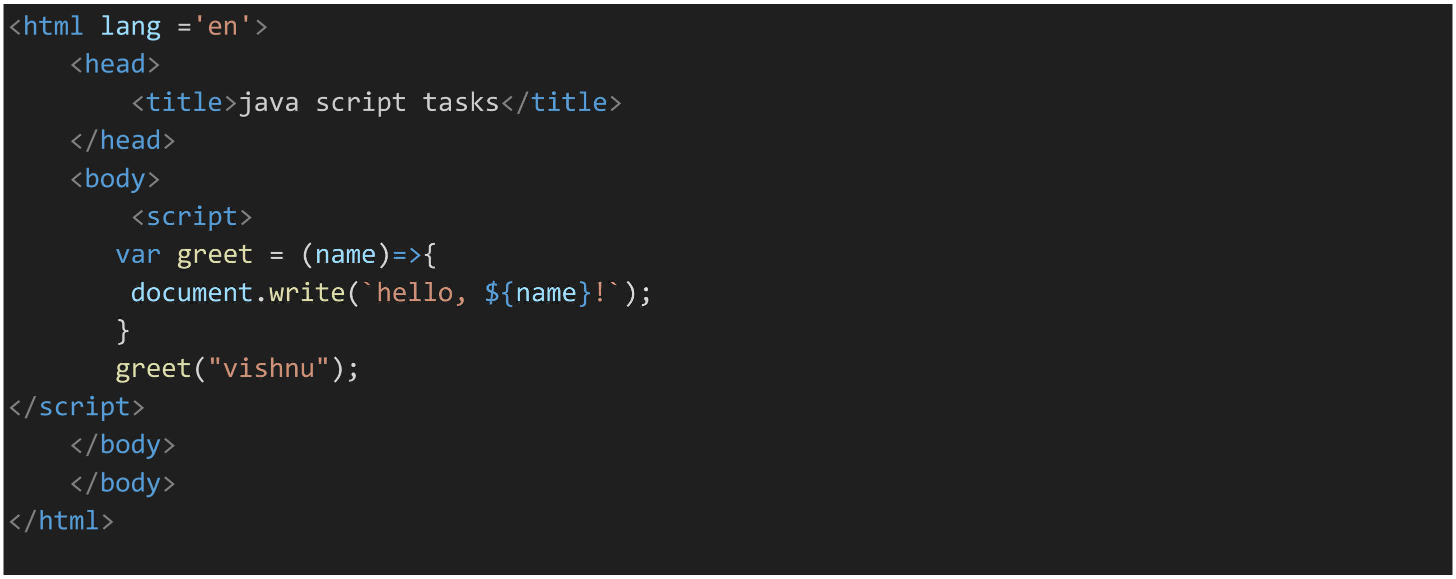
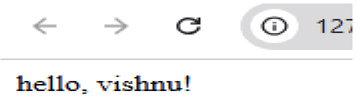
Task 51: Declare a simple arrow function named greet that takes one parameter name and returns the string “Hello, name!”. Test your function with various names.



