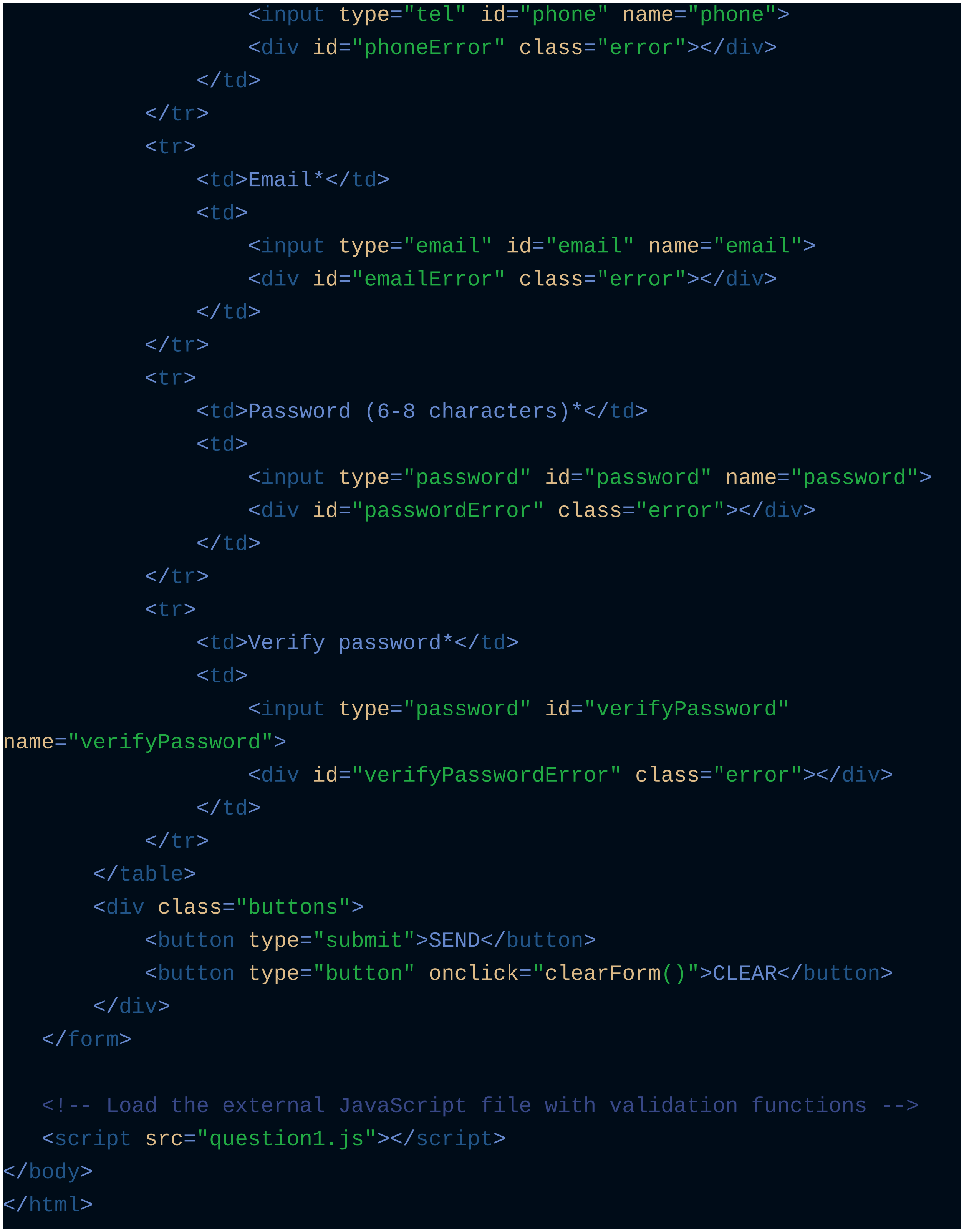
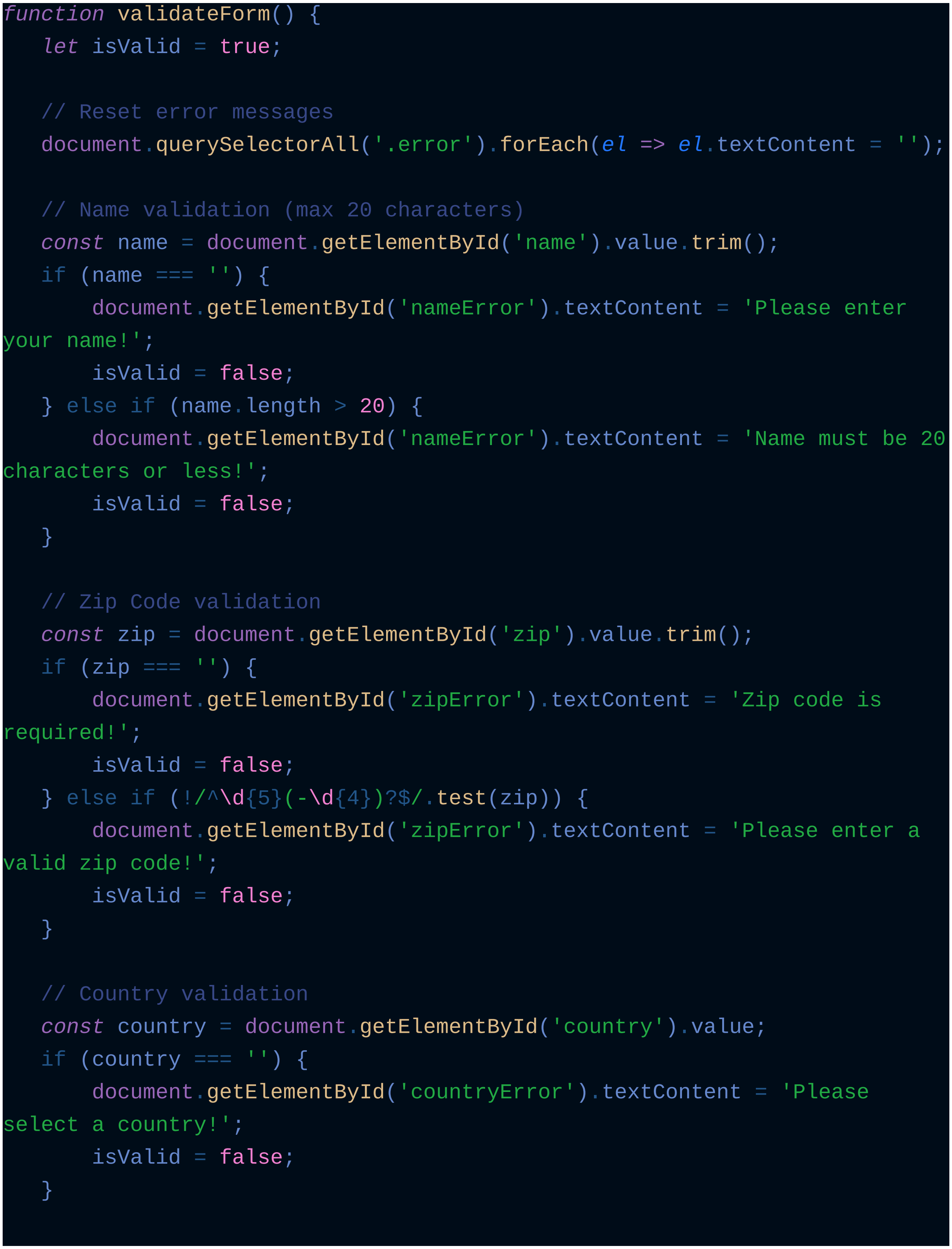
SEIF MWITA MGENI - BCS/18/23/024/TZ

```
QUESTION 1:

<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Form Validation</title>
   <style>
       body { font-family: Arial, sans-serif; margin: 20px; }
       .error { color: red; font-size: 0.8em; }
       textarea { width: 100%; }
       table { width: 100%; border-collapse: collapse; }
       td { padding: 8px; vertical-align: top; }
       .buttons { text-align: center; margin-top: 20px; }
       button { padding: 8px 16px; margin: 0 10px; }
   </style>
</head>
<body>
   <h1>Test JavaScript Form Validation</h1>
   <form id="myForm" onsubmit="return validateForm()">
       <table>
           <tr>
               <td>Name*</td>
               <td>
                   <input type="text" id="name" name="name">
                   <div id="nameError" class="error">Please enter your
name!</div>
               </td>
           </tr>
           <tr>
               <td>Address</td>
               <td><textarea id="address" name="address"
rows="3"></textarea></td>
           </tr>
           <tr>
               <td>Zip Code*</td>
               <td>
                   <input type="text" id="zip" name="zip">
                   <div id="zipError" class="error"></div>
```

