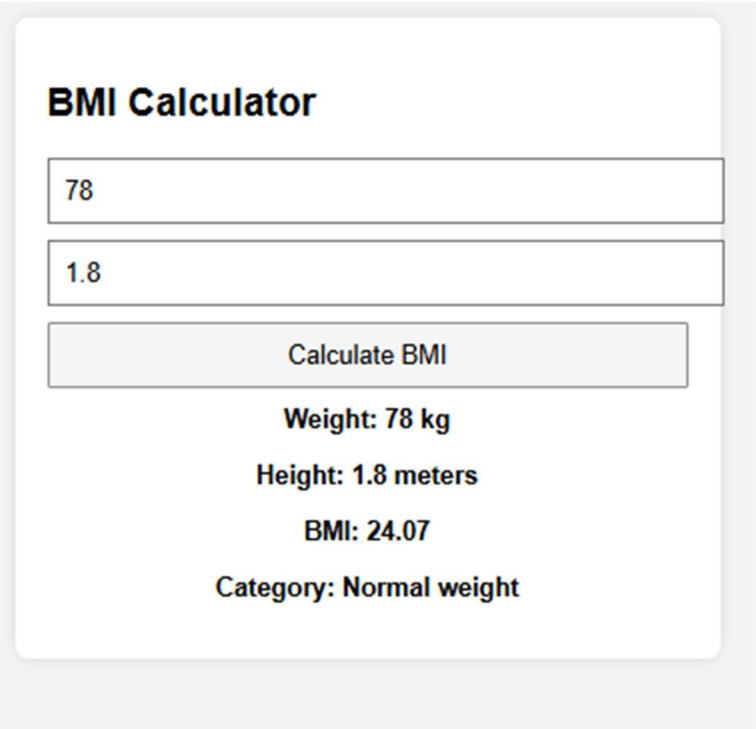
CODE: <!DOCTYPE html>

```
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>BMI Calculator</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            padding: 20px;
            background-color: #f4f4f4;
        }
        .container {
            max-width: 400px;
            margin: auto;
            background: #fff;
            padding: 20px;
            border-radius: 8px;
            box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
        }
        input, button {
            margin: 10px 0;
            display: block;
            width: 100%;
            padding: 10px;
            font-size: 16px;
        }
        .result {
            font-weight: bold;
            margin-top: 10px;
            text-align: center;
        }
```

```
    </style>
</head>
<body>
    <div class="container">
        <h2>BMI Calculator</h2>
        <input type="number" id="weight" placeholder="Weight in kilograms" required>
        <input type="number" id="height" placeholder="Height in meters" step="0.01" required>
        <button onclick="calculateBMI()">Calculate BMI</button>
        <p class="result" id="weightResult"></p>
        <p class="result" id="heightResult"></p>
        <p class="result" id="bmiResult"></p>
        <p class="result" id="categoryResult"></p>
    </div>
    <script>
        function calculateBMI() {
            var weight = parseFloat(document.getElementById('weight').value);
            var height = parseFloat(document.getElementById('height').value);

            if (isNaN(weight) || isNaN(height) || weight <= 0 || height <= 0) {
                alert('Please enter valid positive numbers for weight and height.');
                return;
            }

            var bmi = weight / (height * height);
            var category = '';

            if (bmi < 18.5) {
                category = 'Underweight';
            } else if (bmi >= 18.5 && bmi < 24.9) {
                category = 'Normal weight';
            } else if (bmi >= 25 && bmi < 29.9) {
                category = 'Overweight';
```

```
        } else {
            category = 'Obesity';
        }

        document.getElementById('weightResult').innerText = 'Weight: ' + weight + ' kg';
        document.getElementById('heightResult').innerText = 'Height: ' + height + ' meters';
        document.getElementById('bmiResult').innerText = 'BMI: ' + bmi.toFixed(2);
        document.getElementById('categoryResult').innerText = 'Category: ' + category;
    }
  </script>
</body>
</html>
```

OUTPUT:



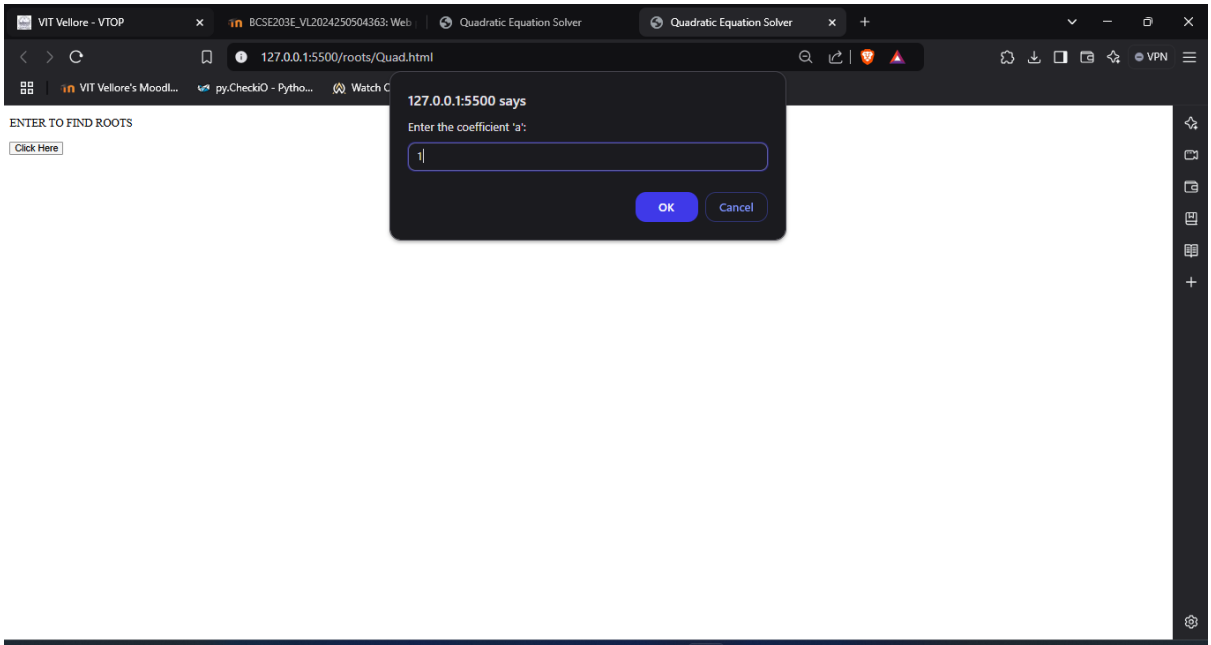
BCSE203E - Web Programming

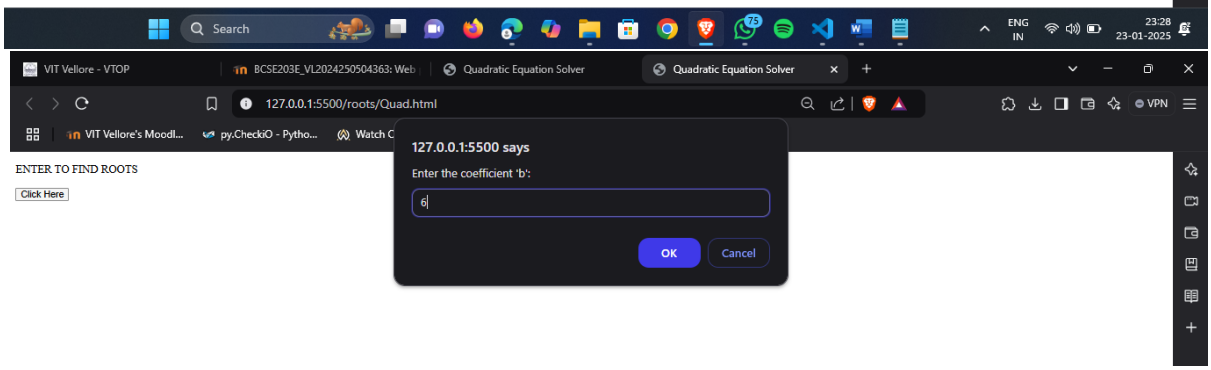
Code:

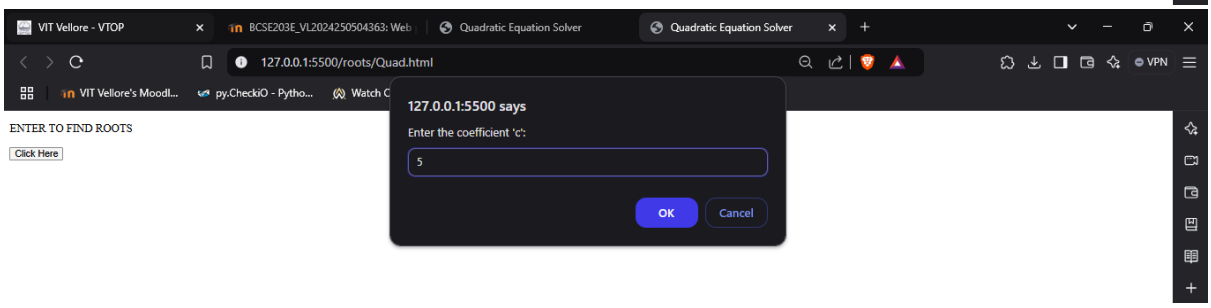
```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Quadratic Equation Solver</title>
</head>
<body>
   <p>ENTER TO FIND ROOTS</p>
   <button onclick="findroots()">Click Here</button>

   <script>
      function findroots() {
         var a = parseFloat(prompt("Enter the coefficient 'a':"));
         var b = parseFloat(prompt("Enter the coefficient 'b':"));
         var c = parseFloat(prompt("Enter the coefficient 'c':"));

         if (a === 0) {
            document.write("This is not a quadratic equation (a cannot be 0).");
            return;
         }

         var discriminant = b * b - 4 * a * c;
         document.write("Discriminant (sr): " + discriminant + "<br>");

         if (discriminant > 0) {
            var root1 = (-b + Math.sqrt(discriminant)) / (2 * a);
            var root2 = (-b - Math.sqrt(discriminant)) / (2 * a);
            document.write("<h1>Roots are real and distinct:</h1>");
            document.write("Root 1: " + root1 + "<br>");
```

```
            document.write("Root 2: " + root2 + "<br>");
        } else if (discriminant === 0) {
            var root = -b / (2 * a);
            document.write("<h1>Roots are real and equal:</h1>");
            document.write("Root: " + root + "<br>");
        } else {
            var realPart = -b / (2 * a);
            var imaginaryPart = Math.sqrt(-discriminant) / (2 * a);
            document.write("<h1>Roots are complex:</h1>");
            document.write("Root 1: " + realPart + " + " + imaginaryPart + "i<br>");
            document.write("Root 2: " + realPart + " - " + imaginaryPart + "i<br>");
        }
    }
  </script>
</body>
</html>
```

Output:







Discriminant (sr): 16

Roots are real and distinct:

Root 1: -1
Root 2: -5

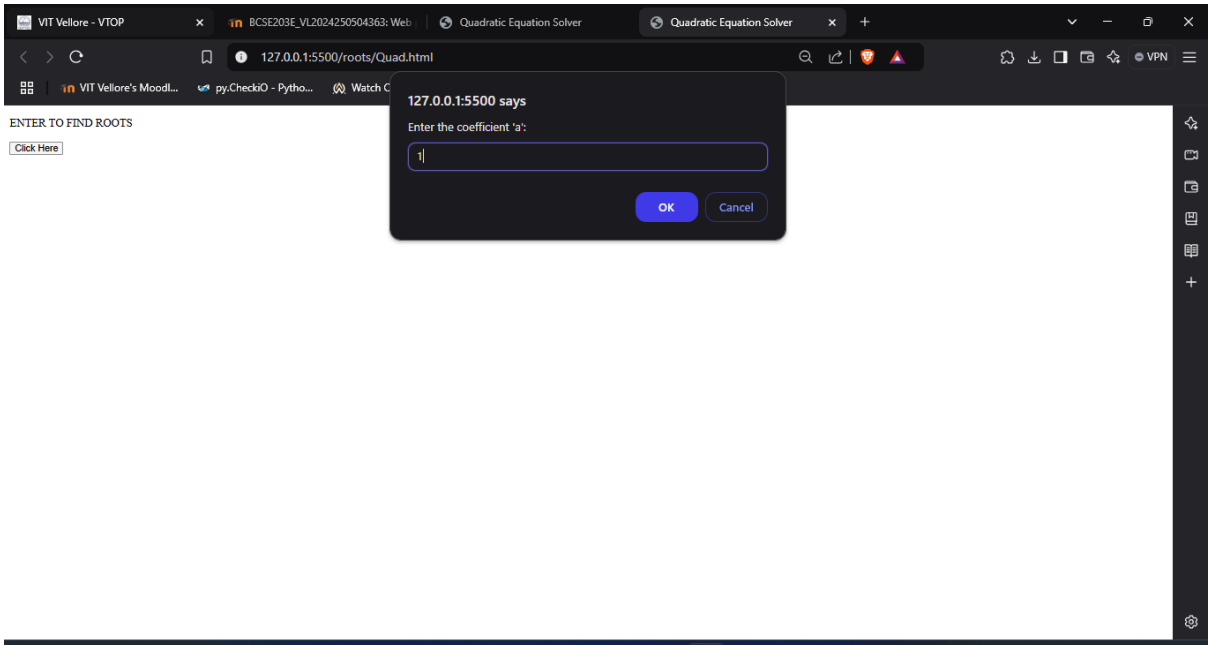
BCSE203E - Web Programming

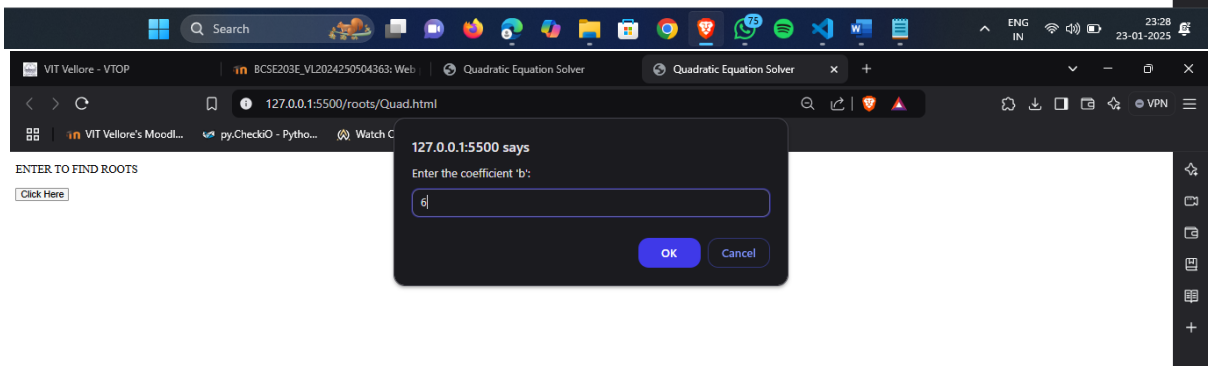
Code:

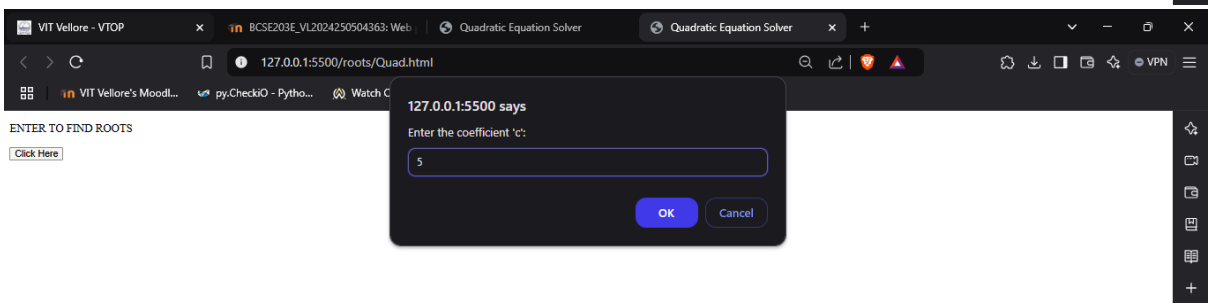
```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Quadratic Equation Solver</title>
</head>
<body>
   <p>ENTER TO FIND ROOTS</p>
   <button onclick="findroots()">Click Here</button>

   <script>
      function findroots() {
         var a = parseFloat(prompt("Enter the coefficient 'a':"));
         var b = parseFloat(prompt("Enter the coefficient 'b':"));
         var c = parseFloat(prompt("Enter the coefficient 'c':"));

         if (a === 0) {
            document.write("This is not a quadratic equation (a cannot be 0).");
            return;
         }

         var discriminant = b * b - 4 * a * c;
         document.write("Discriminant (sr): " + discriminant + "<br>");

         if (discriminant > 0) {
            var root1 = (-b + Math.sqrt(discriminant)) / (2 * a);
            var root2 = (-b - Math.sqrt(discriminant)) / (2 * a);
            document.write("<h1>Roots are real and distinct:</h1>");
            document.write("Root 1: " + root1 + "<br>");
```

```
            document.write("Root 2: " + root2 + "<br>");
        } else if (discriminant === 0) {
            var root = -b / (2 * a);
            document.write("<h1>Roots are real and equal:</h1>");
            document.write("Root: " + root + "<br>");
        } else {
            var realPart = -b / (2 * a);
            var imaginaryPart = Math.sqrt(-discriminant) / (2 * a);
            document.write("<h1>Roots are complex:</h1>");
            document.write("Root 1: " + realPart + " + " + imaginaryPart + "i<br>");
            document.write("Root 2: " + realPart + " - " + imaginaryPart + "i<br>");
        }
    }
  </script>
</body>
</html>
```