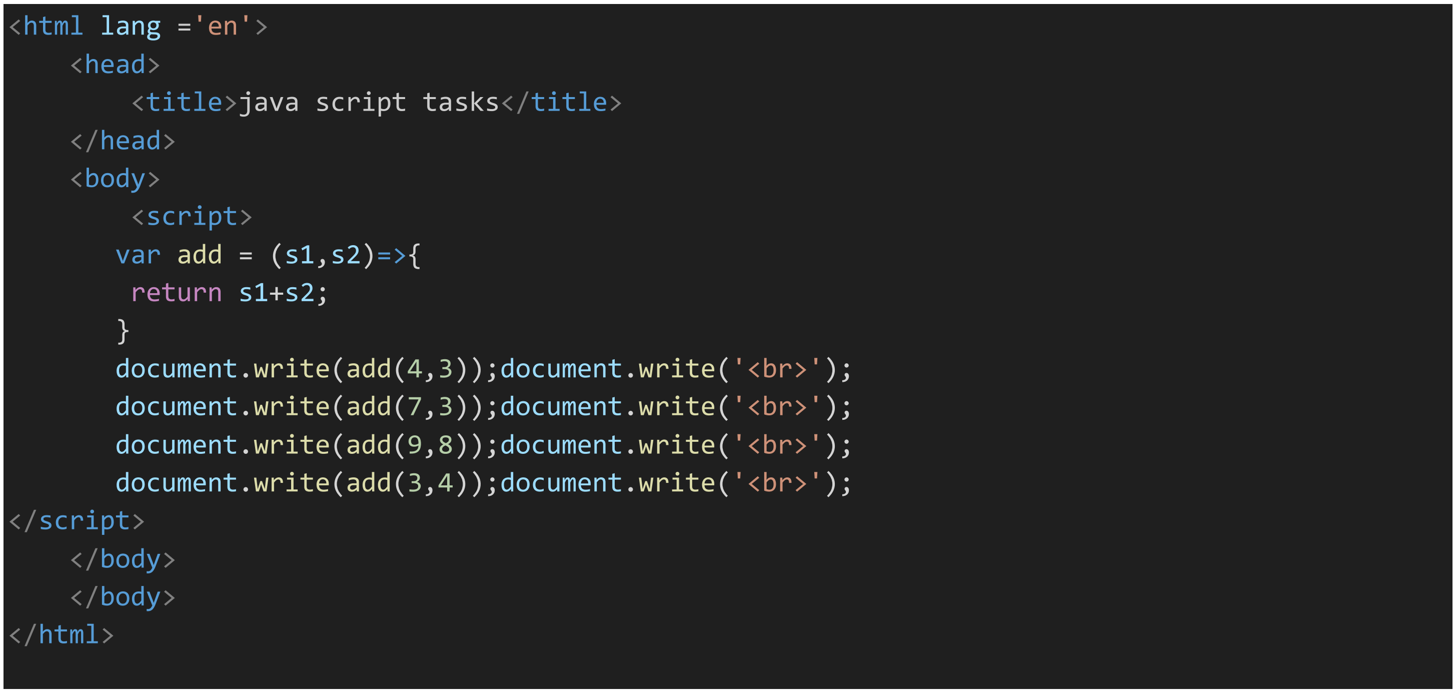
```
<html lang ='en'>
    <head>
        <title>java script tasks</title>
    </head>
    <body>
        <script>
       var greet = (name)=>{
        document.write(`hello, ${name}!`);
       }
       greet("vishnu");
</script>
    </body>
    </body>
</html>
```

Output:



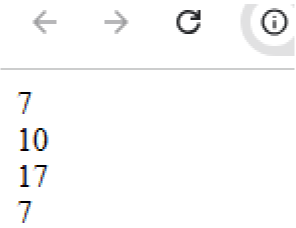
hello, vishnu!

Task 52: Write an arrow function named add that takes two parameters and returns their sum. Validate your function with several pairs of numbers



