1.a)

```
function getDayOfWeek(dateString) {
  const daysOfWeek = ['Sunday', 'Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday',
'Saturday'];
  const date = new Date(dateString);
  const dayOfWeekIndex = date.getDay();
  return daysOfWeek[dayOfWeekIndex];
}
```

b)

```
function getCurrentDayOfWeek() {
  const daysOfWeek = ['Sunday', 'Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday',
'Saturday'];
  const date = new Date();
  const dayOfWeekIndex = date.getDay();
  return daysOfWeek[dayOfWeekIndex];
}
```

2. a)

```
function squareRootOfSumOfSquares(numbers) {
  const sumOfSquares = numbers.reduce((total, number) => total + Math.pow(number, 2), 0);
  return Math.sqrt(sumOfSquares);
}
```

b)

```
import math

def sqrt_sum_of_squares(nums):
    """Calculates the square root of the sum of squares for an array of numbers"""
    sum_of_squares = 0
    for num in nums:
        sum_of_squares += num ** 2
    return math.sqrt(sum_of_squares)
```

3.a)

```
function isPrime(number) {
  if (number <= 1) {
    return false;
  }

  // check for factors up to the square root of the number
  for (let i = 2; i <= Math.sqrt(number); i++) {
    if (number % i === 0) {
      return false;
    }
  }

  return true;
}
```

b)

```
def is_prime(number):
    """Checks if a given positive integer is a prime number"""
    if number <= 1:
        return False

    # check for factors up to the square root of the number
    for i in range(2, int(number**0.5)+1):
        if number % i == 0:
            return False

    return True
```

4.a)

```
function openWindow(url, width, height) {
  const left = (screen.width - width) / 2;
  const top = (screen.height - height) / 2;
  const options = `width=${width},height=${height},left=${left},top=${top}`;

  window.open(url, '', options);
}
```

b)

```
function openExampleWindow() {
  const url = 'https://www.example.com';
  const width = 800;
  const height = 600;

  const left = (screen.width - width) / 2;
  const top = (screen.height - height) / 2;
  const options = `width=${width},height=${height},left=${left},top=${top}`;

  window.open(url, '', options);
}
```

5.a)

```
function detectBrowser() {
  const userAgent = navigator.userAgent;
  let browserName, fullVersion;

  // Detect browser name
  if (userAgent.indexOf("Opera") > -1 || userAgent.indexOf("OPR") > -1) {
    browserName = "Opera";
  } else if (userAgent.indexOf("Edge") > -1) {
    browserName = "Microsoft Edge";
  } else if (userAgent.indexOf("Chrome") > -1) {
    browserName = "Google Chrome";
  } else if (userAgent.indexOf("Safari") > -1) {
```

```
    browserName = "Apple Safari";
  } else if (userAgent.indexOf("Firefox") > -1) {
    browserName = "Mozilla Firefox";
  } else if (userAgent.indexOf("MSIE") > -1 || userAgent.indexOf("Trident/") > -1) {
     browserName = "Microsoft Internet Explorer";
  } else {
    browserName = "unknown";
  }

  // Detect browser version
  if ((fullVersion =
userAgent.match(/(?:Edge|Chrome|CriOS|Firefox|Safari|OPR|MSIE|rv:)\s?([\d\.]+)/)) !== null) {
    fullVersion = fullVersion[1];
  } else {
    fullVersion = "unknown";
  }

  return `Browser: ${browserName}\nVersion: ${fullVersion}`;
}

console.log(detectBrowser());
```

b)

```
function displayBrowserInfo() {
  const userAgent = navigator.userAgent;
  let browserName, fullVersion;

  // Detect browser name
  if (userAgent.indexOf("Opera") > -1 || userAgent.indexOf("OPR") > -1) {
    browserName = "Opera";
  } else if (userAgent.indexOf("Edge") > -1) {
    browserName = "Microsoft Edge";
  } else if (userAgent.indexOf("Chrome") > -1) {
    browserName = "Google Chrome";
  } else if (userAgent.indexOf("Safari") > -1) {
    browserName = "Apple Safari";
  } else if (userAgent.indexOf("Firefox") > -1) {
    browserName = "Mozilla Firefox";
  } else if (userAgent.indexOf("MSIE") > -1 || userAgent.indexOf("Trident/") > -1) {
     browserName = "Microsoft Internet Explorer";
```

```
  } else {
    browserName = "unknown";
  }

  // Detect browser version
  if ((fullVersion =
userAgent.match(/(?:Edge|Chrome|CriOS|Firefox|Safari|OPR|MSIE|rv:)\s?([\d\.]+)/)) !== null) {
    fullVersion = fullVersion[1];
  } else {
    fullVersion = "unknown";
  }

  alert(`Your browser is ${browserName} version ${fullVersion}.`);
}

// Call the function to display the browser info
displayBrowserInfo();
```

