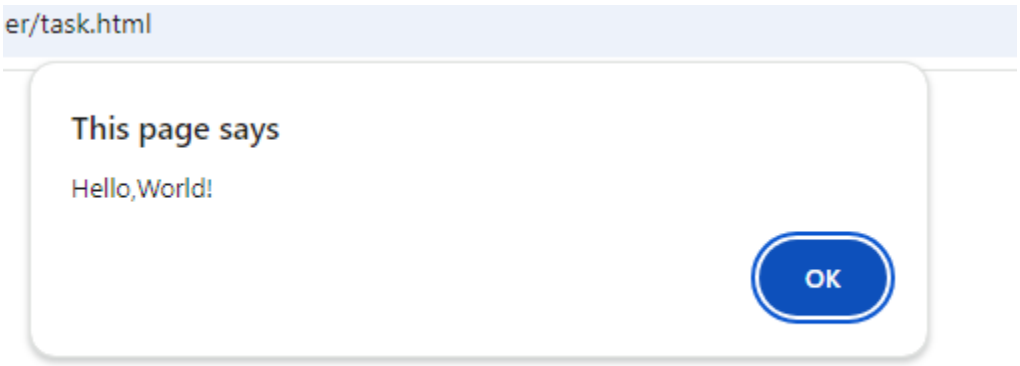
TASK 1:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
          alert("Hello,World!");
    </script>
  </body>
</html>
```

OUTPUT:

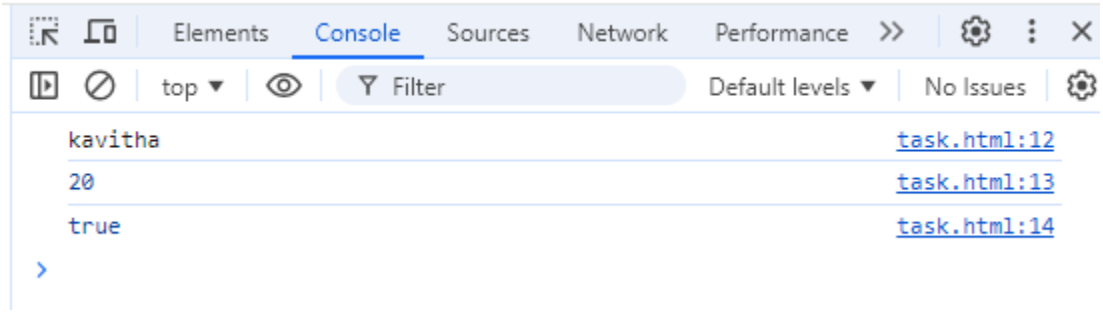


TASK 2:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
        let name= "kavitha";
        let number = 20;
        let  bool = true;
     console.log(name);
     console.log(number);
     console.log(bool);
    </script>
  </body>
```

```
</html>
```

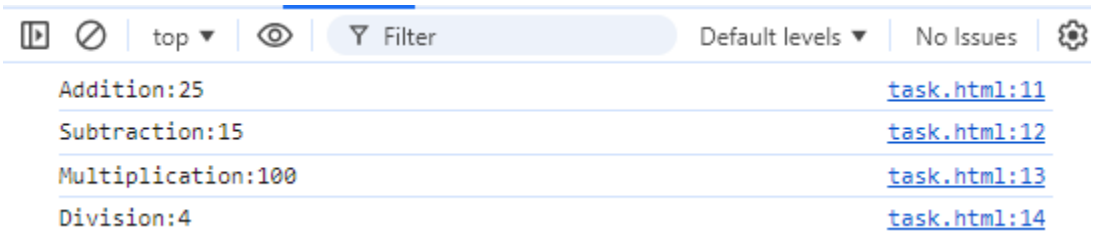
OUTPUT:



TASK 3:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
          let a=20;
          let b=5;
          console.log("Addition:"+ (a+b));
          console.log("Subtraction:"+ (a-b));
          console.log("Multiplication:"+ (a*b));
          console.log("Division:"+ (a/b));
    </script>
  </body>
</html>
```

OUTPUT:

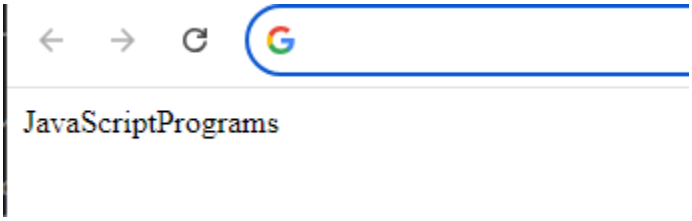


TASK 4:

```
<!DOCTYPE html>
<html>
```

```
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
          let string1 = "JavaScript";
          let string2 = "Programs";
          document.writeln(string1 + string2);
    </script>
  </body>
</html>
```

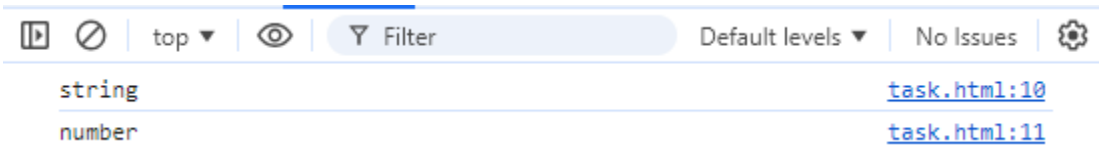
OUTPUT:



TASK 5:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
          console.log(typeof("Nalsa"));
          console.log(typeof(30));
    </script>
  </body>
</html>
```

OUTPUT:



TASK 6:

```
<!DOCTYPE html>
<html>
```

```
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
          // This is single line comment
          /* This is multi-line comment */
    </script>
  </body>
</html>
```

OUTPUT:

Single-line Comment:

Syntax: // (in languages like C++, Java, JavaScript) or # (in Python).

Scope: Quick explanations or annotations for one-liners.

Multi-line Comment:

Syntax: /* to start and */ to end (in languages like C++, Java, JavaScript).

Scope: Used for block-level descriptions or commenting out larger sections of code.

