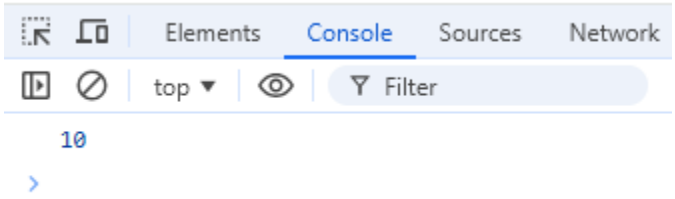
The modern mode, “use strict”, Variables

1. **The modern mode, “use strict”:**

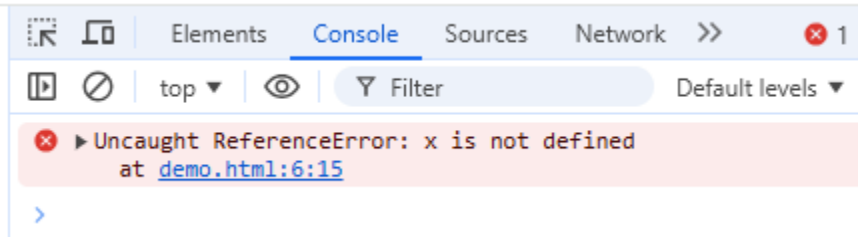
TASK 11

```
<html>
  <head></head>
  <body>
    <script>
      myVar = 10;
      console.log(myVar);
    </script>
  </body>
</html>
```



TASK 12

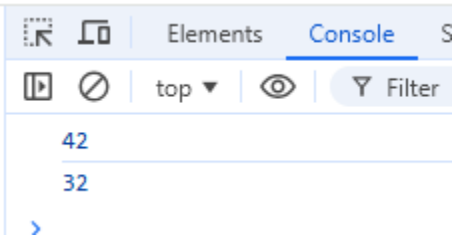
```
<html>
  <head></head>
  <body>
    <script>
      'use strict';
      x = 10;
      console.log(x);
    </script>
  </body>
</html>
```



TASK 14

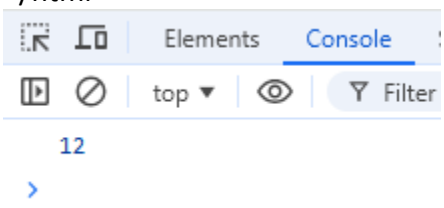
```
<html>
  <head></head>
```

```
  <body>
    <script>
      myVar = 42;
      console.log(myVar);
      'use strict';
      myVar = 32;
      console.log(myVar);
    </script>
  </body>
</html>
```



TASK 15

```
<html>
  <head></head>
  <body>
    <script>
      'use strict';
      var myVar = 12;
      console.log(myVar);
    </script>
  </body>
</html>
```



Data types, Basic operators, maths

1. **Data types:**

TASK 21

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>JavaScript Data Types and Scope Example</title>
</head>
<body>
  <script>
    let x = "Hello      World";
    let y = 42;
    let z = true;
    let p = null;
    let e;
    let myObject = { name: "Alice", age: 30 };
  </script>
</body>
</html>
```

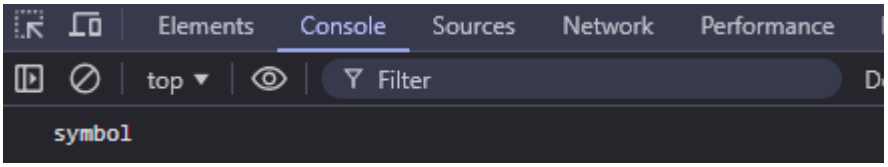
TASK 22

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>JavaScript Data Types and Scope Example</title>
</head>
<body>
  <script>
    let x = "Hello      World";
    let y = 42;
    let z = true;
    let p = null;
    let e;
    let myObject = { name: "Alice", age: 30 };
    console.log(typeof x);
    console.log(typeof y);
    console.log(typeof z);
    console.log(typeof p);
    console.log(typeof e);
  </script>
</body>
```