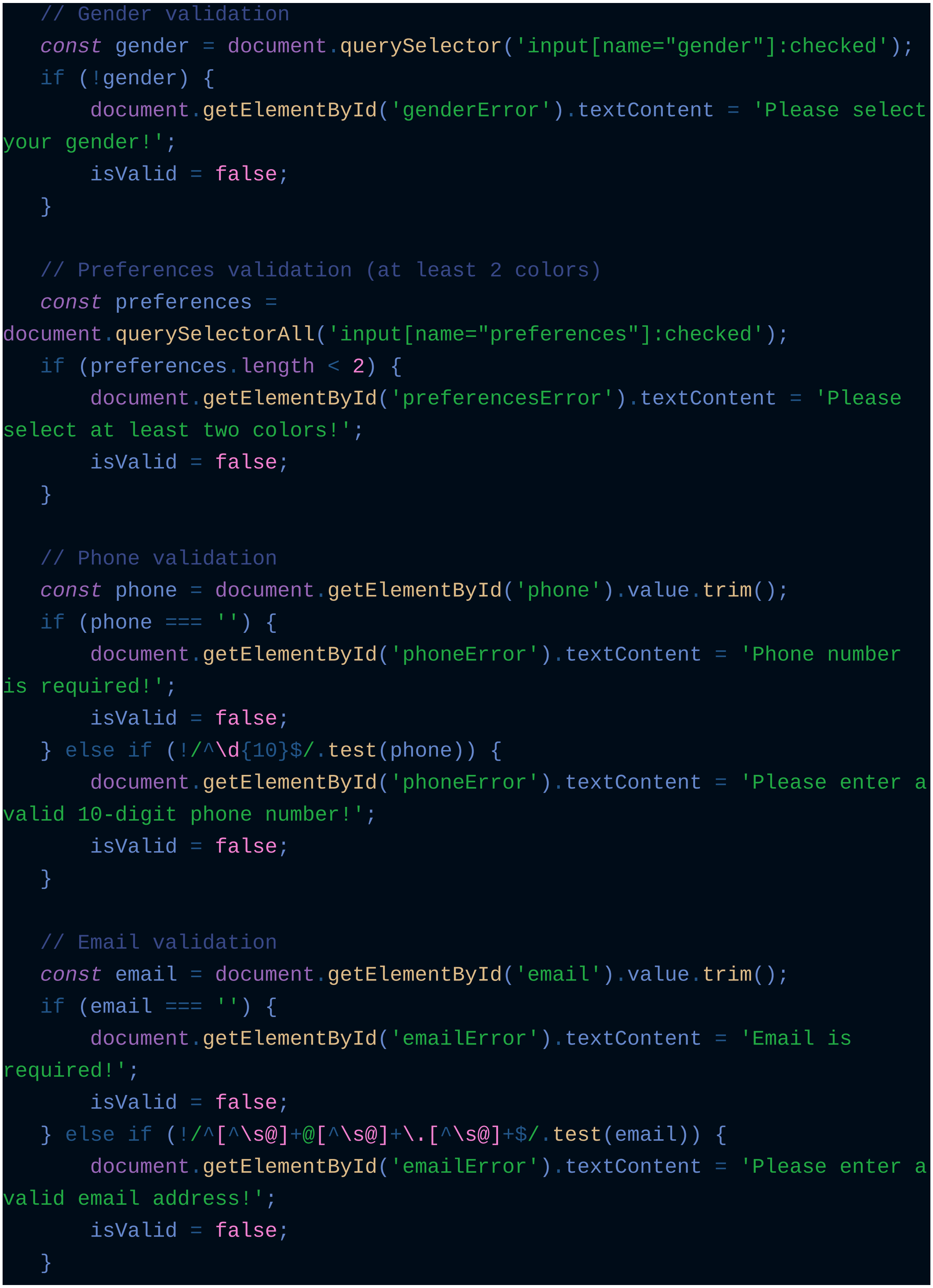
```
                </td>
            </tr>
            <tr>
                <td>Country*</td>
                <td>
                    <select id="country" name="country">
                        <option value="">Please select... ▼</option>
                        <option value="US">United States</option>
                        <option value="UK">United Kingdom</option>
                        <option value="CA">Canada</option>
                        <option value="AU">Australia</option>
                    </select>
                    <div id="countryError" class="error"></div>
                </td>
            </tr>
            <tr>
                <td>Gender*</td>
                <td>
                    <input type="radio" id="male" name="gender"
value="male"> Male
                    <input type="radio" id="female" name="gender"
value="female"> Female
                    <div id="genderError" class="error"></div>
                </td>
            </tr>
            <tr>
                <td>Preferences*</td>
                <td>
                    <input type="checkbox" id="red" name="preferences"
value="red"> Red
                    <input type="checkbox" id="green" name="preferences"
value="green"> Green
                    <input type="checkbox" id="blue" name="preferences"
value="blue"> Blue
                    <div id="preferencesError" class="error"></div>
                </td>
            </tr>
            <tr>
                <td>Phone*</td>
                <td>
```