TASK 7:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
          let num = 45;
          let string = "Malar"
          document.writeln(num +"<br>");
          document.writeln(string);
    </script>
  </body>
</html>
```

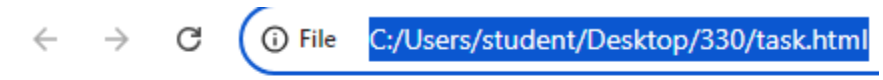
OUTPUT:

45
Malar

TASK 8:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
          let mark = 80;
          if(mark >=50)
          {
            document.writeln("The student has passed"+ "<br>");
            if(mark>=75)
          {
            document.writeln("The student score good marks"+ "<br>");
          }
          else{
            document.writeln("The student try to get good marks"+ "<br>");
          }
          }
          else{
            document.writeln("The student has failed");
          }
    </script>
  </body>
</html>
```

OUTPUT:



The student has passed
The student score good marks

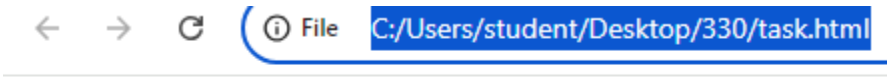
TASK 9:

```
<!DOCTYPE html>
<html>
```

```
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
        let name= " Nalsa ", name1 = " Nithya " ,  bool = true;
        document.writeln(name , name1 , bool);

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

OUTPUT:

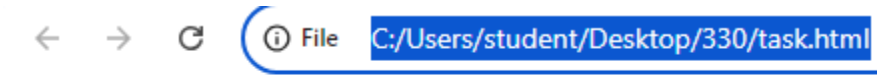


Nalsa Nithya true

TASK 10:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
        let str = "Welcome to javascript programming!";
        document.writeln(str);
    </script>
  </body>
</html>
```

OUTPUT:



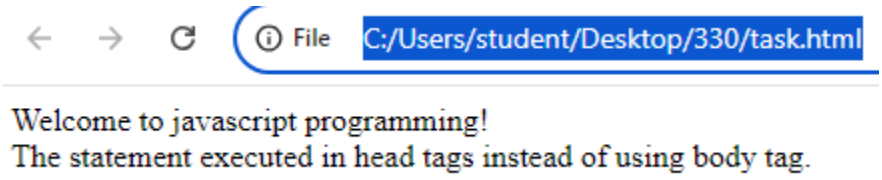
Welcome to javascript programming!

```
<!DOCTYPE html>
```

```
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
<script>
    let str = "Welcome to javascript programming!";
    document.writeln(str + "<br>");

</script>
  </head>
  <body>
         The statement executed in head tags instead of using body tag.
  </body>
</html>
```

OUTPUT:



Execution Timing:

<head>: The script executes before the content is loaded, blocking page rendering until the script finishes.

<body>: The script executes after the content is rendered, allowing the page to load and display to the user without delay.

Impact on Page Load:

<head>: Can slow down the perceived page load time as it blocks rendering until the script is fully executed.

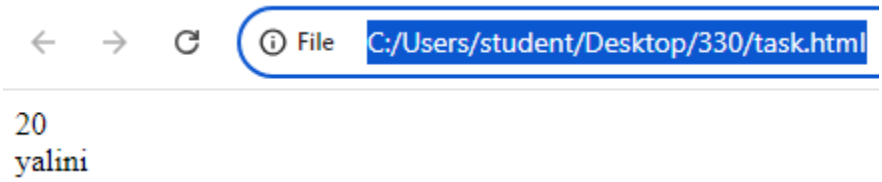
<body>: Improves perceived load time because the page content is displayed first, with the script running afterward**.**

TASK 11:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
```

```
  <body>
    <script>
        num = 20;
        name= "yalini";
        document.writeln(num+ "<br>");
        document.writeln(name+ "<br>");
    </script>
  </body>
</html>
```

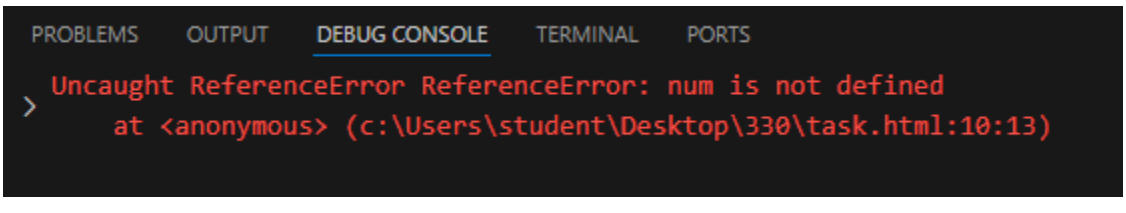
OUTPUT:



TASK 12:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
        "use strict";
        num = 20;
        name= "yalini";
        document.writeln(num+ "<br>");
        document.writeln(name+ "<br>");
    </script>
  </body>
</html>
```

OUTPUT:



TASK 13:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
        "use strict";
       let name= "kavitha";
       delete name;
       "use strict";
       function myfunction(){
        retun Kavi;
       }
       delete myfunction;
       "use strict";
       function myfunction( This statement executed sucessfully)
       delete myfunction;
    </script>
  </body>
</html>
```

OUTPUT:

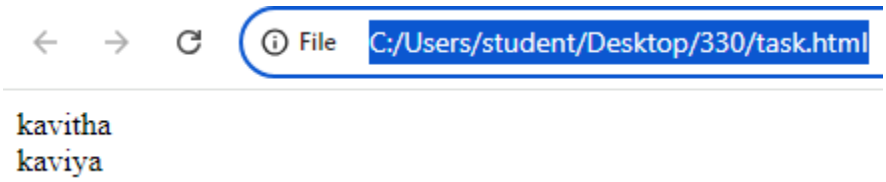
```
PROBLEMS 8   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS
> Uncaught SyntaxError SyntaxError: Delete of an unqualified identifier in strict mode.
    at (program) (c:\Users\student\Desktop\330\task.html:11:15)
```

TASK 14:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
        name= "kavitha";
       document.writeln(name+ "<br>");
       "use strict";
        name1= "kaviya";
       document.writeln(name1);
    </script>
```

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

OUTPUT:

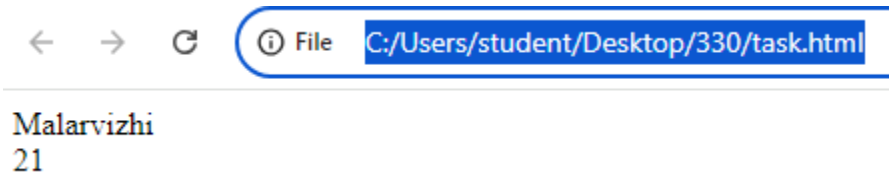


TASK 15:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
       "use strict";
       let  str = "Malarvizhi";
       var number = 21;

       document.writeln(str +"<br>");
       document.writeln(number);
    </script>
  </body>
</html>
```

OUTPUT:

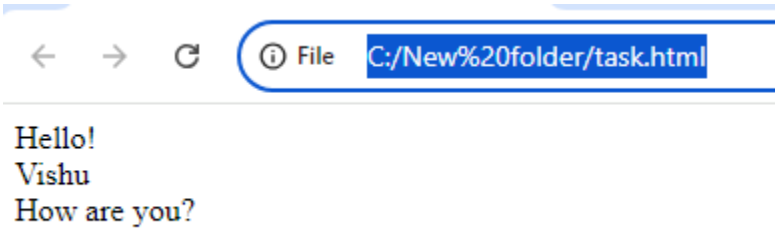


TASK 16:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
```

```
        let s = "Hello!";
        var string = "vishu";
        const str = " How are you?";
        document.writeln(s + "<br>");
        document.writeln(string+ "<br>");
        document.writeln(str+ "<br>");
    </script>
  </body>
</html>
```

OUTPUT:



]

Use **var** only when you need compatibility with legacy code or specific behavior regarding function or global scope (but generally avoid it in new code).

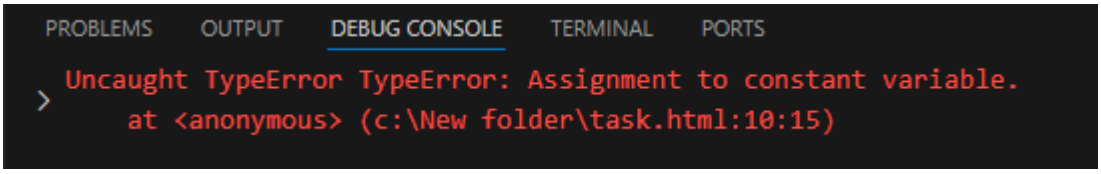
Use **let** for variables that are re-assigned, especially when the scope is block-level (like in loops or conditionals).

Use **const** for variables that should not be reassigned, which helps with readability and maintainability by signaling that a variable's value will remain constant.

TASK 17:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
        const str = " How are you?";
          str = "What are you doing?";
        document.writeln(str+ "<br>");
    </script>
  </body>
</html>
```

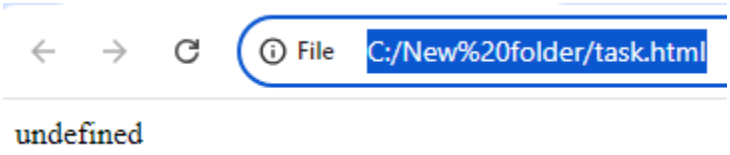
OUTPUT:



TASK 18:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
         var value ;
        document.writeln(value+ "<br>");
    </script>
  </body>
</html>
```

OUTPUT:

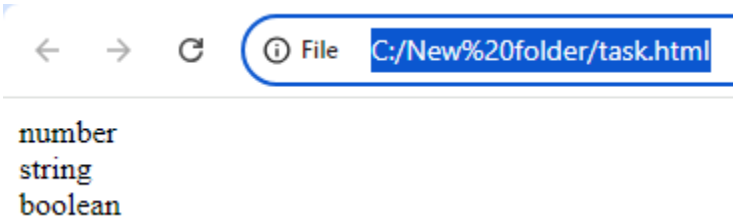


TASK 19:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
         var value  = 35 ;
         var str1 = "Geetha";
         var b = true ;
        document.writeln(typeof(value)+ "<br>");
        document.writeln(typeof(str1)+ "<br>");
        document.writeln(typeof(b)+ "<br>");
    </script>
```

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

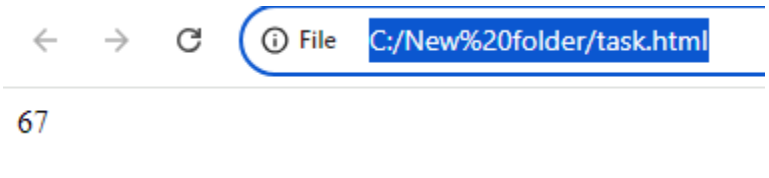
OUTPUT:



TASK 20:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
         var value  = 35 ;
          num = 67 ;
        document.writeln(num+ "<br>");
    </script>
  </body>
</html>
```

OUTPUT:



TASK 21:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
        var name = "Hi";
```

```
         var value  = 35 ;
        let bool = true ;
        let a = null ;
        var x;
        let manager ={
            name:"hema",
            age : 21
        };
        document.writeln(name+ "<br>");
        document.writeln(value+ "<br>");
        document.writeln(bool+ "<br>");
        document.writeln(a+ "<br>");
        document.writeln(x+ "<br>");
        document.writeln(manager.name+ "<br>");
        document.writeln(manager+ "<br>");
    </script>
  </body>
</html>
```

OUTPUT:

Hi
35
true
null
undefined
hema
[object Object]

TASK 22:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
        var name = "Hi";
         var value  = 35 ;
        let bool = true ;
        let a = null ;
```

```
        var x;
        document.writeln(typeof(name)+ "<br>");
        document.writeln(typeof(value)+ "<br>");
        document.writeln(typeof(bool)+ "<br>");
        document.writeln(typeof(a)+ "<br>");
        document.writeln(typeof(x)+ "<br>");
    </script>
  </body>
</html>
```

OUTPUT:

string
number
boolean
object
undefined

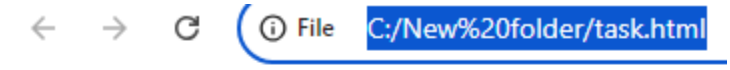
TASK 23:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
        var symbol = "Hello!!";
        var value  = 43 ;
        let bool = true ;
        var x;

        document.writeln(typeof(symbol)+ "<br>");
        document.writeln(typeof(value)+ "<br>");
        document.writeln(typeof(bool)+ "<br>");
        document.writeln(typeof(x)+ "<br>");