</html>

TASK 23

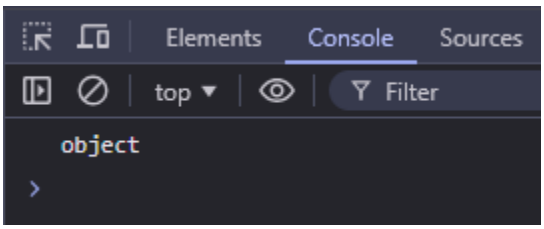
```
    <!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>JavaScript Data Types and Scope Example</title>
</head>
<body>
  <script>
    let demo = Symbol('my');
    console.log(typeof demo);
  </script>
</body>
</html>
```



TASK 24

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>JavaScript Data Types and Scope Example</title>
</head>
<body>
  <script>
    let demo = null;
    console.log(typeof demo);
  </script>
```

</body>
</html>



TASK 25

Var:

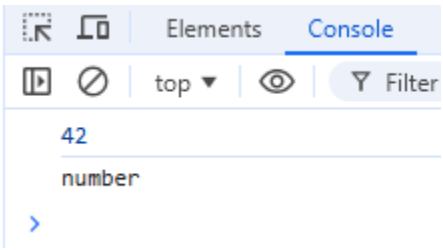
A variable declared with is scoped to the **entire function** where it is declared

Let:

A variable declared with is scoped to the nearest **block**

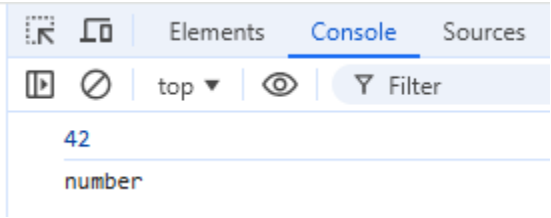
TASK 26

```
<html>
  <head></head>
  <body>
    <script>
      let str = "42";
      let num = str * 1;
      console.log(num);
      console.log(typeof num);
    </script>
  </body>
</html>
```



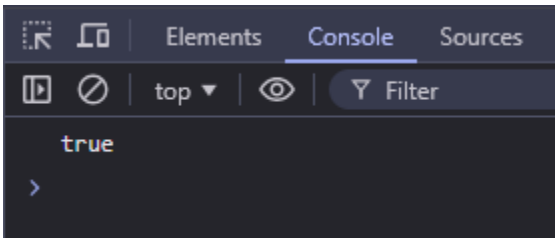

```
<html>
  <head></head>
  <body>
    <script>
      let str = "42";
      let num = Number(str);
      console.log(num);
      console.log(typeof num);
```

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



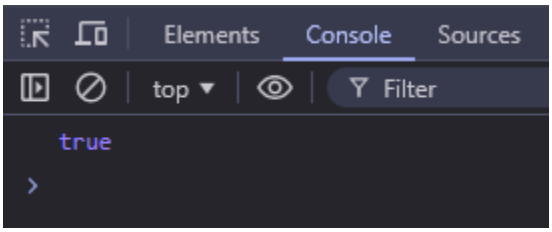
TASK 27

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>JavaScript Data Types and Scope Example</title>
</head>
<body>
  <script>
    let bool = true;
    let string = bool.toString();
    console.log(string);
  </script>
</body>
</html>
```



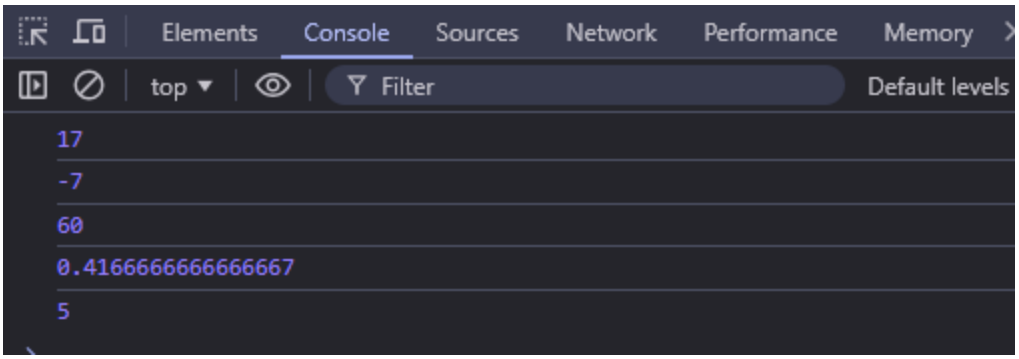

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>JavaScript Data Types and Scope Example</title>
```