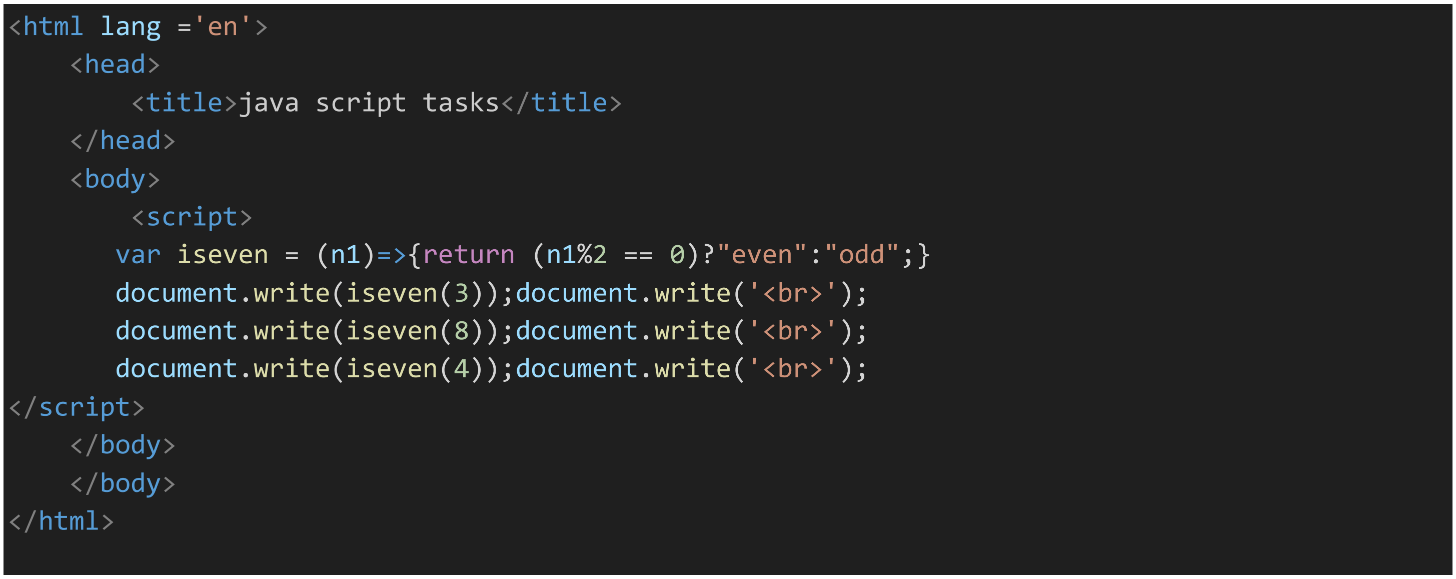
```
<html lang ='en'>
    <head>
        <title>java script tasks</title>
    </head>
    <body>
        <script>
       var add = (s1,s2)=>{
        return s1+s2;
       }
       document.write(add(4,3));document.write('<br>');
       document.write(add(7,3));document.write('<br>');
       document.write(add(9,8));document.write('<br>');
       document.write(add(3,4));document.write('<br>');
</script>
    </body>
    </body>
</html>
```

Output:



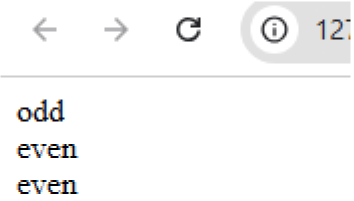
```
7
10
17
7
```

Task 53: Declare an arrow function named isEven that checks if a number is even. If the number is even, it should return true; otherwise, false. Remember that if the arrow function body has a single statement, you can omit the curly braces