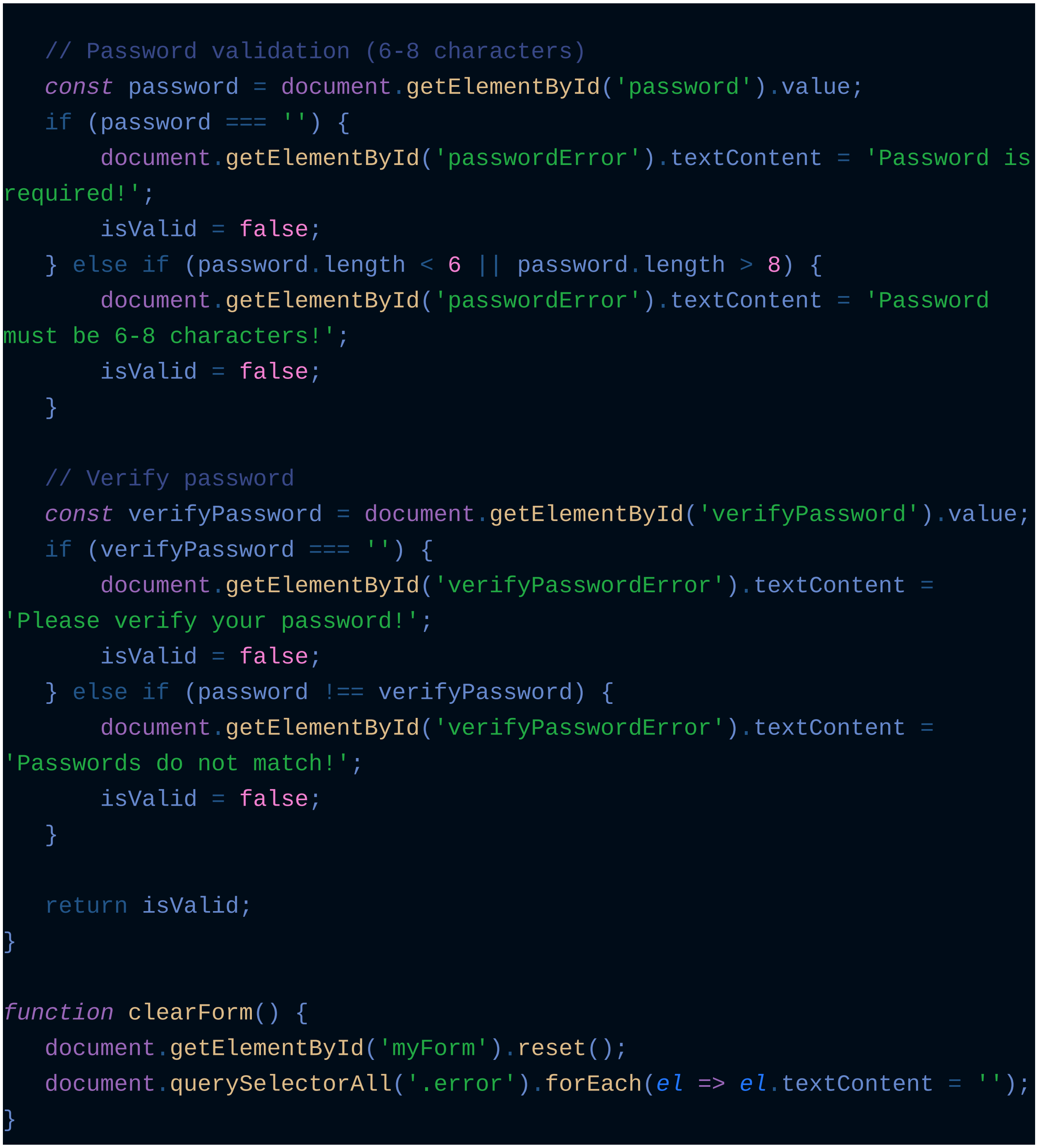
```
                   <input type="tel" id="phone" name="phone">
                   <div id="phoneError" class="error"></div>
               </td>
           </tr>
           <tr>
               <td>Email*</td>
               <td>
                   <input type="email" id="email" name="email">
                   <div id="emailError" class="error"></div>
               </td>
           </tr>
           <tr>
               <td>Password (6-8 characters)*</td>
               <td>
                   <input type="password" id="password" name="password">
                   <div id="passwordError" class="error"></div>
               </td>
           </tr>
           <tr>
               <td>Verify password*</td>
               <td>
                   <input type="password" id="verifyPassword"
name="verifyPassword">
                   <div id="verifyPasswordError" class="error"></div>
               </td>
           </tr>
       </table>
       <div class="buttons">
           <button type="submit">SEND</button>
           <button type="button" onclick="clearForm()">CLEAR</button>
       </div>
   </form>

   <!-- Load the external JavaScript file with validation functions -->
   <script src="question1.js"></script>
</body>
</html>
```



```
function validateForm() {
   let isValid = true;

   // Reset error messages
   document.querySelectorAll('.error').forEach(el => el.textContent = '');

   // Name validation (max 20 characters)
   const name = document.getElementById('name').value.trim();
   if (name === '') {
       document.getElementById('nameError').textContent = 'Please enter your name!';
       isValid = false;
   } else if (name.length > 20) {
       document.getElementById('nameError').textContent = 'Name must be 20 characters or less!';
       isValid = false;
   }

   // Zip Code validation
   const zip = document.getElementById('zip').value.trim();
   if (zip === '') {
       document.getElementById('zipError').textContent = 'Zip code is required!';
       isValid = false;
   } else if (!/^\d{5}(-\d{4})?$/.test(zip)) {
       document.getElementById('zipError').textContent = 'Please enter a valid zip code!';
       isValid = false;
   }

   // Country validation
   const country = document.getElementById('country').value;
   if (country === '') {
       document.getElementById('countryError').textContent = 'Please select a country!';
       isValid = false;
   }
```