```
</head>
<body>
  <script>
    let myString = "true";
    let myBool = Boolean(myString);
    console.log(myBool);

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



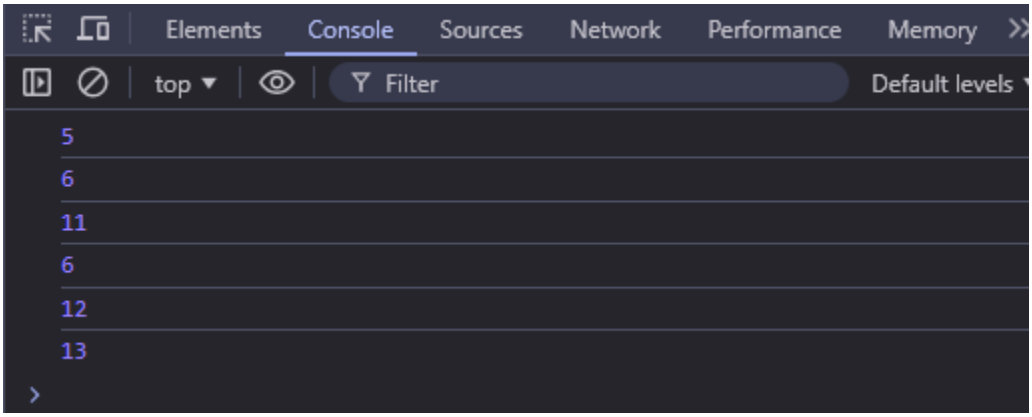
TASK 28

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>JavaScript Data Types and Scope Example</title>
</head>
<body>
  <script>
    var a = 5;
    var b = 12;
    console.log(a+b);
    console.log(a-b);
    console.log(a*b);
    console.log(a/b);
    console.log(a%b);
  </script>
</body>
</html>
```



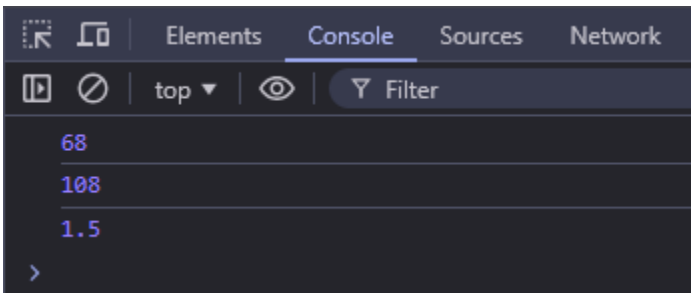
TASK 29

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>JavaScript Data Types and Scope Example</title>
</head>
<body>
   <script>
      var a = 5;
      var b = 12;
      console.log(a++);
      console.log(a);
      console.log(--b);
      console.log(a--);
      console.log(++b);
      console.log(++b);

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



TASK 30

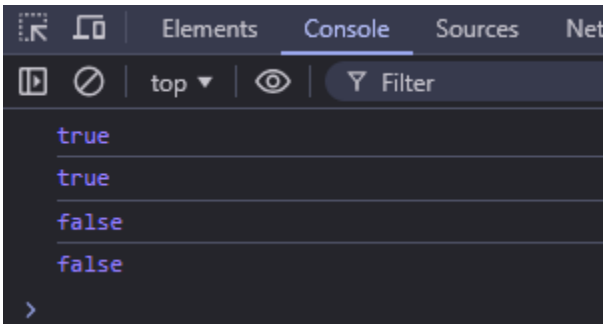
```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>JavaScript Data Types and Scope Example</title>
</head>
<body>
   <script>
      let result = 5 + 7 * 9;
      console.log(result);
      let res = (5+7)*9;
      console.log(res);
      let res1 = 7/14*3;
      console.log(res1);
   </script>
</body>
</html>
```



TASK 31

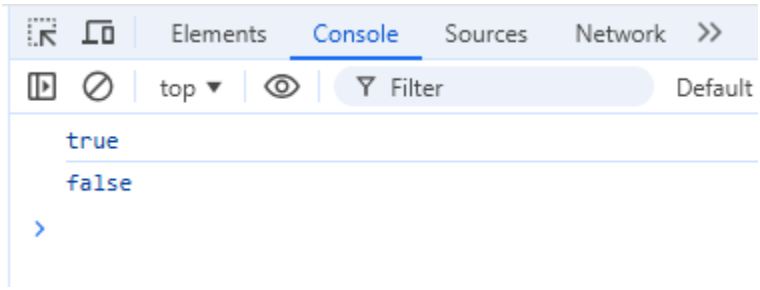
```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>JavaScript Data Types and Scope Example</title>
</head>
<body>
   <script>
      let a = 20;
      let b = 12;
      console.log(a>b);
      console.log(a>=b);
      console.log(a<=b);
```