Output:







Discriminant (sr): 16

Roots are real and distinct:

Root 1: -1
Root 2: -5

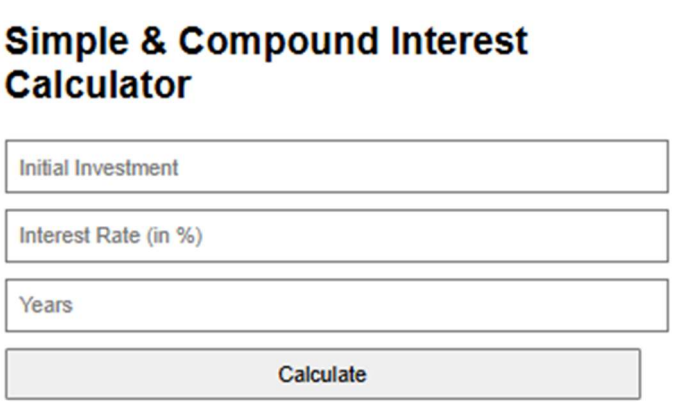
2. Design a Simple Interest Calculator and Compound Interest Calculator webpage using JavaScript function, operators. Use JavaScript to retrieve user input values from the input fields for initial investment, interest rate, and investment years. Update the HTML content dynamically with the calculated result, informing the user about the future value of their investment.

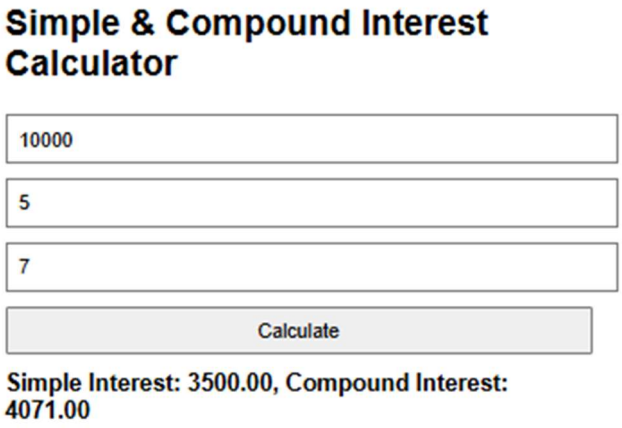
CODE:

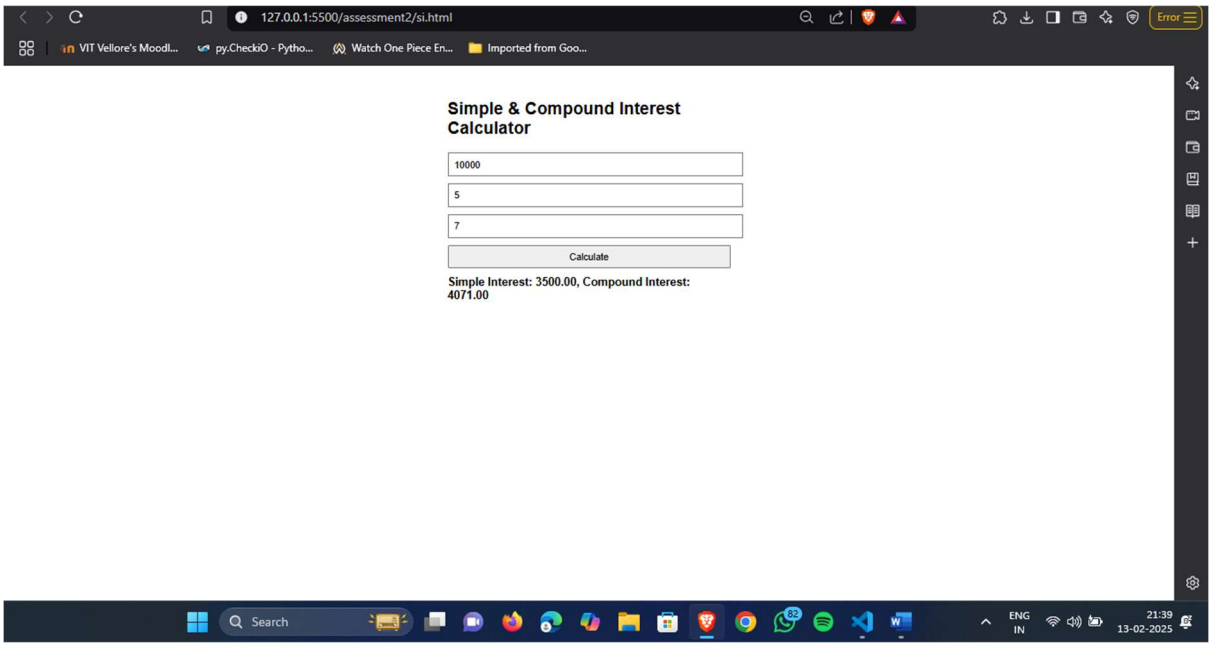
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Interest Calculator</title>
  <style>
    body { font-family: Arial, sans-serif; padding: 20px; }
    .container { max-width: 400px; margin: auto; }
    input, button { margin: 10px 0; display: block; width: 100%; padding: 8px; }
    .result { font-weight: bold; margin-top: 10px; }
  </style>
</head>
<body>
  <div class="container">
    <h2>Simple & Compound Interest Calculator</h2>
    <input type="number" id="principal" placeholder="Initial Investment">
    <input type="number" id="rate" placeholder="Interest Rate (in %)">
    <input type="number" id="years" placeholder="Years">
    <button onclick="calculateInterest()">Calculate</button>
    <p class="result" id="interestResult"></p>
  </div>

  <script>
    function calculateInterest() {
      let p = parseFloat(document.getElementById('principal').value);
      let r = parseFloat(document.getElementById('rate').value) / 100;
      let t = parseFloat(document.getElementById('years').value);
```

```
        let simpleInterest = p * r * t;

        let compoundInterest = p * (Math.pow((1 + r), t) - 1);

            document.getElementById('interestResult').innerText = `Simple Interest: ${simpleInterest.toFixed(2)}, Compound Interest: ${compoundInterest.toFixed(2)}`;

    }

  </script>

</body>

</html>
```

OUTPUT:







3. Design a webpage to accept a positive integer number as input. Write JavaScript to find the following using functions, switch case statement. a. Prime number or not b. Armstrong number or not c. Palindrome or not d. Print smallest even digit e. Print highest odd digit f. Strong number or not

CODE:

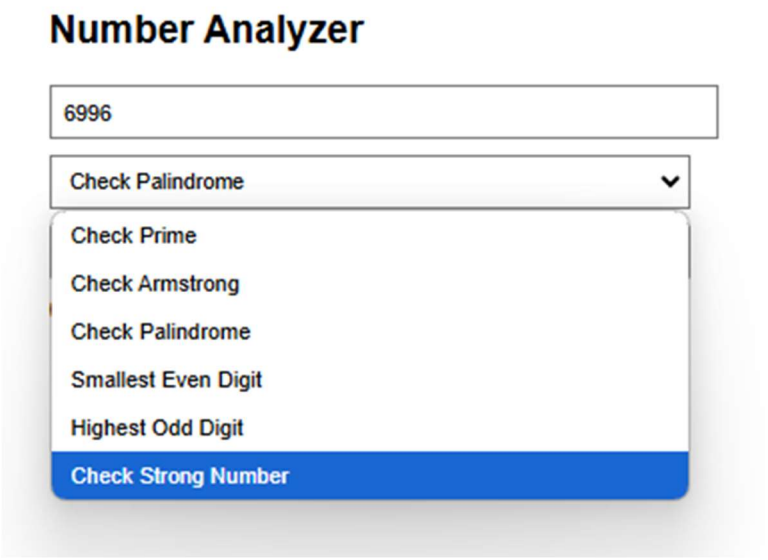
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Number Analyzer</title>
  <style>
    body { font-family: Arial, sans-serif; padding: 20px; }
    .container { max-width: 400px; margin: auto; }
    input, select, button { margin: 10px 0; display: block; width: 100%; padding: 8px; }
    .result { font-weight: bold; margin-top: 10px; }
  </style>
</head>
<body>
  <div class="container">
    <h2>Number Analyzer</h2>
    <input type="number" id="numberInput" placeholder="Enter a positive integer">
    <select id="operation">
      <option value="isPrime">Check Prime</option>
      <option value="isArmstrong">Check Armstrong</option>
      <option value="isPalindrome">Check Palindrome</option>
      <option value="smallestEvenDigit">Smallest Even Digit</option>
      <option value="highestOddDigit">Highest Odd Digit</option>
      <option value="isStrong">Check Strong Number</option>
    </select>
    <button onclick="analyzeNumber()">Analyze</button>
    <p class="result" id="result"></p>
  </div>
```

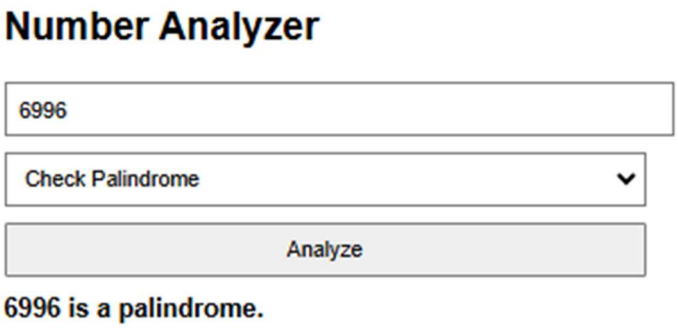
```
<script>
    function analyzeNumber() {
        var num = parseInt(document.getElementById('numberInput').value);
        var operation = document.getElementById('operation').value;
        var resultText = '';

        switch(operation) {
            case 'isPrime':
                resultText = isPrime(num) ? num + ' is a prime number.' : num + ' is not a prime number.';
                break;
            case 'isArmstrong':
                resultText = isArmstrong(num) ? num + ' is an Armstrong number.' : num + ' is not an Armstrong number.';
                break;
            case 'isPalindrome':
                resultText = isPalindrome(num) ? num + ' is a palindrome.' : num + ' is not a palindrome.';
                break;
            case 'smallestEvenDigit':
                var smallestEven = smallestEvenDigit(num);
                resultText = smallestEven !== null ? 'Smallest even digit is ' + smallestEven + '.' : 'No even digits found.';
                break;
            case 'highestOddDigit':
                var highestOdd = highestOddDigit(num);
                resultText = highestOdd !== null ? 'Highest odd digit is ' + highestOdd + '.' : 'No odd digits found.';                break;
            case 'isStrong':
                resultText = isStrong(num) ? num + ' is a strong number.' : num + ' is not a strong number.';
                break;
            default:
                resultText = 'Invalid operation selected.';
        }
        document.getElementById('result').innerText = resultText;
```

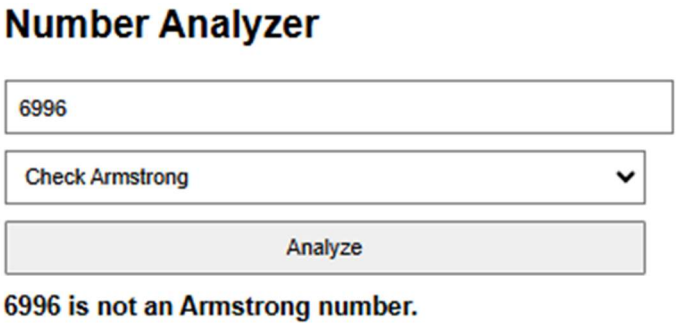
```
}

function isPrime(n) {
    if (n <= 1) return false;
    for (var i = 2; i <= Math.sqrt(n); i++) {
        if (n % i === 0) return false;
    }
    return true;
}

function isArmstrong(n) {
    var sum = 0;
    var temp = n;
    var digits = n.toString().length;
    while (temp > 0) {
        var remainder = temp % 10;
        sum += Math.pow(remainder, digits);
        temp = Math.floor(temp / 10);
    }
    return sum === n;
}       function isPalindrome(n) {
    var str = n.toString();
    var reversedStr = str.split('').reverse().join('');
    return str === reversedStr;
}
function smallestEvenDigit(n) {
    var digits = n.toString().split('');
    var evenDigits = digits.filter(function(digit) {
        return parseInt(digit) % 2 === 0;
    });          if (evenDigits.length === 0) return null;
    return Math.min.apply(null, evenDigits.map(Number));
}
```

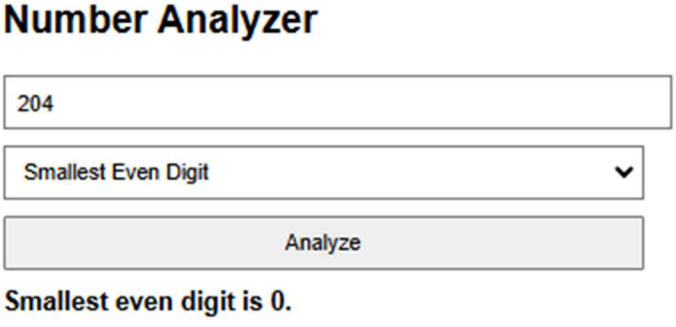
```
    function highestOddDigit(n) {
      var digits = n.toString().split('');
      var oddDigits = digits.filter(function(digit) {
        return parseInt(digit) % 2 !== 0;
      });
      if (oddDigits.length === 0) return null;
      return Math.max.apply(null, oddDigits.map(Number));
    }

    function isStrong(n) {
      var sum = 0;
      var temp = n;
      while (temp > 0) {
        var remainder = temp % 10;
        sum += factorial(remainder);
        temp = Math.floor(temp / 10);}
      return sum === n; }

    function factorial(x) {
      if (x === 0 || x === 1) return 1;
      var fact = 1;
      for (var i = 2; i <= x; i++) {
        fact *= i;   }
      return fact;
    }
  </script>
</body>
</html>
```

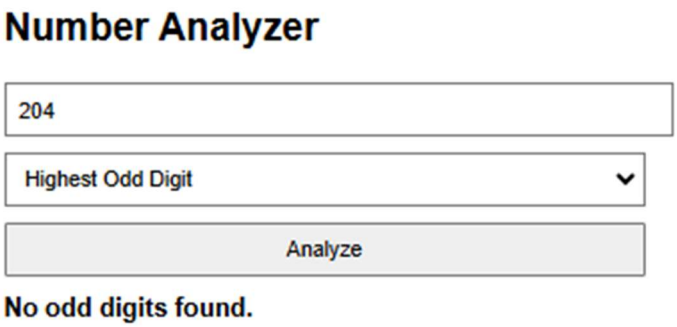
OUTPUT:


Number Analyzer
6996
Check Palindrome
Check Prime
Check Armstrong
Check Palindrome
Smallest Even Digit
Highest Odd Digit
Check Strong Number


Number Analyzer
6996
Check Palindrome
Analyze
6996 is a palindrome.


Number Analyzer
6996
Check Armstrong
Analyze
6996 is not an Armstrong number.


Number Analyzer
204
Smallest Even Digit
Analyze
Smallest even digit is 0.


Number Analyzer
204
Highest Odd Digit
Analyze
No odd digits found.