6.a)

```
function getUserLocation() {
  if (navigator.geolocation) {
    navigator.geolocation.getCurrentPosition(showPosition);
  } else {
    alert("Geolocation is not supported by this browser.");
  }
}

function showPosition(position) {
  const latitude = position.coords.latitude;
  const longitude = position.coords.longitude;
  alert(`Your current location is: ${latitude}, ${longitude}`);
}
```

7.a)

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Click Coordinates</title>
  </head>
```

```
  <body onclick="showCoords(event);">
    <h1>Click Anywhere on the Page</h1>
    <script>
      function showCoords(event) {
        const x = event.clientX;
        const y = event.clientY;
        alert(`Clicked at coordinates (${x}, ${y})`);
      }
    </script>
  </body>
</html>
```

b)

```
window.addEventListener("keydown", function(event) {
  const keyCode = event.keyCode;
  alert(`You pressed the key with code ${keyCode}.`);
});
```

c)

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Change Image on Hover</title>
  </head>
  <body>
    <img src="original.jpg" alt="Original Image" id="myImg">
    <script>
      const img = document.getElementById("myImg");
      img.addEventListener("mouseover", function() {
        img.src = "hover.jpg";
      });
      img.addEventListener("mouseout", function() {
        img.src = "original.jpg";
      });
    </script>
  </body>
</html>
```

8.a)

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Button and Paragraph Element</title>
  </head>
  <body>
    <p id="myPara">This is a paragraph element.</p>
    <button id="myButton">Click me!</button>
    <script>
      const button = document.getElementById("myButton");
      const para = document.getElementById("myPara");
      button.addEventListener("click", function() {
        para.style.color = "red";
      });
    </script>
  </body>
</html>
```

b)

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Button and Paragraph Element</title>
  </head>
  <body>
    <p id="myPara">This is a paragraph element.</p>
    <button id="myButton">Click me!</button>
    <script>
      const button = document.getElementById("myButton");
      const para = document.getElementById("myPara");
      button.addEventListener("click", function() {
        para.textContent = "Button Clicked!";
      });
    </script>
  </body>
</html>
```

9.a)

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Form Validation</title>
  </head>
  <body>
    <form id="myForm">
      <div>
        <label for="name">Name:</label>
        <input type="text" id="name" name="name" required>
        <span class="error" id="nameError"></span>
      </div>
      <div>
        <label for="email">Email:</label>
        <input type="email" id="email" name="email" required>
        <span class="error" id="emailError"></span>
      </div>
      <div>
        <label for="password">Password:</label>
        <input type="password" id="password" name="password" required>
        <span class="error" id="passwordError"></span>
      </div>
      <div>
        <label for="confirmPassword">Confirm Password:</label>
        <input type="password" id="confirmPassword" name="confirmPassword" required>
        <span class="error" id="confirmPasswordError"></span>
      </div>
      <button type="submit" id="submitButton">Submit</button>
    </form>
    <script>
      const form = document.getElementById("myForm");
      const nameInput = document.getElementById("name");
      const emailInput = document.getElementById("email");
      const passwordInput = document.getElementById("password");
      const confirmPasswordInput = document.getElementById("confirmPassword");
      const nameError = document.getElementById("nameError");
      const emailError = document.getElementById("emailError");
      const passwordError = document.getElementById("passwordError");
      const confirmPasswordError = document.getElementById("confirmPasswordError");
      const submitButton = document.getElementById("submitButton");
```

```
  form.addEventListener("submit", function(event) {
    let errorCount = 0;

    // Validate name field
    const nameValue = nameInput.value.trim();
    if (nameValue === "") {
      nameError.textContent = "Name field is required";
      errorCount++;
    } else if (!/^[a-zA-Z]+$/.test(nameValue)) {
      nameError.textContent = "Name should contain only letters";
      errorCount++;
    } else {
      nameError.textContent = "";
    }