```
    console.log(a<b);

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



TASK 32

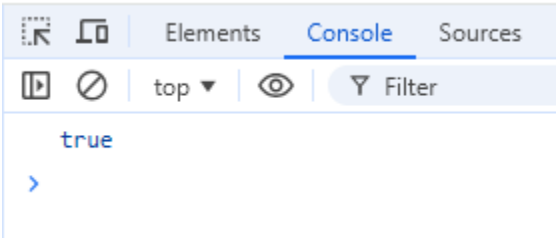
```
<html>
  <head>

  </head>
  <body>
    <script>
      let num1=12,num2="12";

      console.log(num1==num2);

      console.log(num1===num2);
    </script>
  </body>
</html>
```



TASK 33

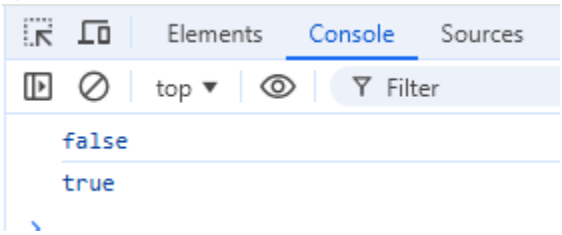
```
<html>
  <head></head>
  <body>
```

```
    <script>
      let s1="cookies";
      let s2="cOOkies";
      console.log(s1>s2);
    </script>
  </body>
</html>
```



TASK 34

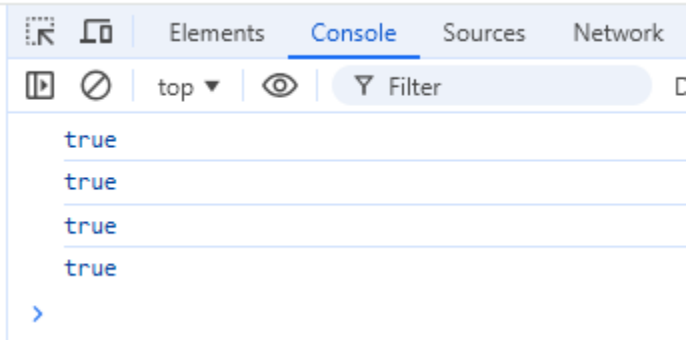
```
<html>
  <head></head>
  <body>
    <script>
      let n1=14;
      let n2="14";
      console.log(n1!=n2);
      console.log(n1!==n2);
    </script>
  </body>
</html>
```



TASK 35

```
<html>
  <head></head>
  <body>
    <script>
      let a=null;
      let b=null;
      let num1;
      let num2;
      console.log(a==b);
```