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

OUTPUT:



string
number
boolean
undefined

TASK 24:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
        let name = null;
        document.writeln(typeof(name)+ "<br>");

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

OUTPUT:

object

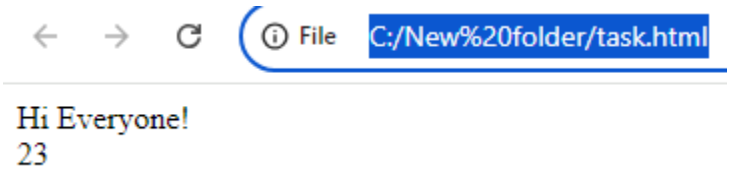
TASK 25:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
        var Str = "Hi Everyone!";
        let a = 23;
        document.writeln(Str+ "<br>");
```

```
        document.writeln(a+"<br>");

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

OUTPUT:



VAR:

Scope : Function scope or global scope

Hoisting : Hoisted and initialized to undefined

Re-declaration : Allowed in the same scope

LET:

Scope : Block scope (inside {} block)

Hoisting : Hoisted but not initialized (temporal dead zone)

Re-declaration : Not allowed in the same scope

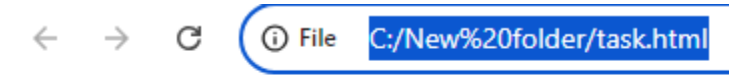
TASK 26:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
        let a = 23;
        let num = a* 2;
        document.writeln(num+ "<br>");

        let b = 20;
        let number = b % 3;
        document.writeln(number+ "<br>");
    </script>
  </body>
```

```
</html>
```

OUTPUT:



46
2

TASK 27:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>

        let bool = true;
        let str = String(bool);
        document.writeln(str+ "<br>");

        let str1 = "hi";
        let bool2 = Boolean(str1);
        document.writeln(bool2+ "<br>");
    </script>
  </body>
</html>
```

OUTPUT:

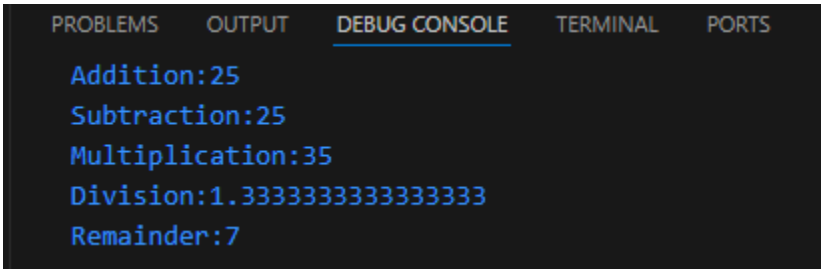
true
true

TASK 28:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
```

```
    <script>
       let a=20;
          let b=5;
          console.log("Addition:"+ (a+b));
          let c = 45;
          console.log("Subtraction:"+ (c-a));
          let d= 7;
          console.log("Multiplication:"+ (d*b));
          let x = 15 ;
          console.log("Division:"+ (a/x));
          let y = 37;
          console.log("Remainder:"+(y%x));
    </script>
  </body>
</html>
```

OUTPUT:

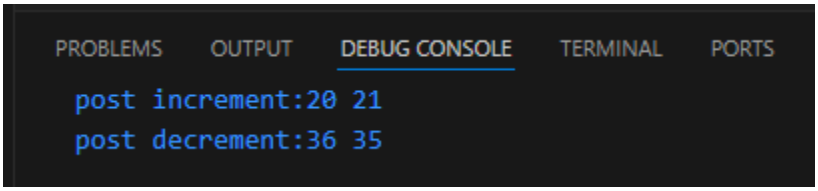


TASK 29:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
          let a=20;
          console.log("post increment:" + a++,a);

          let b = 36;
          console.log("post decrement:" + b--,b);
    </script>
  </body>
</html>
```

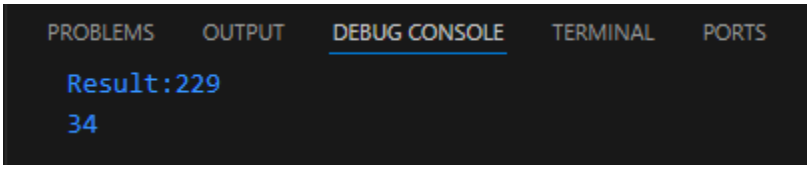
OUTPUT:



TASK 30:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
          let a=20;
          let x = 3;
          let y = 74;
          let b = 36;
          let result = x *y + --a % (b/x);
          console.log("Result:" + result);
          let set = b*(x % a) - y;
          console.log(set);
    </script>
  </body>
</html>
```

OUTPUT:

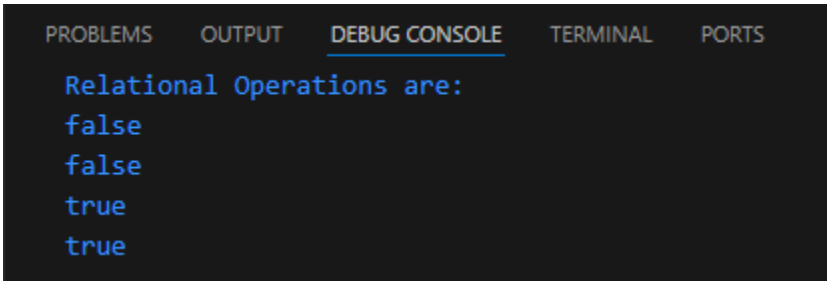


TASK 31:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
```

```
  <body>
    <script>
          let a=20;
          let b = 36;
          console.log("Relational Operations are:");
          console.log(a>b);
          console.log(a>=b);
          console.log(a<b);
          console.log(a<=b);
    </script>
  </body>
</html>
```