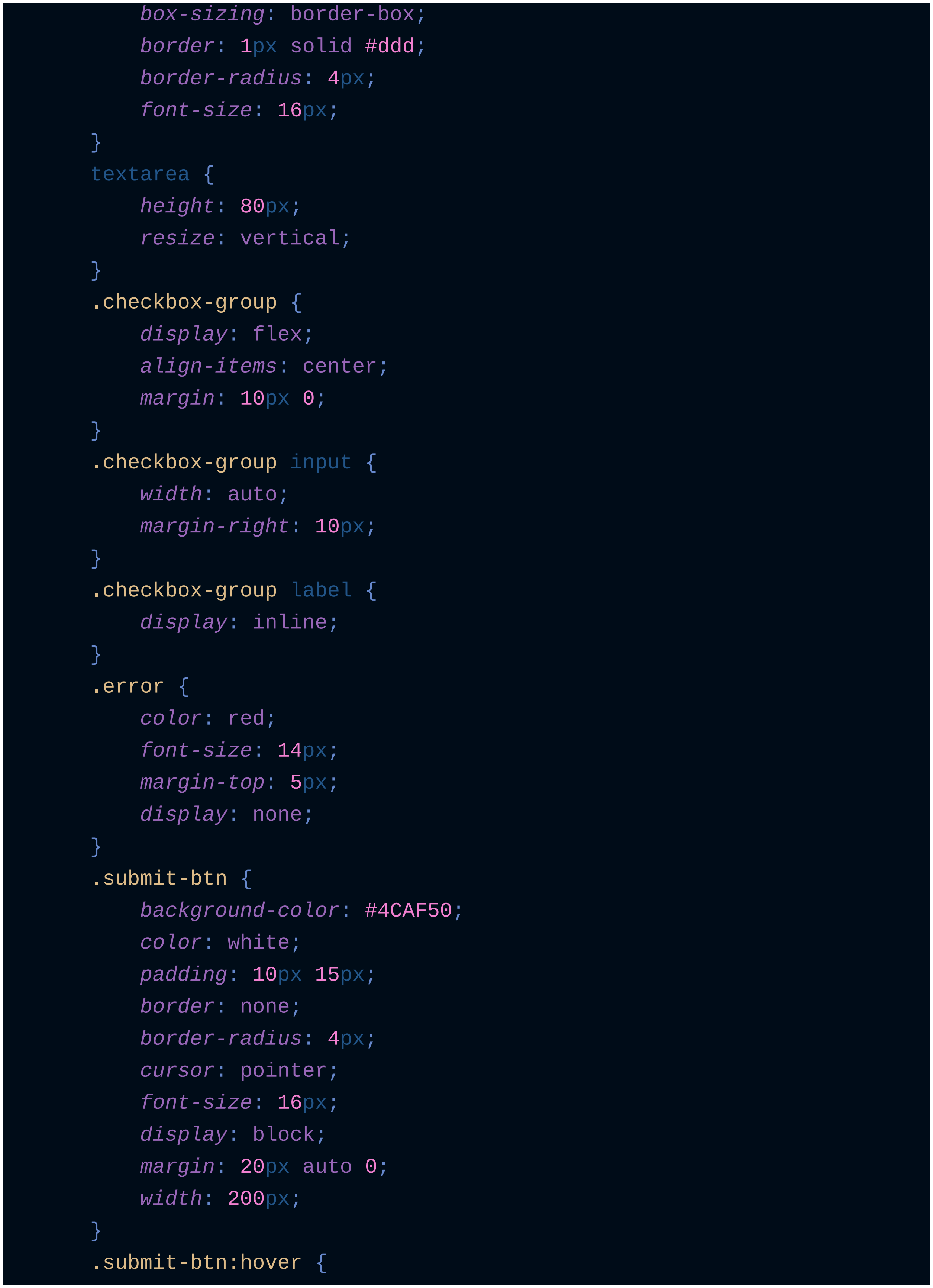
```
    // Gender validation
    const gender = document.querySelector('input[name="gender"]:checked');
    if (!gender) {
        document.getElementById('genderError').textContent = 'Please select your gender!';
        isValid = false;
    }

    // Preferences validation (at least 2 colors)
    const preferences = document.querySelectorAll('input[name="preferences"]:checked');
    if (preferences.length < 2) {
        document.getElementById('preferencesError').textContent = 'Please select at least two colors!';
        isValid = false;
    }

    // Phone validation
    const phone = document.getElementById('phone').value.trim();
    if (phone === '') {
        document.getElementById('phoneError').textContent = 'Phone number is required!';
        isValid = false;
    } else if (!/^\d{10}$/.test(phone)) {
        document.getElementById('phoneError').textContent = 'Please enter a valid 10-digit phone number!';
        isValid = false;
    }

    // Email validation
    const email = document.getElementById('email').value.trim();
    if (email === '') {
        document.getElementById('emailError').textContent = 'Email is required!';
        isValid = false;
    } else if (!/^[^\s@]+@[^\s@]+\.[^\s@]+$/.test(email)) {
        document.getElementById('emailError').textContent = 'Please enter a valid email address!';
        isValid = false;
    }
```