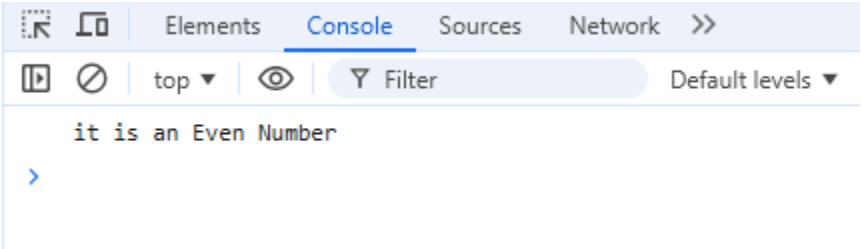
```
        console.log(a===b);
        console.log(num1==num2);
        console.log(num1===num2);
    </script>
  </body>
</html>
```



2. Conditional branching: if, '?':

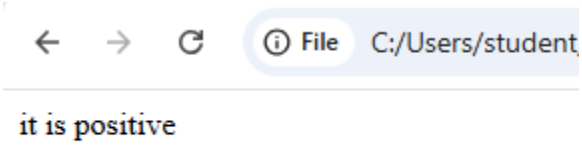
TASK 36

```
<html>
  <head></head>
  <body>
    <script>
      let n1=20;
      if(n1%2==0){
        console.log(" it is an Even Number");
      }
      else{
        console.log("it is a Odd Number");
      }
    </script>
  </body>
</html>
```



TASK 37

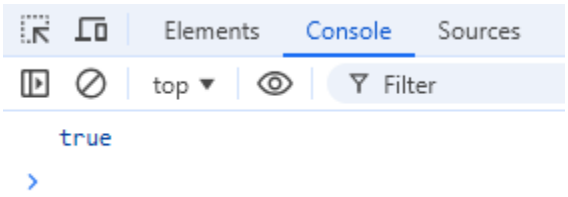
```
<html>
  <head></head>
  <body>
    <script>
      let n = 8;
      if(n==0){
        document.writeln("It is zero");
      }
      else if(n > 0){
        document.writeln("It is positive");
      }
      else{
        document.writeln("It is negetive");
      }
    </script>
  </body>
</html>
```



it is positive

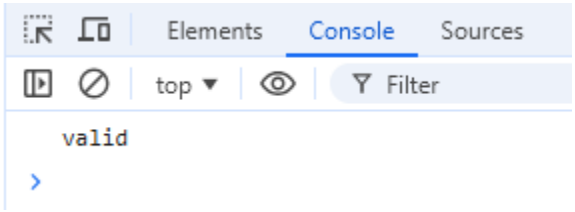
TASK 38

```
<html>
  <head></head>
  <body>
    <script>
      let age = 18;
      let vote = (age >= 18) ? true : false;
      console.log(vote);
    </script>
  </body>
</html>
```



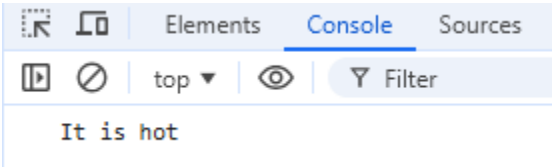
TASK 39

```
<html>
  <head></head>
  <body>
    <script>
      let age = 18;
      let vote = (age >= 18) ? “valid” : “invalid”;
      console.log(vote);
    </script>
  </body>
</html>
```



TASK 40

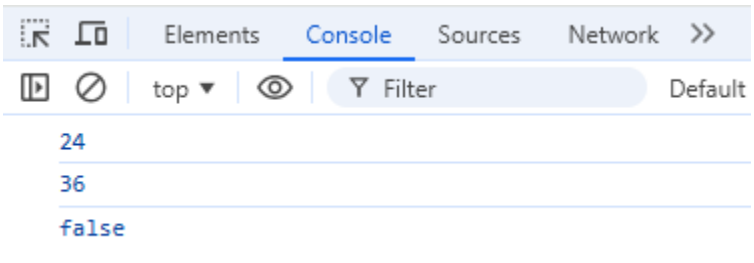
```
<html>
  <head></head>
  <body>
    <script>
      let temp = 32;
      let message = (temp > 25) ? "It is hot": "It is cold";
      console.log(message);
    </script>
  </body>
</html>
```