    // Validate email field
    const emailValue = emailInput.value.trim();
    if (emailValue === "") {
      emailError.textContent = "Email field is required";
      errorCount++;
    } else if (!/\S+@\S+\.\S+/.test(emailValue)) {
      emailError.textContent = "Email address is invalid";
      errorCount++;
    } else {
      emailError.textContent = "";
    }

    // Validate password field
    const passwordValue = passwordInput.value.trim();
    if (passwordValue === "") {
      passwordError.textContent = "Password field is required";
      errorCount++;
    } else if (passwordValue.length < 8 || !/[A-Z]/.test(passwordValue) ||
!/[a-z]/.test(passwordValue) || !/\d/.test(passwordValue)) {
      passwordError.textContent = "Password should have a minimum length of 8 characters
and contain at least one uppercase letter, one lowercase letter, and one digit";
      errorCount++;
    } else {
      passwordError.textContent = "";
    }

    // Validate confirm password field
    const confirmPasswordValue = confirmPasswordInput.value.trim();
    if (confirmPasswordValue === "") {
```

```
        confirmPasswordError.textContent = "Confirm Password field is required";
        errorCount++;
      } else if (confirmPasswordValue !== passwordValue) {
        confirmPasswordError.textContent = "Confirm Password should match the Password
field";
        errorCount++;
      } else {
        confirmPasswordError.textContent = "";
      }

      // Prevent form submission if there are errors
      if (errorCount > 0) {
        event.preventDefault();
      }
    });
  </script>
 </body>
</html>
```

10.a)

```
<!DOCTYPE html>
<html>
<head>
        <title>Button and Paragraph Element</title>
</head>
<body>

        <p id="text">This is a paragraph element.</p>
        <button onclick="changeText()">Click me!</button>

        <script>
                function changeText() {
                        document.getElementById("text").innerHTML = "The text has been
changed!";
                }
        </script>

</body>
</html>
```

b)

```
<!DOCTYPE html>
<html>
<head>
	<title>Button and Paragraph Element</title>

	<style>
		.highlight {
			background-color: yellow;
		}
	</style>
</head>
<body>

	<p id="text">This is a paragraph element.</p>
	<button onclick="changeText()">Click me!</button>

	<script>
		function changeText() {
			var text = document.getElementById("text");
			text.classList.add("highlight");
		}
	</script>

</body>
</html>
```

11.a)

```
<!DOCTYPE html>
<html>
  <head>
    <title>Empty Unordered List</title>
  </head>
  <body>
    <ul>
      <!-- the list is empty for now -->
    </ul>
  </body>
</html>
```

b)

```
<!DOCTYPE html>
<html>
  <head>
    <title>Dynamic List Creation</title>
  </head>
  <body>
    <ul id="list"></ul>

    <script>
      function createListItems() {
        // Get a reference to the unordered list element
        var ul = document.getElementById("list");

        // Loop through and create five list items with sequential numbers (1 to 5)
        for (var i = 1; i <= 5; i++) {
          // Create a new list item element
          var li = document.createElement("li");

          // Set the text content of the list item to the current number in the loop
          li.textContent = i;

          // Append the list item to the unordered list
          ul.appendChild(li);
        }
      }

      // Call the function to create the list items when the page loads
      window.onload = createListItems;
    </script>
  </body>
</html>
```

12.a)

```
<!DOCTYPE html>
<html>
  <head>
    <title>Image Element Example</title>
  </head>
```

```
  <body>
    <img src="https://example.com/image.jpg" alt="Example Image">
  </body>
</html>
```

b)

```
<!DOCTYPE html>
<html>
  <head>
    <title>Image Source Change Example</title>
  </head>
  <body>
    <img id="myImage" src="https://example.com/image1.jpg" alt="Example Image">
    <br>
    <button onclick="changeImage()">Change Image</button>

    <script>
      function changeImage() {
        // Get a reference to the image element
        var img = document.getElementById("myImage");

        // Set the new source URL for the image
        img.src = "https://example.com/image2.jpg";
      }
    </script>
  </body>
</html>
```

13.a)

```
<!DOCTYPE html>
<html>
  <head>
    <title>Paragraph Example</title>
  </head>
  <body>
    <p id="first-paragraph" class="intro">This is the first paragraph.</p>