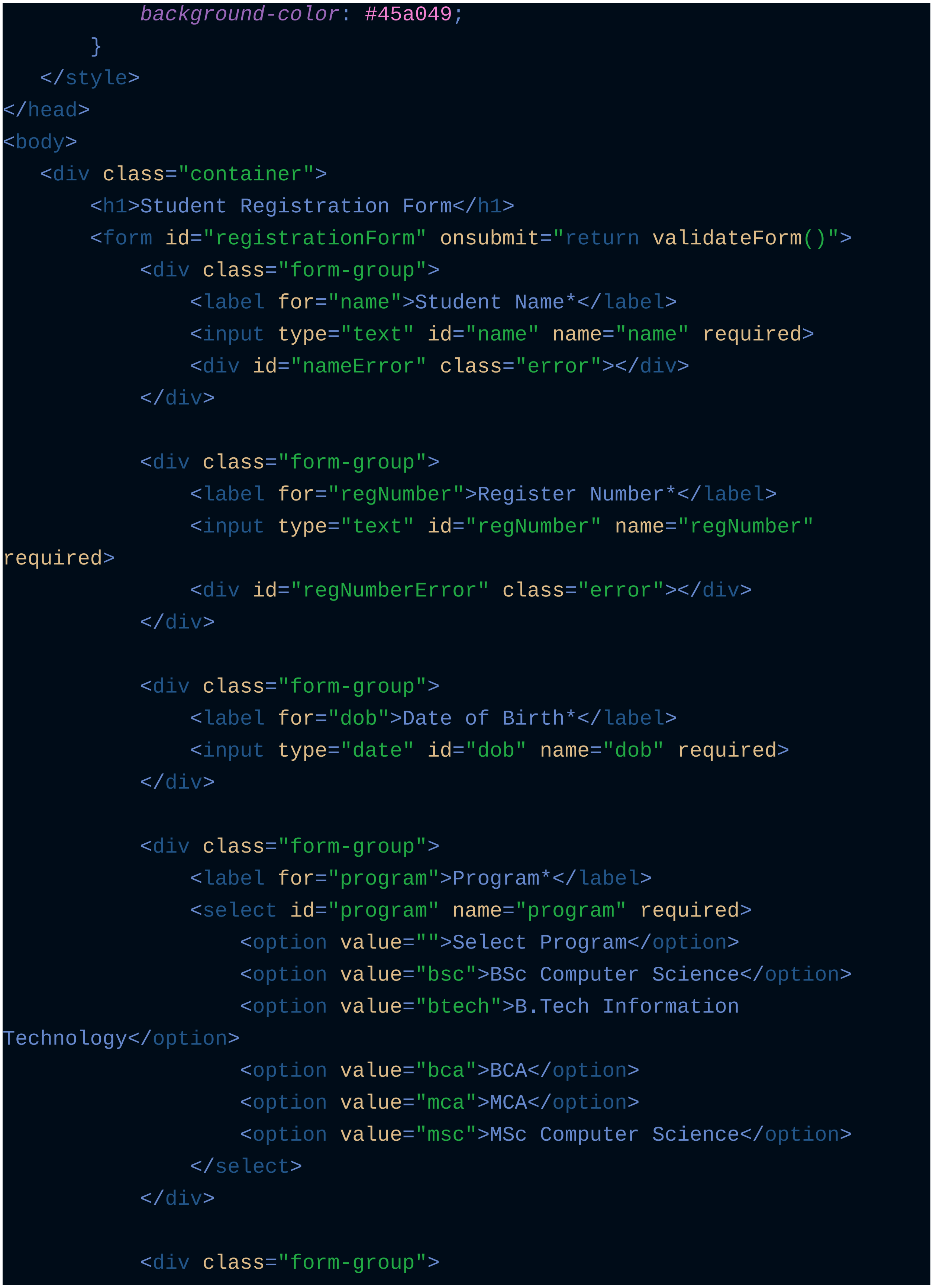
```
   // Password validation (6-8 characters)
   const password = document.getElementById('password').value;
   if (password === '') {
       document.getElementById('passwordError').textContent = 'Password is required!';
       isValid = false;
   } else if (password.length < 6 || password.length > 8) {
       document.getElementById('passwordError').textContent = 'Password must be 6-8 characters!';
       isValid = false;
   }

   // Verify password
   const verifyPassword = document.getElementById('verifyPassword').value;
   if (verifyPassword === '') {
       document.getElementById('verifyPasswordError').textContent = 'Please verify your password!';
       isValid = false;
   } else if (password !== verifyPassword) {
       document.getElementById('verifyPasswordError').textContent = 'Passwords do not match!';
       isValid = false;
   }

   return isValid;
}

function clearForm() {
   document.getElementById('myForm').reset();
   document.querySelectorAll('.error').forEach(el => el.textContent = '');
}
```

QUESTION 2:

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Student Registration Form</title>
   <style>
       body {
           font-family: Arial, sans-serif;
           margin: 0;
           padding: 20px;
           background-color: #f5f5f5;
       }
       .container {
           max-width: 700px;
           margin: 0 auto;
           background-color: white;
           padding: 20px;
           border-radius: 8px;
           box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
       }
       h1 {
           text-align: center;
           color: #333;
       }
       .form-group {
           margin-bottom: 15px;
       }
       label {
           display: block;
           margin-bottom: 5px;
           font-weight: bold;
       }
       input, select, textarea {
           width: 100%;
           padding: 8px;
```

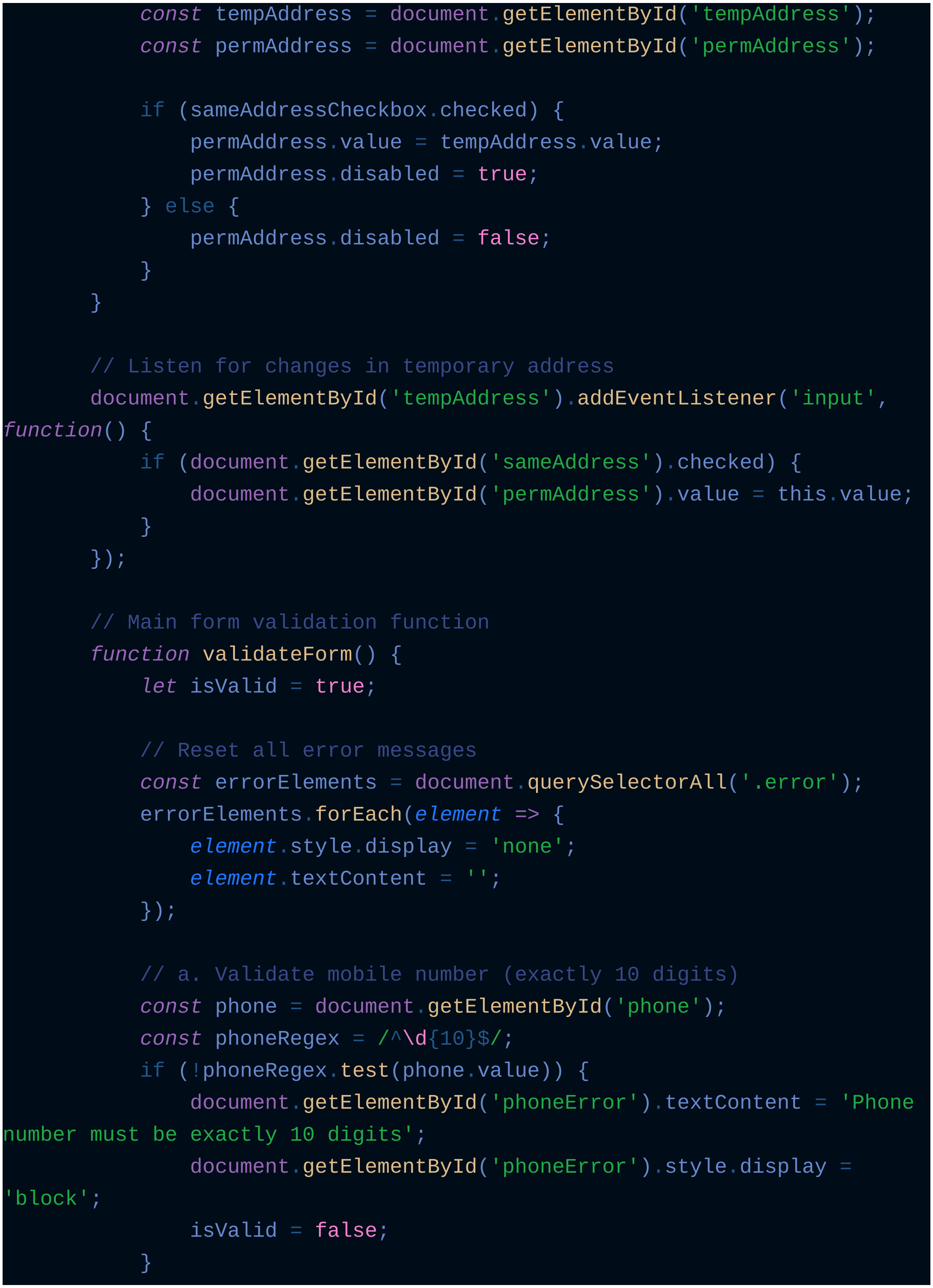


```
        box-sizing: border-box;
        border: 1px solid #ddd;
        border-radius: 4px;
        font-size: 16px;
    }
    textarea {
        height: 80px;
        resize: vertical;
    }
    .checkbox-group {
        display: flex;
        align-items: center;
        margin: 10px 0;
    }
    .checkbox-group input {
        width: auto;
        margin-right: 10px;
    }
    .checkbox-group label {
        display: inline;
    }
    .error {
        color: red;
        font-size: 14px;
        margin-top: 5px;
        display: none;
    }
    .submit-btn {
        background-color: #4CAF50;
        color: white;
        padding: 10px 15px;
        border: none;
        border-radius: 4px;
        cursor: pointer;
        font-size: 16px;
        display: block;
        margin: 20px auto 0;
        width: 200px;
    }
    .submit-btn:hover {
```

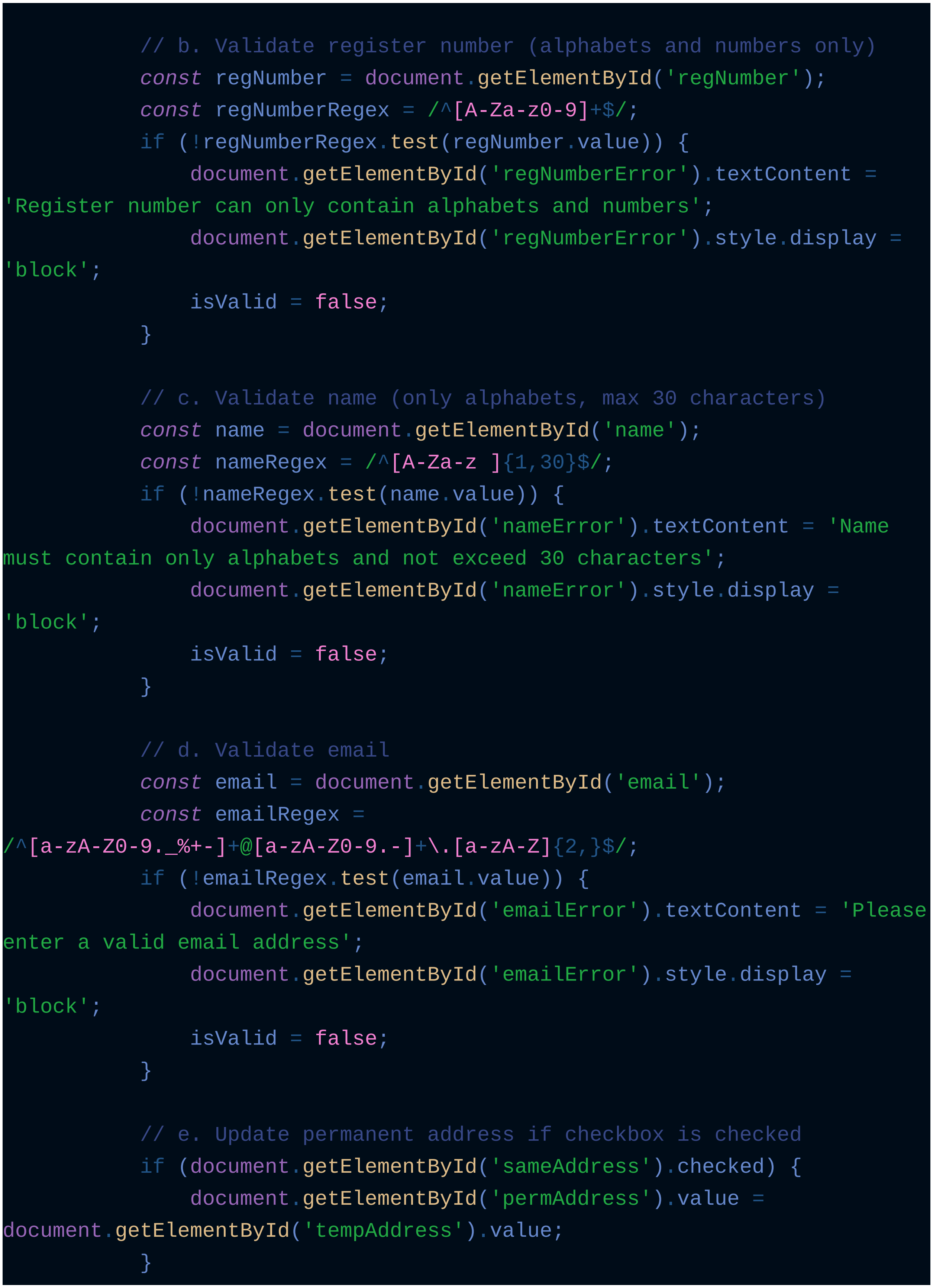