TASK 41

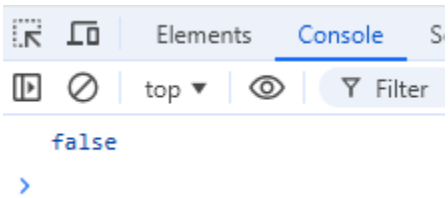
```
<html>
  <head></head>
  <body>
```

```
    <script>
      let n1= 36;
      let n2= 24;
      res1 = n1 && n2;
      res2 = n1 || n2;
      res3 = !n1;
      console.log(res1);
      console.log(res2);
      console.log(res3);
    </script>
  </body>
</html>
```



TASK 42

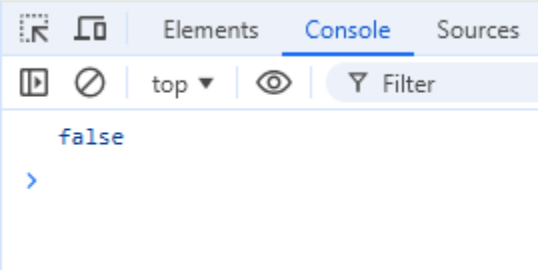
```
<html>
  <head></head>
  <body>
    <script>
      let x = 22;
      let min = 12;
      let max = 25;
      let res = (x > min && x < max);
      console.log(res);
    </script>
  </body>
</html>
```



TASK 43

```
<html>
```

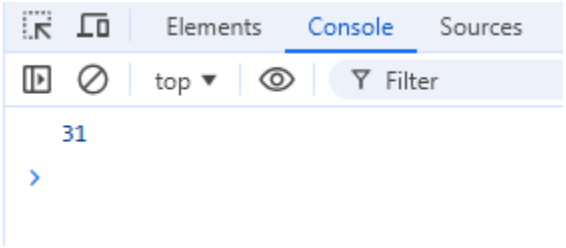
```
  <head></head>
  <body>
    <script>
      let n1 = true;
      console.log(!n1);
    </script>
  </body>
</html>
```



2. **Functions**:

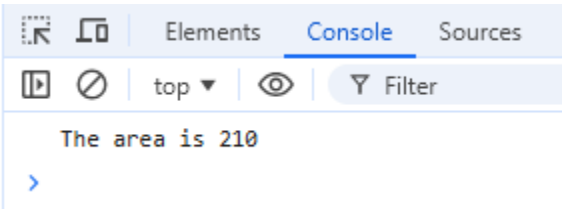
TASK 46

```
<html>
  <head></head>
  <body>
    <script>
    function calculate(a,b){
       res = a+b;
       return (res);
      }
      let a = 10;
      let b = 21;
      console.log(calculate(a,b));
    </script>
  </body>
</html>
```



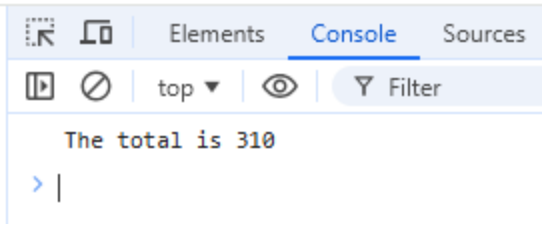
TASK 47

```
<html>
  <head></head>
  <body>
    <script>
    function calculate(w,b){
       return w*b;
      }
      let w = 10;
      let b = 21;
      console.log("The area is "+calculate(w,b));
    </script>
  </body>
</html>
```



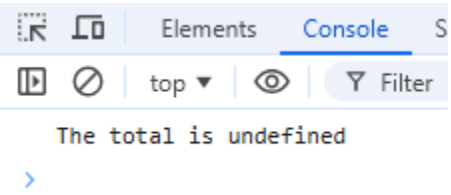
TASK 48

```
<html>
  <head></head>
  <body>
    <script>
    function calculate(){
       return jan+feb;
      }
      let jan = 210;
      let feb = 100;
      console.log("The total is "+calculate());
    </script>
  </body>
</html>
```