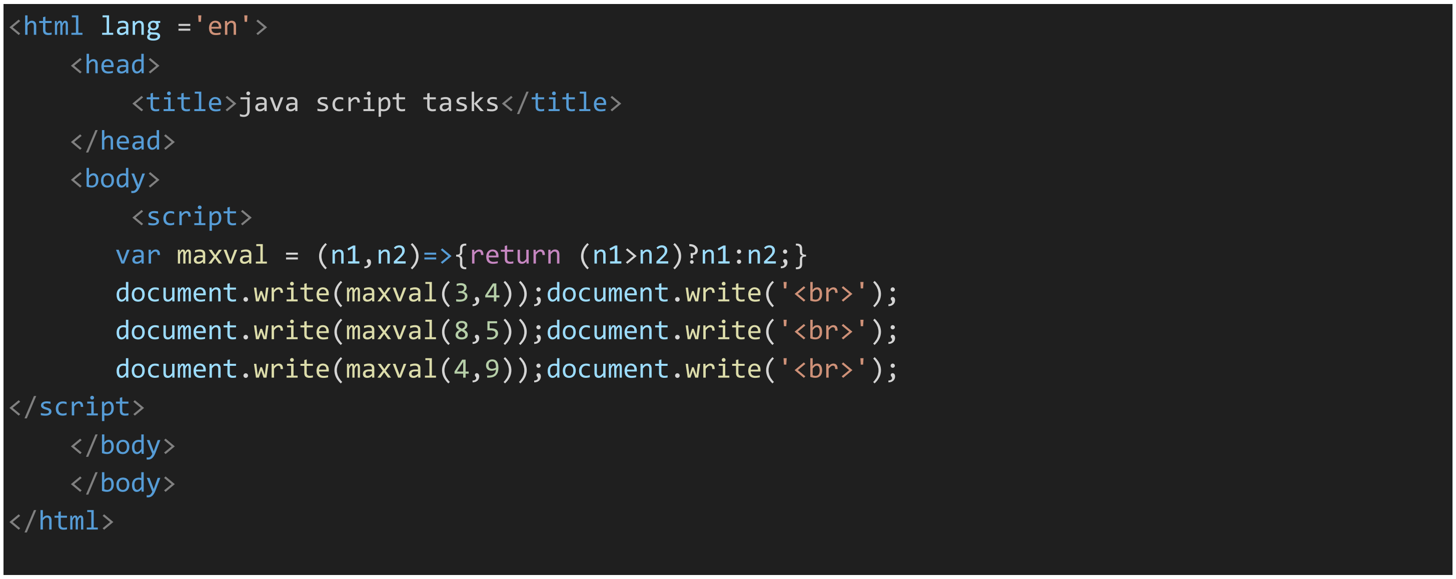
```
<html lang ='en'>
    <head>
        <title>java script tasks</title>
    </head>
    <body>
        <script>
       var iseven = (n1)=>{return (n1%2 == 0)?"even":"odd";}
       document.write(iseven(3));document.write('<br>');
       document.write(iseven(8));document.write('<br>');
       document.write(iseven(4));document.write('<br>');
</script>
    </body>
    </body>
</html>
```

Output:



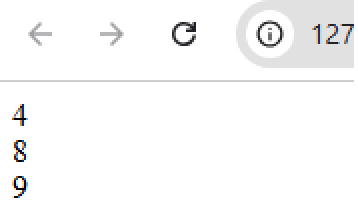
```
odd
even
even
```

Task 54: Implement an arrow function named maxValue that takes two numbers as parameters and returns the larger number. Here, you'll need to use curly braces for the function body and the return statement.