```
           background-color: #45a049;
       }
   </style>
</head>
<body>
   <div class="container">
       <h1>Student Registration Form</h1>
       <form id="registrationForm" onsubmit="return validateForm()">
           <div class="form-group">
               <label for="name">Student Name*</label>
               <input type="text" id="name" name="name" required>
               <div id="nameError" class="error"></div>
           </div>

           <div class="form-group">
               <label for="regNumber">Register Number*</label>
               <input type="text" id="regNumber" name="regNumber"
required>
               <div id="regNumberError" class="error"></div>
           </div>

           <div class="form-group">
               <label for="dob">Date of Birth*</label>
               <input type="date" id="dob" name="dob" required>
           </div>

           <div class="form-group">
               <label for="program">Program*</label>
               <select id="program" name="program" required>
                   <option value="">Select Program</option>
                   <option value="bsc">BSc Computer Science</option>
                   <option value="btech">B.Tech Information
Technology</option>
                   <option value="bca">BCA</option>
                   <option value="mca">MCA</option>
                   <option value="msc">MSc Computer Science</option>
               </select>
           </div>

           <div class="form-group">
```

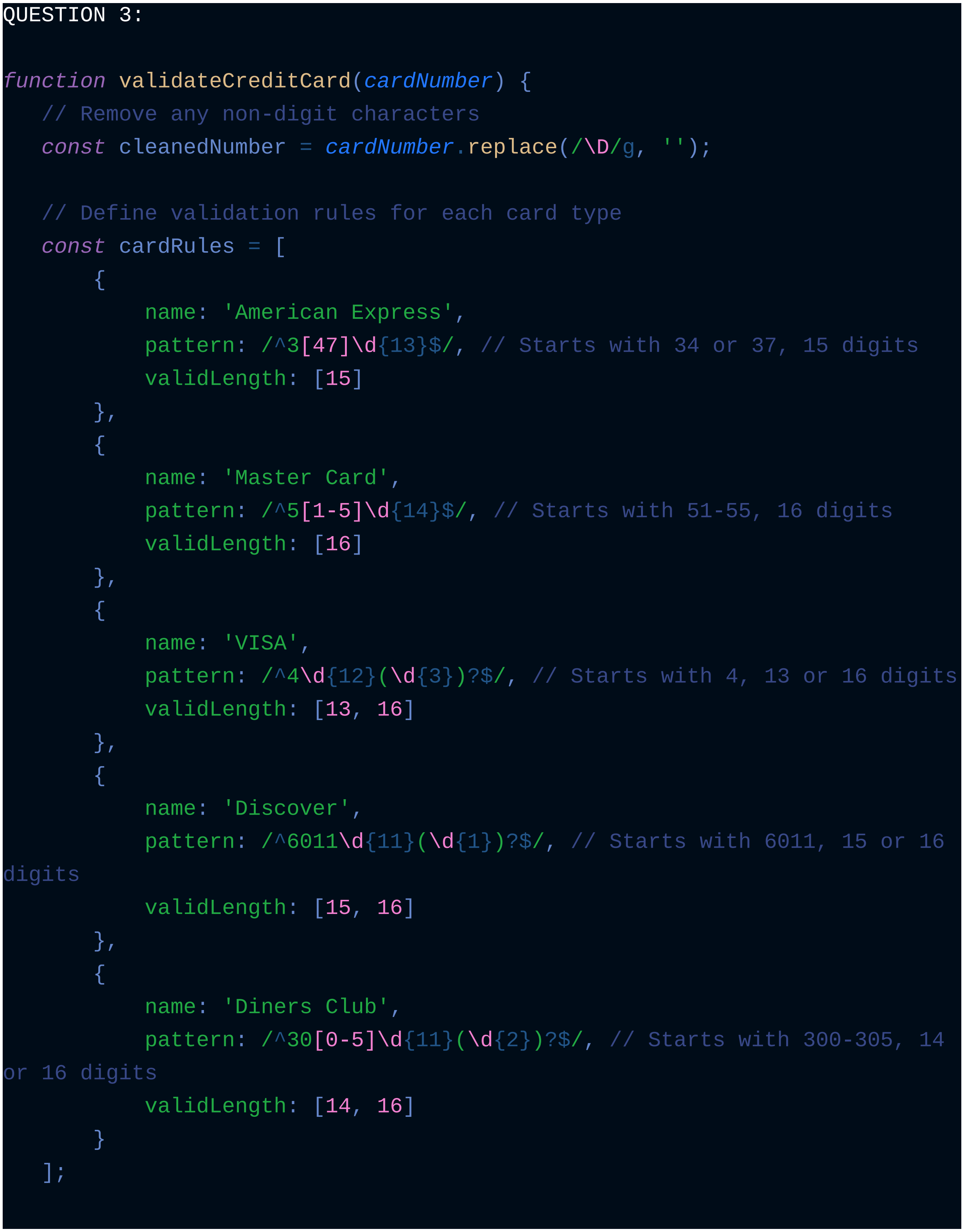
```
               <label for="email">Email ID*</label>
               <input type="text" id="email" name="email" required>
               <div id="emailError" class="error"></div>
           </div>