    <p id="second-paragraph" class="content">This is the second paragraph.</p>
```

```
    <p id="third-paragraph" class="content">This is the third paragraph.</p>

    <p id="fourth-paragraph" class="conclusion">This is the fourth paragraph.</p>
  </body>
</html>
```

b)

```
<!DOCTYPE html>
<html>
  <head>
    <title>Paragraph Color Change Example</title>
    <style>
      .content {
        color: black;
      }
    </style>
  </head>
  <body>
    <p id="first-paragraph" class="intro">This is the first paragraph.</p>

    <p id="second-paragraph" class="content">This is the second paragraph.</p>

    <p id="third-paragraph" class="content">This is the third paragraph.</p>

    <p id="fourth-paragraph" class="conclusion">This is the fourth paragraph.</p>

    <button onclick="changeColor()">Change Color</button>

    <script>
      function changeColor() {
        // Get a reference to all paragraphs with class "content"
        var paragraphs = document.querySelectorAll(".content");

        // Loop through each paragraph and change its text color to red
        for (var i = 0; i < paragraphs.length; i++) {
          paragraphs[i].style.color = "red";
        }
      }
    </script>
  </body>
</html>
```

14.a)

```
<!DOCTYPE html>
<html>
  <head>
    <title>AJAX GET Request Example</title>
  </head>
  <body>
    <button onclick="getData()">Get Data</button>

    <script>
      function getData() {
        // Create a new XMLHttpRequest object
        var xhr = new XMLHttpRequest();

        // Set up a function to handle the response when it arrives
        xhr.onload = function() {
          if (xhr.status === 200) {
            console.log(xhr.responseText);
          }
        };

        // Open a new GET request with the specified URL
        xhr.open('GET', 'https://api.example.com/data');

        // Send the request
        xhr.send();
      }
    </script>
  </body>
</html>
```

15.a)

```
<!DOCTYPE html>
<html>
  <head>
    <title>AJAX POST Request Example</title>
  </head>
  <body>
    <form id="myForm">
      <input type="text" name="name">
```

```
    <input type="email" name="email">
    <button type="submit">Submit</button>
  </form>

  <script>
    function postData(event) {
      // Prevent the default form submission behavior
      event.preventDefault();

      // Create a new XMLHttpRequest object
      var xhr = new XMLHttpRequest();

      // Set up a function to handle the response when it arrives
      xhr.onload = function() {
        if (xhr.status === 200) {
          console.log(xhr.responseText);
        }
      };

      // Open a new POST request with the specified URL
      xhr.open('POST', 'https://api.example.com/submit');

      // Set the Content-Type header to indicate JSON data
      xhr.setRequestHeader('Content-Type', 'application/json');

      // Get the form data and convert it to a JSON string
      var formData = new FormData(document.getElementById('myForm'));
      var jsonData = {};

      for (var [key, value] of formData.entries()) {
        jsonData[key] = value;
      }

      var jsonString = JSON.stringify(jsonData);

      // Send the request with the JSON data as the request body
      xhr.send(jsonString);
    }

    // Attach the postData function to the form submit event
    document.getElementById('myForm').addEventListener('submit', postData);
  </script>
 </body>
</html>
```

b)

```
<!DOCTYPE html>
<html>
  <head>
    <title>AJAX POST Request Example</title>
  </head>
  <body>
    <button onclick="postData()">Send Data</button>

    <script>
      function postData() {
        // Create a new XMLHttpRequest object
        var xhr = new XMLHttpRequest();

        // Set up a function to handle the response when it arrives
        xhr.onload = function() {
          if (xhr.status === 200) {
            console.log(xhr.responseText);
          }
        };

        // Open a new POST request with the specified URL
        xhr.open('POST', 'https://api.example.com/submit');

        // Set the Content-Type header to indicate JSON data
        xhr.setRequestHeader('Content-Type', 'application/json');

        // Define the JSON data to send as the request body
        var jsonData = {
          "name": "John Doe",
          "email": "johndoe@example.com"
        };

        // Convert the JavaScript object to a JSON string
        var jsonString = JSON.stringify(jsonData);

        // Send the request with the JSON data as the request body
        xhr.send(jsonString);
      }
    </script>
  </body>
</html>
```