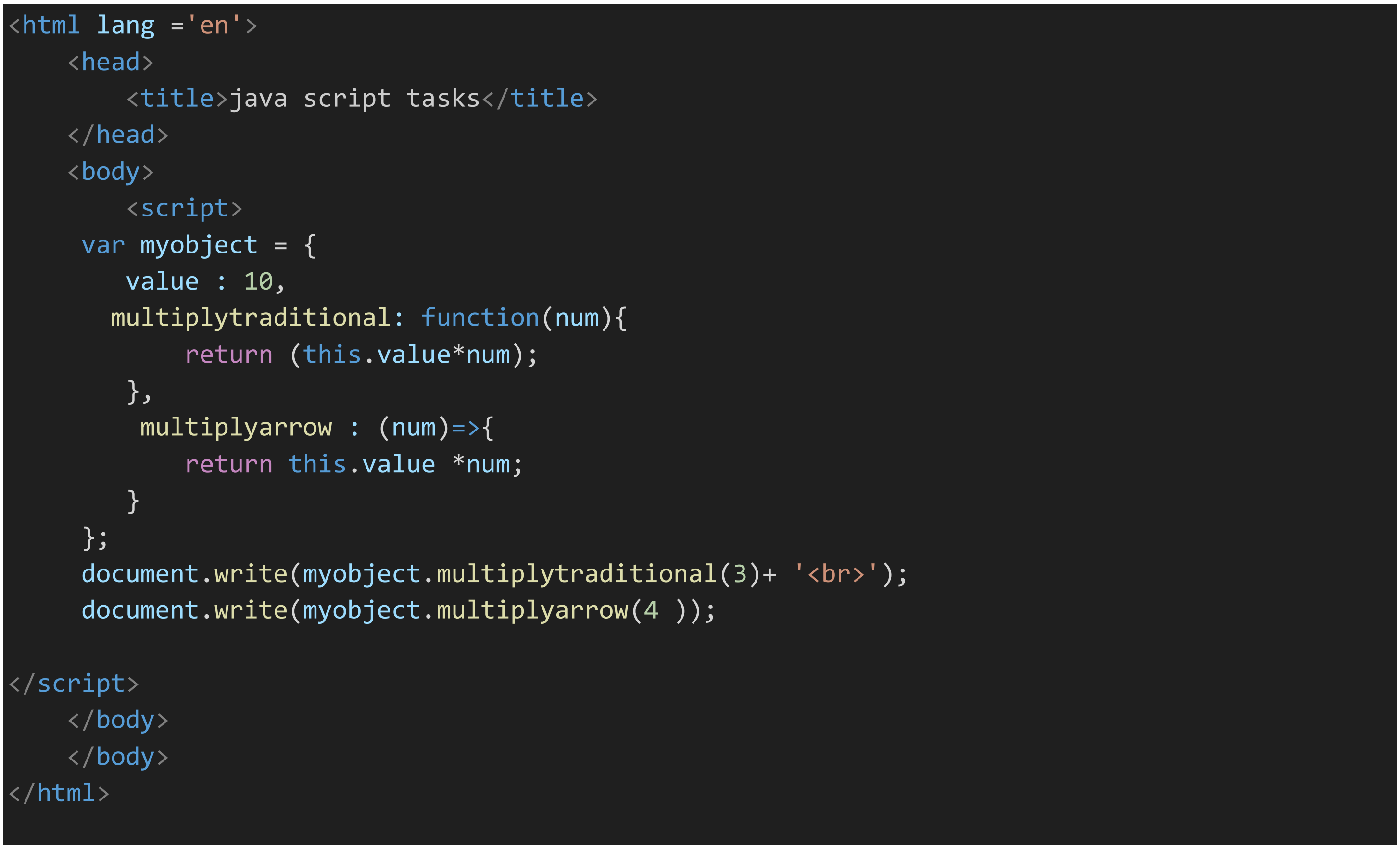
```
<html lang ='en'>
    <head>
        <title>java script tasks</title>
    </head>
    <body>
        <script>
       var maxval = (n1,n2)=>{return (n1>n2)?n1:n2;}
       document.write(maxval(3,4));document.write('<br>');
       document.write(maxval(8,5));document.write('<br>');
       document.write(maxval(4,9));document.write('<br>');
</script>
    </body>
    </body>
</html>
```

Output:



```
4
8
9
```

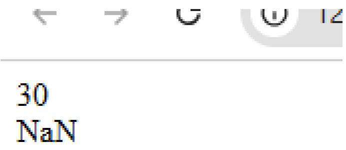
Task 55: Examine the behavior of the this keyword inside an arrow function vs a traditional function. Create an object named myObject with a property value set to 10 and two methods: multiplyTraditional using a traditional function and multiplyArrow using an arrow function. Both methods should attempt to multiply the value property by a number passed as a parameter. Check the value of this inside both methods.



```
<html lang ='en'>
    <head>
        <title>java script tasks</title>
    </head>
    <body>
        <script>
     var myobject = {
        value : 10,
       multiplytraditional: function(num){
            return (this.value*num);
        },
         multiplyarrow : (num)=>{
            return this.value *num;
        }
     };
     document.write(myobject.multiplytraditional(3)+ '<br>');
     document.write(myobject.multiplyarrow(4 ));

</script>
    </body>
    </body>
</html>
```

Output:



```
30
NaN
```