           <div class="form-group">
               <label for="tempAddress">Temporary Address*</label>
               <textarea id="tempAddress" name="tempAddress"
required></textarea>
           </div>

           <div class="checkbox-group">
               <input type="checkbox" id="sameAddress" name="sameAddress"
onchange="handleAddressCheckbox()">
               <label for="sameAddress">Permanent address is same as
temporary address</label>
           </div>

           <div class="form-group">
               <label for="permAddress">Permanent Address*</label>
               <textarea id="permAddress" name="permAddress"
required></textarea>
           </div>

           <div class="form-group">
               <label for="phone">Phone Number*</label>
               <input type="text" id="phone" name="phone" required
maxlength="10">
               <div id="phoneError" class="error"></div>
           </div>

           <button type="submit" class="submit-btn">Register</button>
       </form>
   </div>

   <script>
       // Function to handle the address checkbox
       function handleAddressCheckbox() {
           const sameAddressCheckbox =
document.getElementById('sameAddress');
```



```
            const tempAddress = document.getElementById('tempAddress');
            const permAddress = document.getElementById('permAddress');

            if (sameAddressCheckbox.checked) {
                permAddress.value = tempAddress.value;
                permAddress.disabled = true;
            } else {
                permAddress.disabled = false;
            }
        }

        // Listen for changes in temporary address
        document.getElementById('tempAddress').addEventListener('input',
function() {
            if (document.getElementById('sameAddress').checked) {
                document.getElementById('permAddress').value = this.value;
            }
        });

        // Main form validation function
        function validateForm() {
            let isValid = true;

            // Reset all error messages
            const errorElements = document.querySelectorAll('.error');
            errorElements.forEach(element => {
                element.style.display = 'none';
                element.textContent = '';
            });

            // a. Validate mobile number (exactly 10 digits)
            const phone = document.getElementById('phone');
            const phoneRegex = /^\d{10}$/;
            if (!phoneRegex.test(phone.value)) {
                document.getElementById('phoneError').textContent = 'Phone
number must be exactly 10 digits';
                document.getElementById('phoneError').style.display =
'block';
                isValid = false;
            }
```



```
           // b. Validate register number (alphabets and numbers only)
           const regNumber = document.getElementById('regNumber');
           const regNumberRegex = /^[A-Za-z0-9]+$/;
           if (!regNumberRegex.test(regNumber.value)) {
               document.getElementById('regNumberError').textContent =
'Register number can only contain alphabets and numbers';
               document.getElementById('regNumberError').style.display =
'block';
               isValid = false;
           }

           // c. Validate name (only alphabets, max 30 characters)
           const name = document.getElementById('name');
           const nameRegex = /^[A-Za-z ]{1,30}$/;
           if (!nameRegex.test(name.value)) {
               document.getElementById('nameError').textContent = 'Name
must contain only alphabets and not exceed 30 characters';
               document.getElementById('nameError').style.display =
'block';
               isValid = false;
           }

           // d. Validate email
           const email = document.getElementById('email');
           const emailRegex =
/^[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$/;
           if (!emailRegex.test(email.value)) {
               document.getElementById('emailError').textContent = 'Please
enter a valid email address';
               document.getElementById('emailError').style.display =
'block';
               isValid = false;
           }

           // e. Update permanent address if checkbox is checked
           if (document.getElementById('sameAddress').checked) {
               document.getElementById('permAddress').value =
document.getElementById('tempAddress').value;
           }
           if (isValid) {
               alert('Form submitted successfully!');
               // For testing purposes - normally this would submit to
server
           }

           return isValid;
       }
   </script>
</body>
</html>
```



```
QUESTION 3:

function validateCreditCard(cardNumber) {
   // Remove any non-digit characters
   const cleanedNumber = cardNumber.replace(/\D/g, '');

   // Define validation rules for each card type
   const cardRules = [
       {
           name: 'American Express',
           pattern: /^3[47]\d{13}$/, // Starts with 34 or 37, 15 digits
           validLength: [15]
       },
       {
           name: 'Master Card',
           pattern: /^5[1-5]\d{14}$/, // Starts with 51-55, 16 digits
           validLength: [16]
       },
       {
           name: 'VISA',
           pattern: /^4\d{12}(\d{3})?$/, // Starts with 4, 13 or 16 digits
           validLength: [13, 16]
       },
       {
           name: 'Discover',
           pattern: /^6011\d{11}(\d{1})?$/, // Starts with 6011, 15 or 16
digits
           validLength: [15, 16]
       },
       {
           name: 'Diners Club',
           pattern: /^30[0-5]\d{11}(\d{2})?$/, // Starts with 300-305, 14
or 16 digits
           validLength: [14, 16]
       }
   ];
```

```javascript
    // Check each card type
    for (const rule of cardRules) {
        if (rule.pattern.test(cleanedNumber) &&
            rule.validLength.includes(cleanedNumber.length)) {
            return { valid: true, type: rule.name };
        }
    }

    // If no match found
    return {
        valid: false,
        message: 'Invalid credit card number. Please check the number and try again.'
    };
}

// Example usage:
const result = validateCreditCard('4111111111111111'); // VISA
if (result.valid) {
    console.log(`Valid ${result.type} card`);
} else {
    console.log(result.message);
}

// Test cases
console.log(validateCreditCard('341234567890123'));    // American Express
console.log(validateCreditCard('5112345678901234'));   // Master Card
console.log(validateCreditCard('4123456789012'));      // VISA (13 digits)
console.log(validateCreditCard('4123456789012345'));   // VISA (16 digits)
console.log(validateCreditCard('601112345678901'));    // Discover (15 digits)
console.log(validateCreditCard('6011123456789012'));   // Discover (16 digits)
console.log(validateCreditCard('30512345678901'));     // Diners Club (14 digits)
console.log(validateCreditCard('3051234567890123'));   // Diners Club (16 digits)
console.log(validateCreditCard('1234'));               // Invalid
```