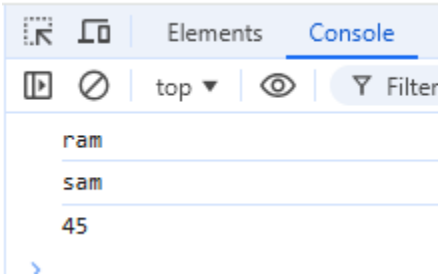
TASK 49

```
<html>
  <head></head>
  <body>
    <script>
    function calculate(){
       return;
      }
      let a = 210;
      let b = 100;
      console.log("The total is "+calculate());
    </script>
  </body>
</html>
```



TASK 50

```
<html>
  <head></head>
  <body>
    <script>
       let name="ram";
       let age = 21;
    function calculate(name ="ram",age = "21"){
       return name;
      }
    console.log(calculate());
    console.log(calculate(name="sam"));
    console.log(calculate(age = "45",name ));
    </script>
  </body>
</html>
```


Elements
Console
top
Filter
ram
sam
45

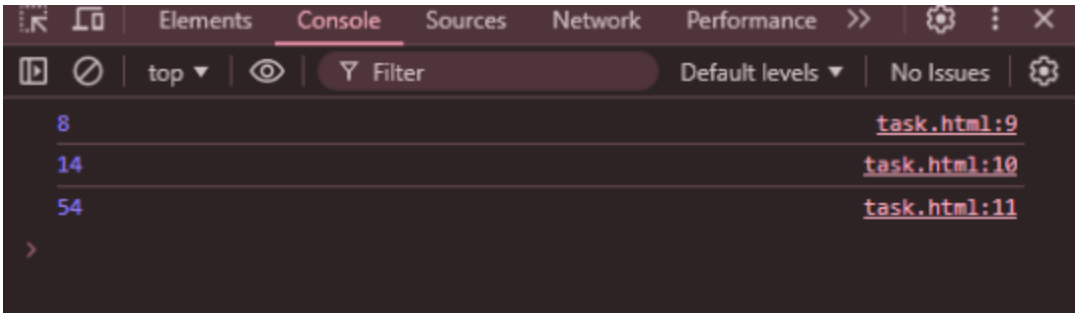
Arrow Functions:

TASK 51

```
<!DOCTYPE html>
<html lang "en">
    <head>

    </head>
    <body>
        <script>
            let greet =  name  => `Hello, ${name}`

            console log greet "Max"
            console log greet "John"


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

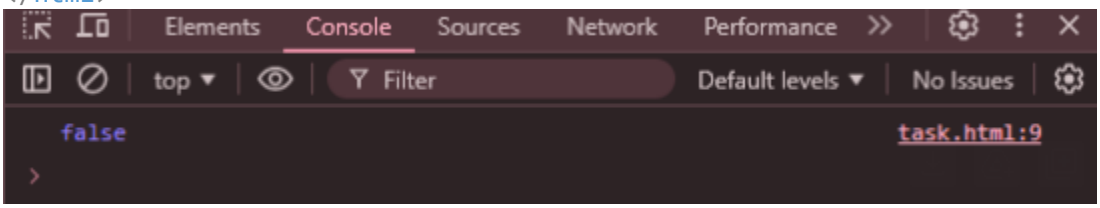
TASK 52

```
<!DOCTYPE html>
<html lang "en">
    <head>
    </head>
    <body>
        <script>
            let add =  a b  => a + b
            console log add 5 3
            console log add 6 8
            console log add 22 32

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



TASK 53

```
<!DOCTYPE html>
<html lang "en">
    <head>
        <title>             </title>
    </head>
    <body>
        <script>
            let isEven =  num  => num % 2 == 0
            console log isEven 65


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



TASK 54

```
<!DOCTYPE html>
<html lang "en">
    <head>
    </head>
    <body>
        <script>
            let maxValue =  num1 num2  =>
                return  num1 > num2 ?num1:num2

           console log maxValue 7 67
           console log maxValue 98 65
           console log maxValue 9 89

        </script>
    </body>
```

```
</html>
```


Elements
Console
Sources
Network
Performance
top
Filter
Default levels
No Issues
67
task.html:11
98
task.html:12
89
task.html:13