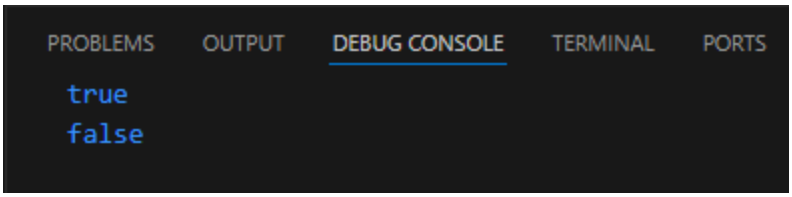
OUTPUT:



TASK 32:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
          let a="20";
          let b = 20;
          console.log(a==b);
          console.log(a===b);
    </script>
  </body>
</html>
```

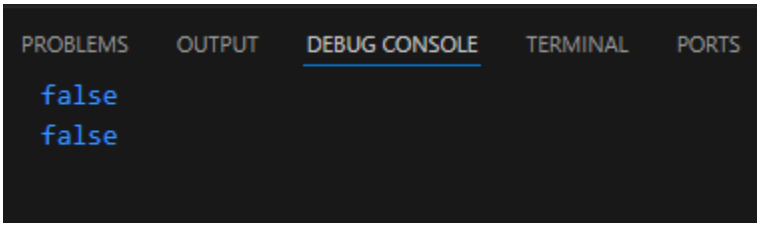
OUTPUT:



TASK 33:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
          var string1 = "Hello";
          var string2 = "hello";
          console.log(string1 === string2);
          console.log(string1 == string2);
    </script>
  </body>
</html>
```

OUTPUT:



TASK 34:

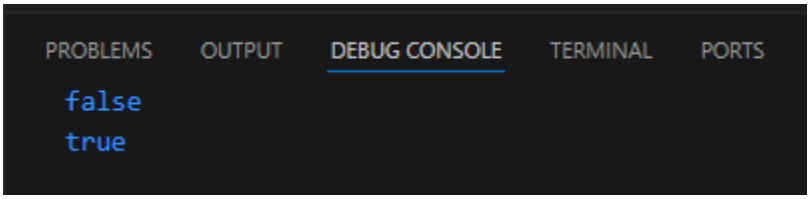
```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>

          let a="20";
          let b = 20;
```

```
          console.log(a!=b);
          console.log(a!==b);

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

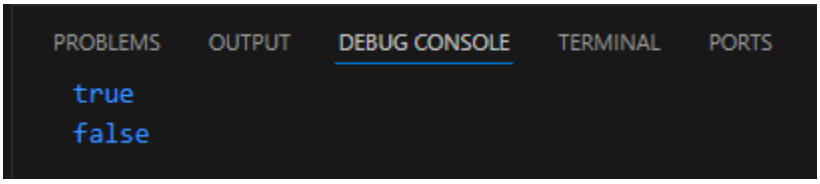
OUTPUT:



TASK 35:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
          let a= null;
          let b ;
          console.log(a==b);
          console.log(a===b);
    </script>
  </body>
</html>
```

OUTPUT:

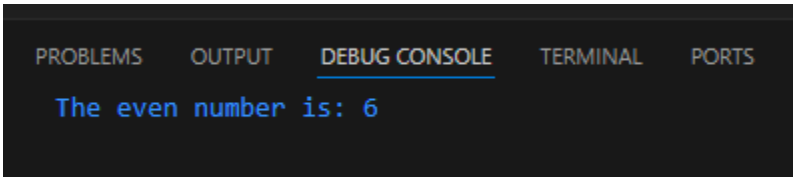


TASK 36:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
```

```
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
          let num = 6;
          if(num%2==0)
    {
        console.log("The even number is:",+ num);
    }
    else
    {
        console.log("The odd number is:",+ num)
    }
    </script>
  </body>
</html>
```

OUTPUT:

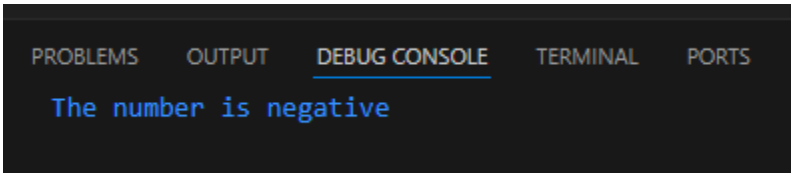


TASK 37:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
          let num = -3;
          if(num > 0)
    {
        console.log("The  number is positive");
    }
    else if(num < 0)
    {
        console.log("The number is negative");
    }
    else{
        console.log("The number is zero");  }
    </script>
```

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

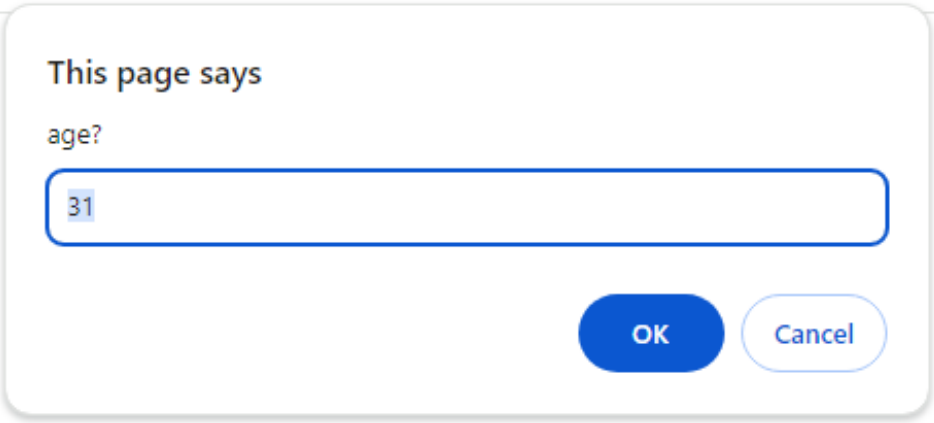
OUTPUT:

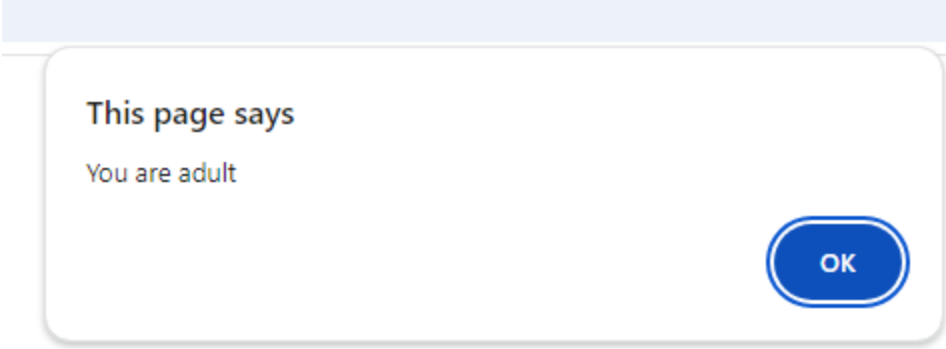


TASK 38:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
          let age = prompt("age?", 31);
          let  result = (age < 18)? "You are young" : (age > 18)? "You are adult"
: "Eligible to vote";
          alert(result);
    </script>
  </body>
</html>
```

OUTPUT:

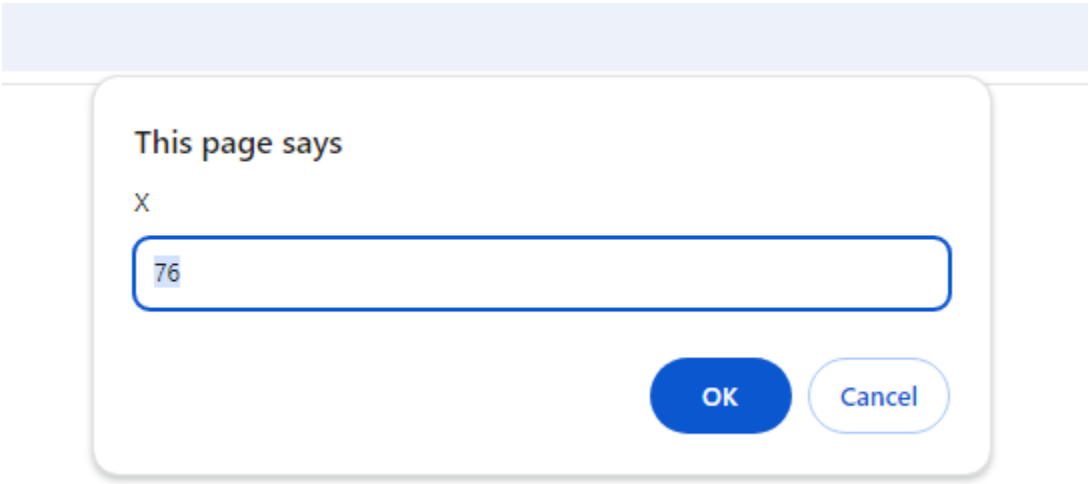


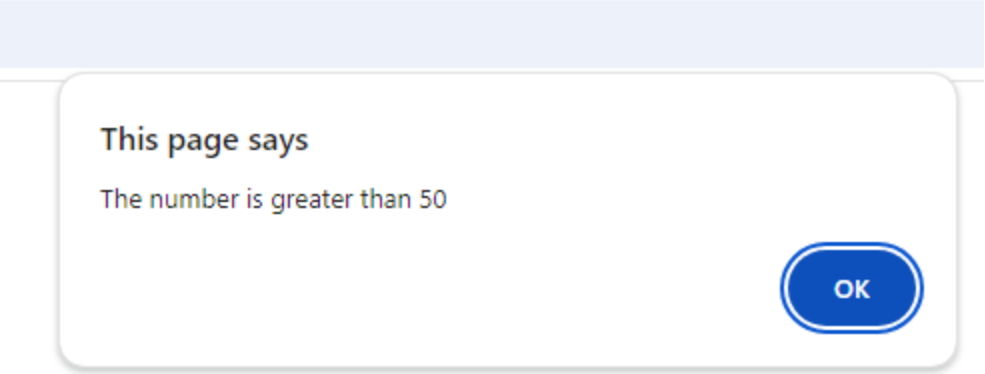


TASK 39:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
          let X = prompt("X", 76);
          let  result = (X < 50)? "The number is less than 50" : (X > 50)? "The
number is greater than 50 " : "Half of 100";
          alert(result);
    </script>
  </body>
</html>
```

OUTPUT:

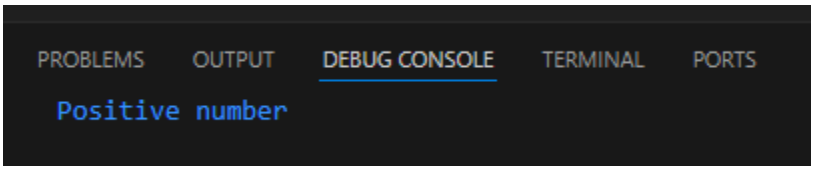




TASK 40:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
          let X = 5;
          let  result = (X >= 0)? "Positive number" : "negative number";
          console.log(result);
    </script>
  </body>
</html>
```

OUTPUT:

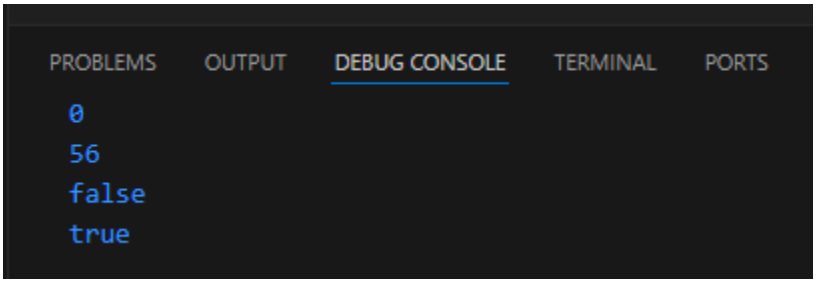


TASK 41:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
```

```
        let X = 56;
        let y = 0;
        console.log( X && y);
        console.log( X || y);
        console.log(!X);
        console.log(!y);
    </script>
  </body>
</html>
```

OUTPUT:

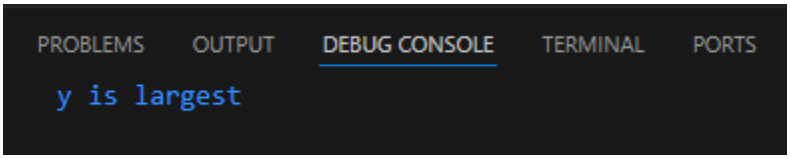


TASK 42:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
        let x = 12;
        let y = 67;
        let z = 45;
        if( x>y && x>z)
        {
          console.log("X is largest");
        }
        else if(y> z)
        {
          console.log("y is largest");
        }
        else{
          console.log("z is largest");
        }
    </script>
```

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

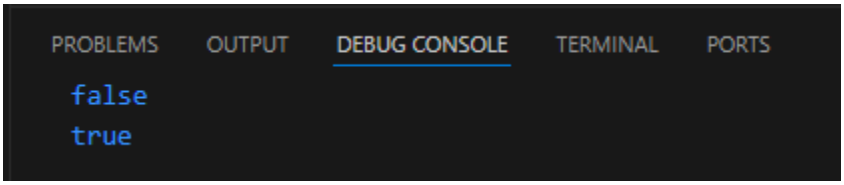
OUTPUT:



TASK 43:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
          let bool = true;
          console.log(!bool);
          let bool1 = false;
          console.log(!bool1);
    </script>
  </body>
</html>
```

OUTPUT:

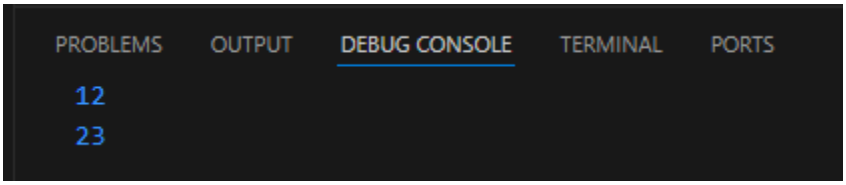


TASK 44:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
          let A = 23;
```

```
        let B = 12;
        console.log( A && B);
        console.log( A || B);
    </script>
  </body>
</html>
```

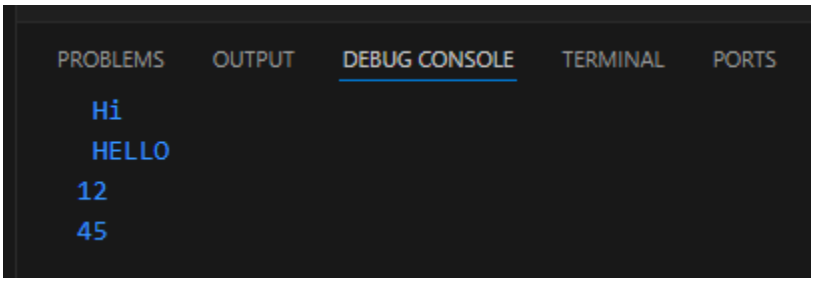
OUTPUT:



TASK 45:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
        let str = " HELLO";
        let str1 = " Hi";
        console.log(str && str1);
        console.log(str || str1);
        let A = 45;
        let B = 12;
        console.log(A && B);
        console.log(A || B);
    </script>
  </body>
</html>
```

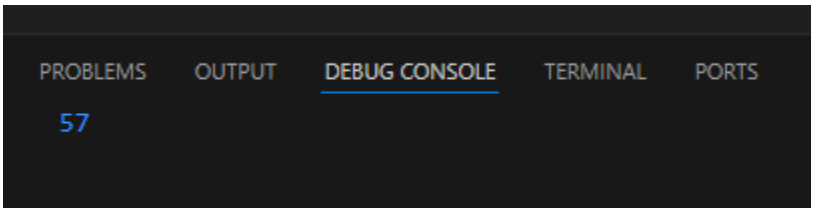
OUTPUT:



TASK 46:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
          let A = 45;
          let B = 12;
          let sum = function(A,B)
          {
            return A+B;
          }
          console.log(sum(A,B));
    </script>
  </body>
</html>
```

OUTPUT:

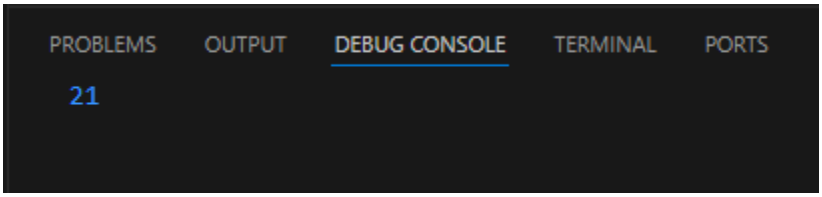


TASK 47:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
          let length = 7;
          let Width = 3;
          let area = function(length ,width)
          {
            return  length * width;
          }
          console.log(area(length , width));
```

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

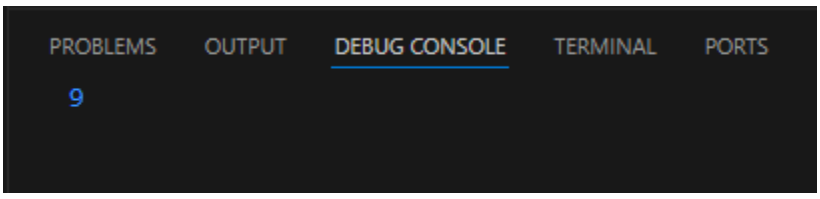
OUTPUT:



TASK 48:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
         let a =2;
         let b =7;
          let result = function()
          {
            console.log(a + b);
          }
           result ();
    </script>
  </body>
</html>
```

OUTPUT:



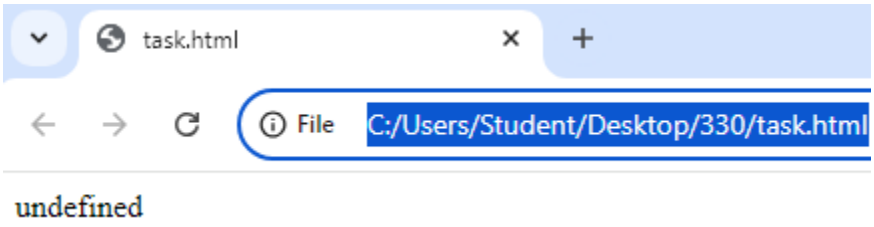
TASK 49:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
```

```
  </head>
  <body>
    <script>
         let a =2;
         let b =7;
          let result = function(a,b)
          {
            return;
          }
          document.writeln(result(a,b));

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

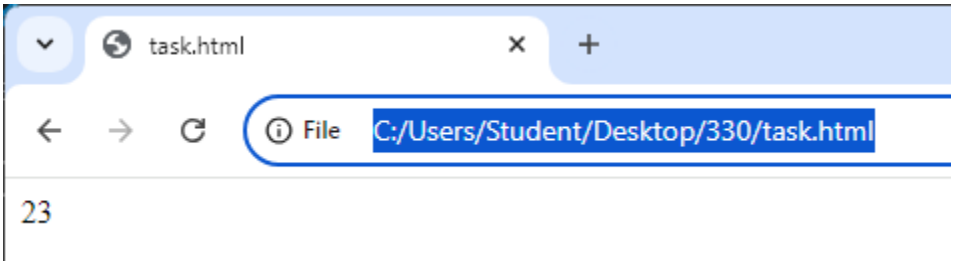
OUTPUT:



TASK 50:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
         let result = function( x=45, y=56)
         {
            return  y-x;
         }
         document.writeln(result(34,57));
    </script>
  </body>
</html>
```

OUTPUT:

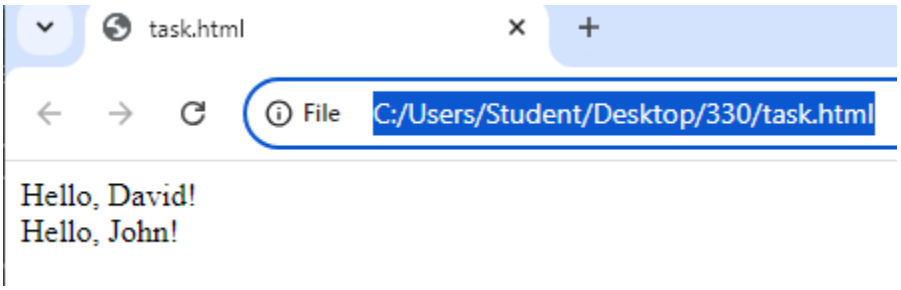


TASK 51:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
         let greet = (name)=>
         {
            return ("Hello, " + name + "!" + "<br>");
         }
         document.writeln(greet("David"));

         let greet1 = (name)=>
         {
            return ("Hello, " + name + "!" + "<br>");
         }
         document.writeln(greet1("John"));
    </script>
  </body>
</html>
```

OUTPUT:

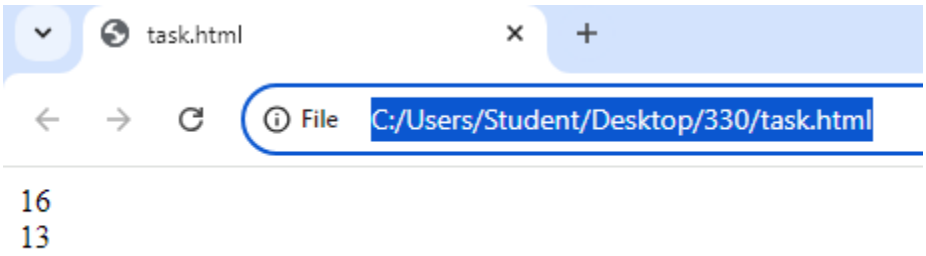


TASK 52:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
       let num1 =9,num2 = 7;
        let sum  = (num1,num2)=>
        {
           return (num1+num2);
        }
        document.writeln(sum(num1,num2) +"<br>");
        let num =5,num3 = 8;
        let sum1  = (num,num3)=>
        {
           return (num+num3);
        }
        document.writeln(sum(num,num3) +"<br>");

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

OUTPUT:

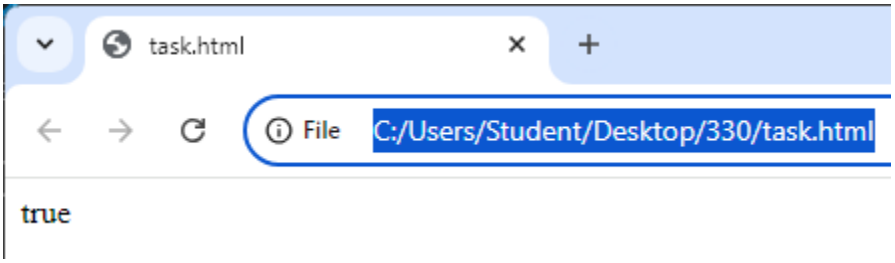


TASK 53:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
```

```
    <script>
       let isEven =(a) =>
       {
        if(a%2==0)
       {
        return true;
       }
       else
       {
       return false;
       }
       }
    document.writeln(isEven(4) +"<br>");
    </script>
  </body>
</html>
```

OUTPUT:

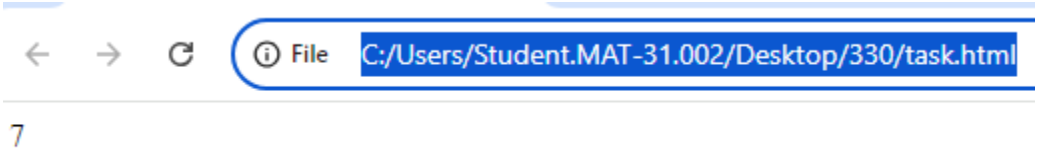


TASK 54:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
       let Maxvalue =(a,b) =>
       {
        if(a>b)
       {
        return a;
       }
       else
```

```
        {
        return b;
        }
        }
    document.writeln(Maxvalue( 7,3) +"<br>");
    </script>
  </body>
</html>
```

OUTPUT:



TASK 55:

```
<!DOCTYPE html>
<html>
  <head>
<meta charset ="UTF-8">
<meta name : "viewport" content="width+device_width,initial-scale=1.0">
  </head>
  <body>
    <script>
       let myObject = {
         value :23,
         multiplyTradition: function(num){
            return this.value*num;
         },
         multiplyArrow:(num)=>{
            return this.value*num;
         },
        };
    console.log(myObject.multiplyTradition(6));
   console.log(myObject.multiplyArrow(6));
    </script>
  </body>
</html>
```

OUTPUT: