that allows a user to enter their name, email, and a message, and then submit the form.

**Implementation Details:**

* The form contains three main input fields:
  + **Name** – implemented using a <input type="text"> element.
  + **Email** – implemented using a <input type="email"> element.
  + **Message** – implemented using a <textarea> element.
* Each input field is associated with a <label> element using the for attribute to improve accessibility and usability.
* A **Submit** button (<button type="submit">) is included to send the form data.
* The page also includes a simple navigation bar linking to other tasks.

**HTML Example:**

****

**Task 1. Form with HTML5 Validation**

**Objective:**  
Enhance the previous form by adding HTML5 validation attributes to ensure data accuracy and completeness before submission.

**Validation Features Implemented:**

* The required attribute was added to ensure that the user cannot submit the form with empty fields.
* The type="email" attribute automatically checks whether the entered email has a valid format.
* The maxlength and minlength attributes were used to control the length of input text.

**Code Example:**



**Part 2. Bootstrap Forms**

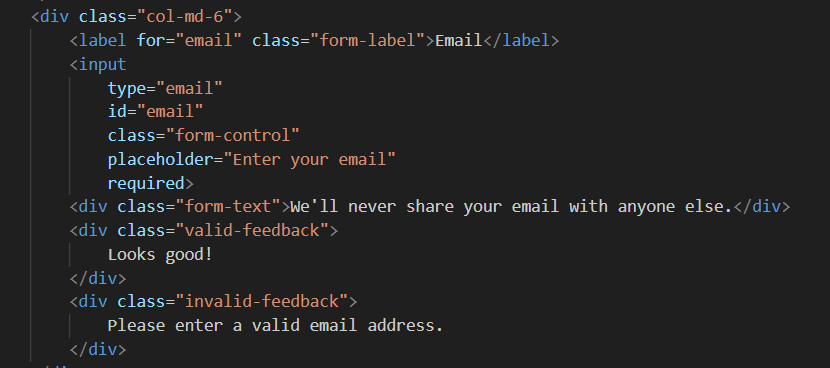
**Task 2. Bootstrap Form Styling**

**Objective:**  
The goal of this task was to recreate the basic contact form using **Bootstrap** classes to enhance its visual design, consistency, and responsiveness.

**Implementation Details:**

* The form was rebuilt using Bootstrap’s built-in classes such as .form-control, .form-label, .form-text, and .btn.
* The **Bootstrap grid system** (.row, .col-md-6, .col-12) was used to organize input fields neatly and responsively.
  + The **Name** and **Email** fields are placed side by side on medium and larger screens (col-md-6).
  + The **Message** field spans the full width (col-12).
* Each input field includes a **placeholder** for better user guidance.
* Helper text was added under the **Email** input using the .form-text class to provide additional context to the user.

**Code Example:**



**Task 3. Bootstrap Validation Styles**

**Objective:**  
The aim of this task was to enhance the user experience by integrating **Bootstrap’s built-in validation feedback system**.

**Implementation Details:**

* Validation feedback classes .is-valid and .is-invalid were applied dynamically using JavaScript.
* The form uses the Bootstrap validation workflow with the class .needs-validation and the novalidate attribute to disable the browser’s default validation messages.
* Custom feedback messages were added below each input field:
  + **Valid feedback:** “Looks good!”
  + **Invalid feedback:** specific error messages such as “Please enter your name.”
* JavaScript checks the form’s validity on submission. If the form is invalid, submission is prevented and the was-validated class is applied to trigger Bootstrap’s validation styles.

**JavaScript Snippet:**



**Part 3. Advanced Form Project**

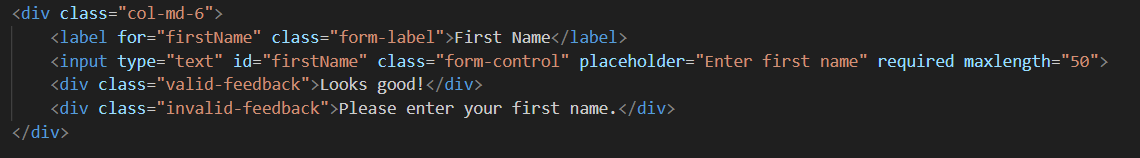
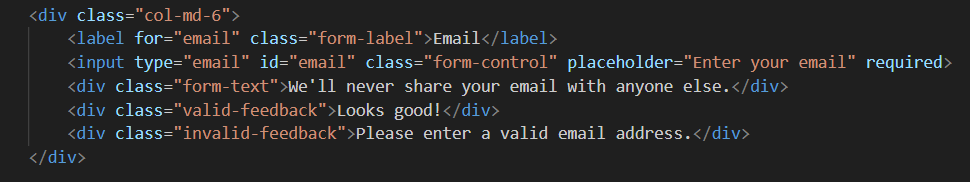
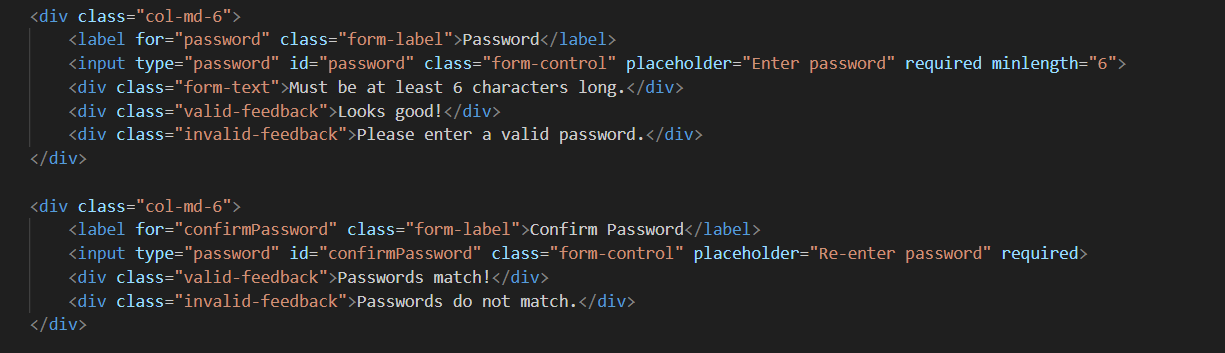
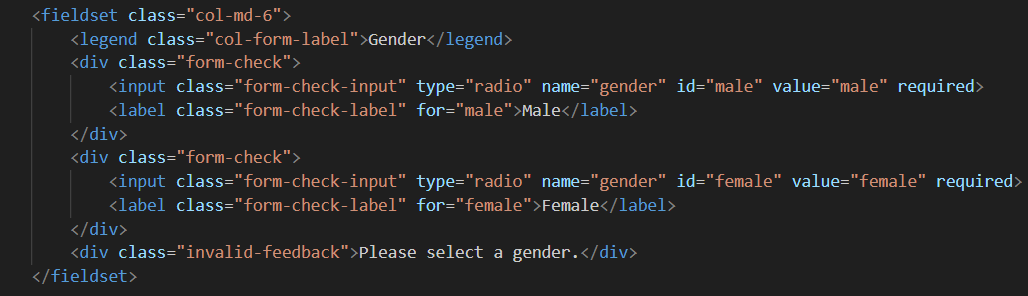
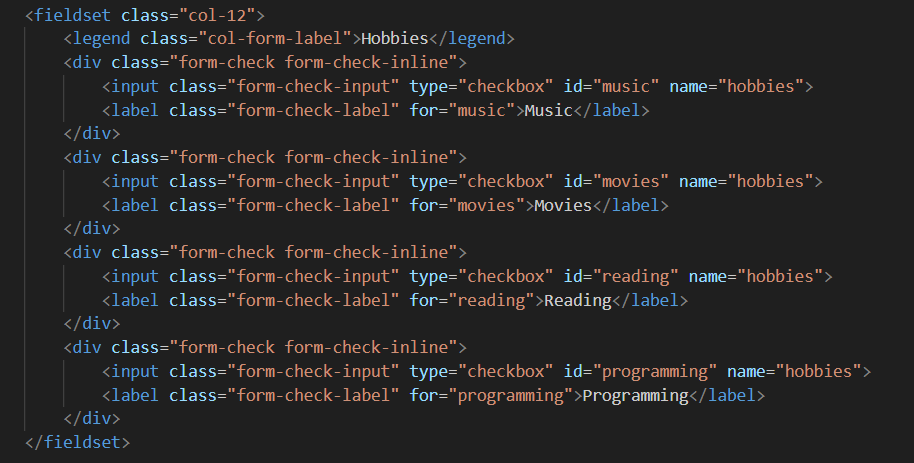
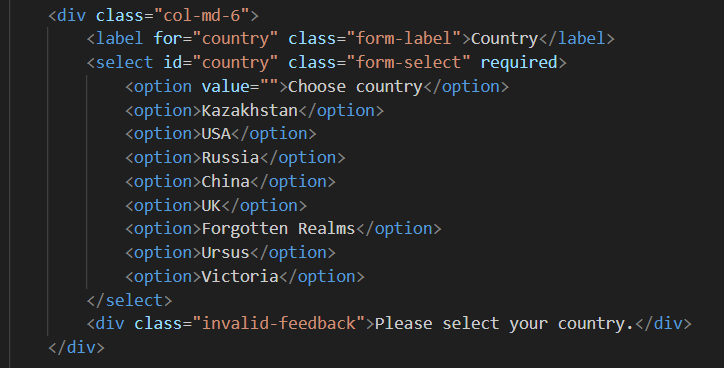
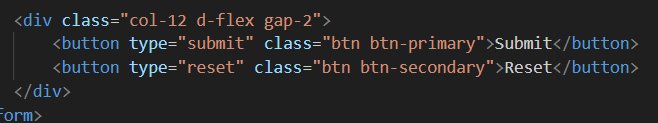
**Task 4. Registration Form**

**Objective:**  
The goal of this task was to create a **fully functional and responsive registration form** for a website using **Bootstrap 5**. The form collects essential user information (name, email, password, gender, hobbies, and country) and includes both **HTML5 validation** and **Bootstrap validation feedback** for a modern and user-friendly interface.

**Implementation Details**

**Form Structure and Fields**

The form includes the following input fields and elements:

1. **First Name** and **Last Name**
   * Implemented using <input type="text"> with required and maxlength="50".
   * Validation feedback messages indicate whether the input is valid or missing.
   * 
2. **Email**
   * Implemented with <input type="email"> to automatically validate email format.
   * Includes helper text using Bootstrap’s .form-text to reassure users about privacy.
   * 
3. **Password** and **Confirm Password**
   * Password input uses minlength="6" to enforce a minimum length requirement.
   * The “Confirm Password” field checks if it matches the first password using custom JavaScript validation.
   * Both fields include success (“Looks good!”) and error (“Passwords do not match”) messages.
   * 
4. **Gender** (Radio Buttons)
   * Implemented with two radio buttons inside a <fieldset> for accessibility.
   * Validation ensures one option is selected before submission.
   * 
5. **Hobbies** (Checkboxes)
   * Four checkboxes allow multiple selections: Music, Movies, Reading, and Programming.
   * Styled using .form-check and .form-check-inline classes for a clean horizontal layout.
   * 
6. **Country** (Dropdown Menu)
   * Implemented using <select> with several country options.
   * Includes validation feedback if no country is selected.
   * 
7. **Submit** and **Reset Buttons**
   * The Submit button validates the form and displays Bootstrap feedback.
   * The Reset button clears all input fields.
   * Buttons are styled with .btn, .btn-primary, and .btn-secondary.
   * 

**Styling and Layout**

* The form is built with **Bootstrap’s grid system** (.row, .col-md-6, .col-12) to ensure responsiveness.
* On larger screens, fields like **First Name**, **Last Name**, and **Password** appear side-by-side.
* On smaller screens, the layout stacks vertically for better readability and usability.
* Consistent spacing between elements is achieved using .g-3 for gutters and .d-flex.gap-2 for button alignment.

**Validation Implementation**

**HTML5 Validation**

* Attributes such as required, minlength, maxlength, and type enforce input constraints before submission.

**Bootstrap Validation Feedback**

* Bootstrap’s .is-valid and .is-invalid states are applied dynamically when the user interacts with the form.

**Custom JavaScript Validation**

A script ensures real-time validation for password matching and prevents form submission if any input fails validation.

