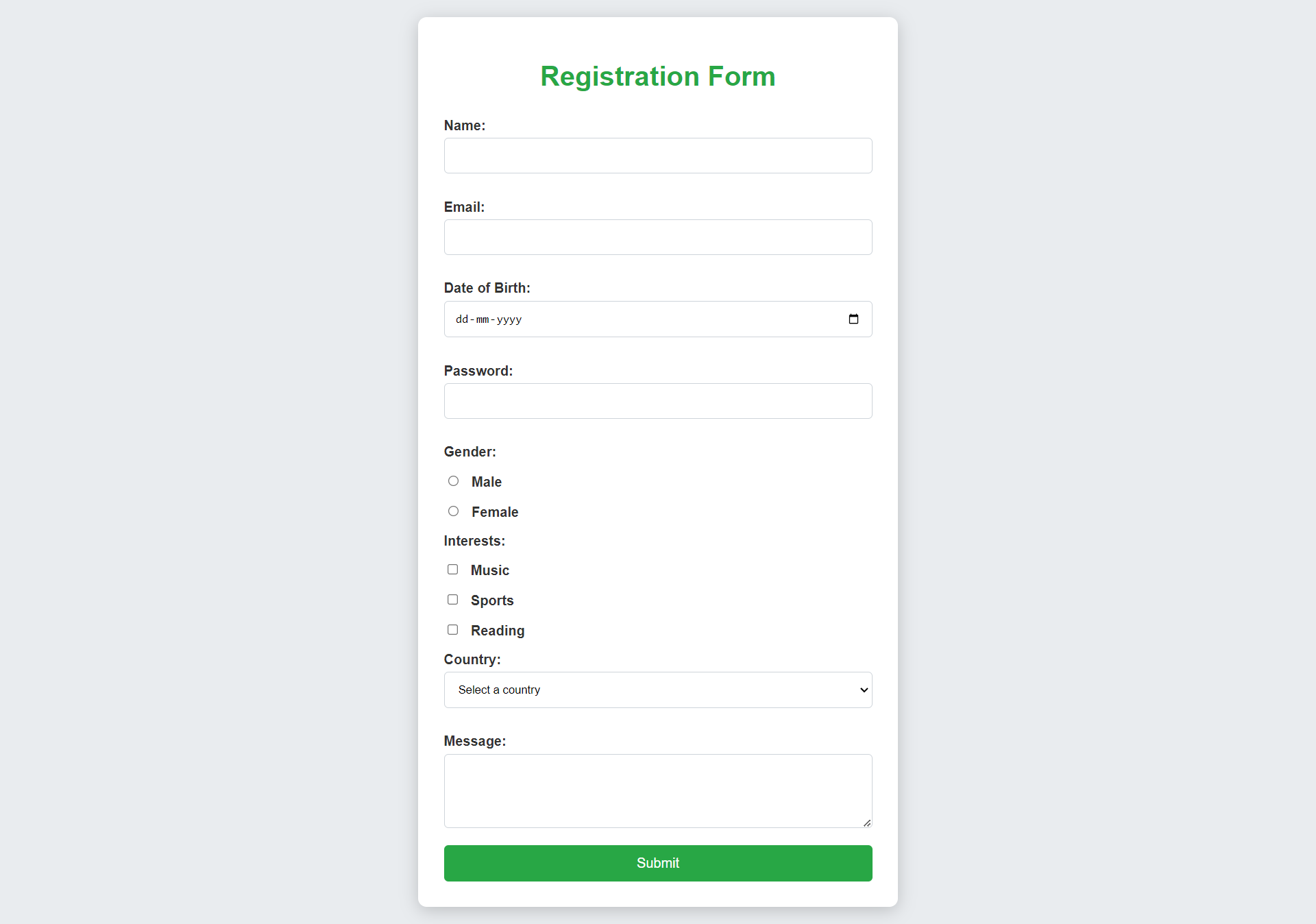
**Problem Statement:**

Create a form in HTML that includes various UI elements like text input, email input, radio buttons, checkboxes, a dropdown, and a submit button. Implement functionality to display the user's age based on the entered date of birth and handle form submission with JavaScript.

**Steps Taken to Solve the Problem:**

1. **Text Input Fields**:
   * Created input fields for **Name**, **Email**, **Date of Birth**, and **Password** using <input> elements with appropriate types (text, email, date, password) and added the required attribute to ensure users fill in the fields.
2. **Date of Birth Input with Age Calculation**:
   * Added a date picker for the **Date of Birth** field. A JavaScript function calculateAge() is triggered when the user changes the date. This function calculates the age based on the selected date of birth and displays it below the date field using a <span> element.
3. **Radio Buttons**:
   * Used <input type="radio"> for selecting **Gender**. Two radio buttons allow users to choose between "Male" and "Female." One of them must be selected since the required attribute is applied.
4. **Checkboxes**:
   * Added checkboxes for **Interests** with options like "Music," "Sports," and "Reading." Multiple checkboxes can be selected.
5. **Dropdown (Select Box)**:
   * A **Country** dropdown is created using the <select> element. It contains a list of countries like India, USA, Russia, etc. The required attribute ensures that users select an option.
6. **Textarea for Messages**:
   * A **Message** input field is provided using the <textarea> element where users can enter text, and the number of visible rows is set to 4.
7. **Submit Button**:
   * A **Submit** button is implemented using the <button> element. On form submission, JavaScript handles the event to prevent default submission, reset the form, and display a success message (alert("Form submitted!")).
8. **Form Validation and Styling**:
   * The form fields are styled with basic CSS, including padding, border-radius, and transitions for hover and focus effects. Focused input fields get a green border (border-color: #28a745), indicating active input.
   * The container is centered using Flexbox, and a shadow effect is applied to create a card-like UI.

**OUTPUT:**

****