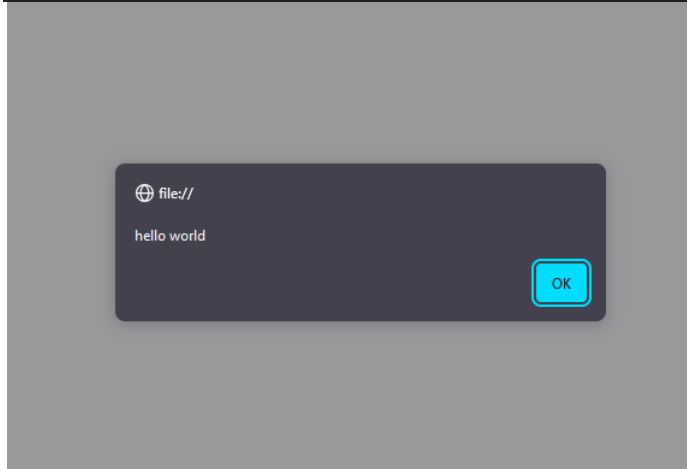


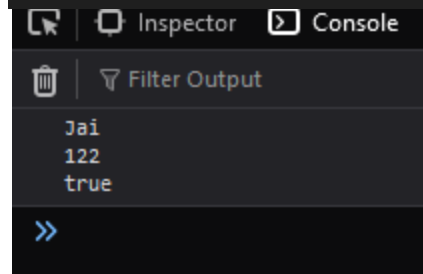
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document Title</title>
</head>
<body>
  <script>
    alert ("hello world");
  </script>
</body>
</html>
```



TASK 2

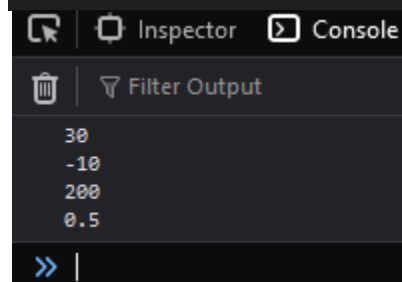
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document Title</title>
</head>
<body>
  <script>
    let name="Jai";
    let roll=122;
    let value=true;
    console.log(name+'\n'+roll+'\n'+value);
  </script>
```

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



TASK 3

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document Title</title>
</head>
<body>
  <script>
    let a=10;
    let b=20;
    console.log((a+b) +'\n'+ (a-b ) +'\n'+ a*b  +'\n'+ a/b);
  </script>
</body>
</html>
```



TASK 4

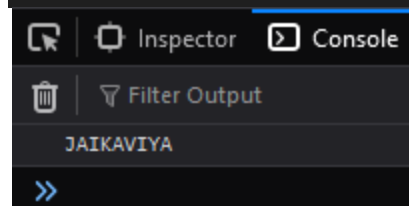
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```

    <title>Document Title</title>
</head>
<body>
    <script>
        let str1="JAI";
        let str2="KAVIYA";
        console.log(str1+str2);
    </script>

</body>
</html>

```



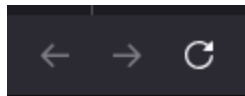
TASK 5

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document Title</title>
</head>
<body>
    <script>
        let name ="jai";
        document.writeln(typeof name +"<br>" );
        document.writeln("NAME "+name+"<br>");
        let name1=Number(name);
        document.writeln(typeof name1+"<br>");
        document.writeln("NAME1"+ name);
    </script>

</body>
</html>

```



```
string  
NAME jai  
number  
NAME1jai
```

2.CODE STRUCTURES

TASK 6

```
// single line comment  
  
/* this  
is  
a muliti line  
comment  
*/
```

SINGLE LINE COMMENT :

In Java Script the single line comment is used to comment the particular line which is marked with double backslach .. It is useful for the viewers of the code .

MULTILINE COMMENT :

In Java Script the multi line comment is used to comment the more than one line which is marked with/* ____ */.. It is useful for the viewers of the code .

TASK 7

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
  <meta charset="UTF-8">  
  <meta name="viewport" content="width=device-width, initial-scale=1.0">  
  <title>Document Title</title>  
</head>  
<body>  
  <script>  
    let d=1;  
    let e=2;  
    let name = "jai" //Error does not occurs because thje intepreter will do  
the automatic semicolon insertion  
    console.log(name);
```

```

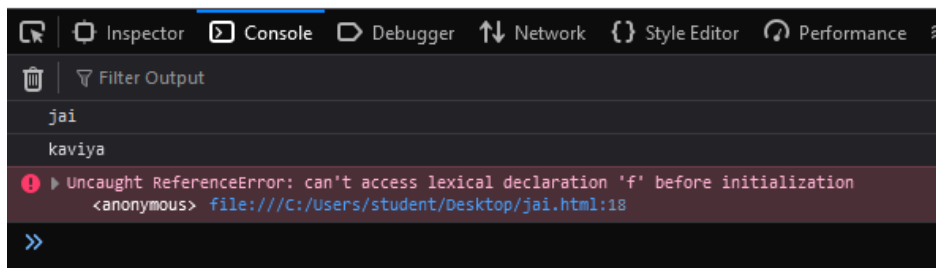
    let name1 = "kaviya";
    console.log(name1)

    let f = d + e    // In this case in rare time the error may occurs because
the interpreter thinks that the both are same line
    console.log(f)

</script>

</body>
</html>

```



Script with semicolon : It is the best way to practice or implement the java script code .

Script without semicolon : It may occurs some errors in the rare cases the interpreter thinks the console is a function of e.

TASK 8

```

<script>
    for(let i=2;i<=3;i++)
    {
        for(let j=1;j<=5;j++)
        {
            document.writeln(i +"x"+j +"="+(i*j)+"<br>");
        }
    }

</script>

```

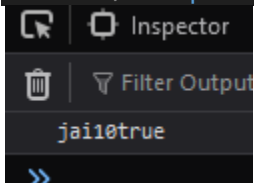


2x1=2
2x2=4
2x3=6
2x4=8
2x5=10
3x1=3
3x2=6
3x3=9
3x4=12
3x5=15

TASK 9

```
<script>
  let name ="jai" ,n=10, b=true
  console.log(name + n+ b);

</script>
```

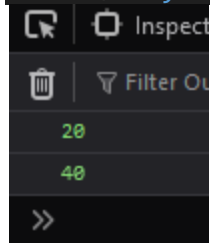


TASK 10

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document Title</title>
  <script>
    let a=10,b=10;
    console.log(a+b);

  </script>
</head>
<body>
  <script>
    let c=20,d=20;
    console.log(c+d);
```

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



The screenshot shows the bottom part of an HTML document with closing tags for script, body, and html. Below the code editor, the browser's developer tools are open, showing the 'Console' tab. It contains two log entries: the first is the number 20, and the second is the number 40. Navigation arrows are visible at the bottom of the console.

DIFFERENCE IN BEHAVIOUR :

If the script is inside the head the execution of the head will be first in the html document so the script executes first and gives the result as 20... Then the body will be execute and returns as 40.

TASK 11

```
<script>
  x=10;
  console.log(x);
</script>
```



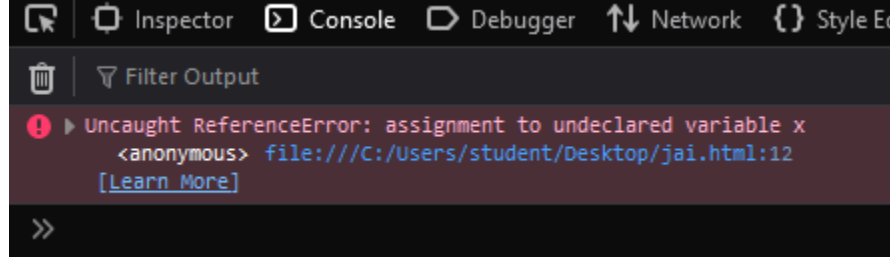
The screenshot shows an HTML document with a script tag in the head containing the code to set x=10 and log it. The developer tools console shows a single log entry with the value 10.

TASK 12

```
<SCRIPT>

"use strict";
x=10;
console.log(x);

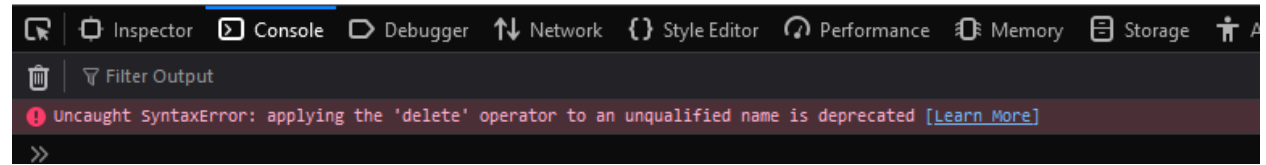
</script>
```



The screenshot shows an HTML document with a script tag in the head containing the code to set x=10 and log it, with "use strict" mode enabled. The developer tools console shows an error: "Uncaught ReferenceError: assignment to undeclared variable x" at file:///C:/Users/student/Desktop/jai.html:12. The error message is highlighted in red.

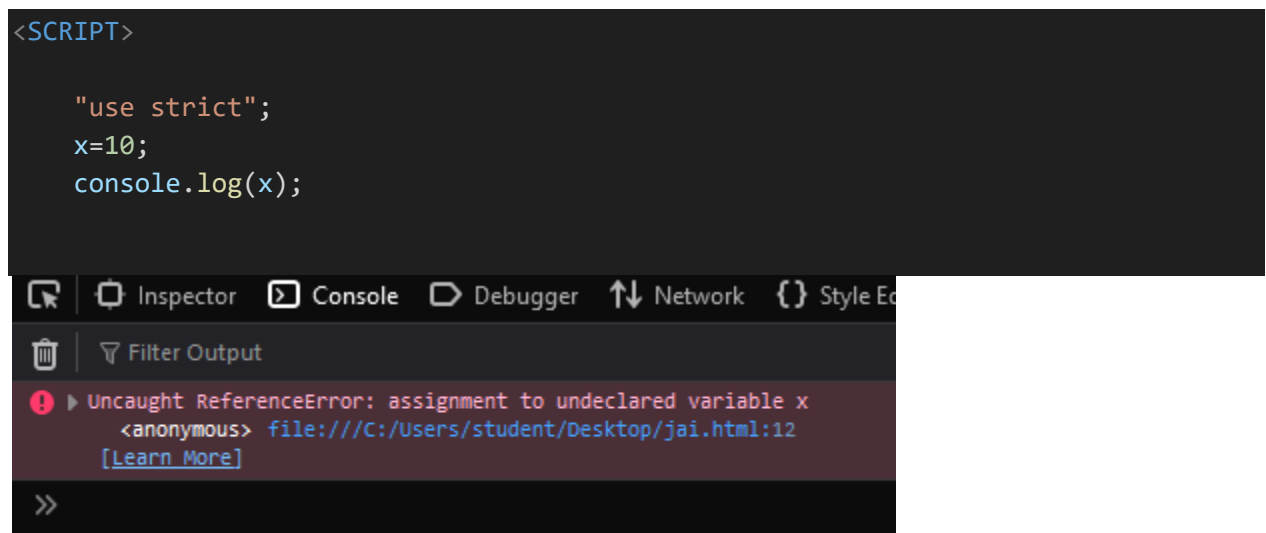
TASK 13

```
<script>"use strict";
    let myVar = 10;
    delete myVar;
</script>
```



Cannot able to delete

TASK 14



TASK 15

```
<SCRIPT>
```



```
"use strict";
let x=10;
console.log(x);

</script>
```



VARIABLES:

TASK 16

```
<script>
    let name="jai";
    var a="kaviya";
    const c=10;

</script>
```

LET : We cannot redeclare the variable we can only reassign the value

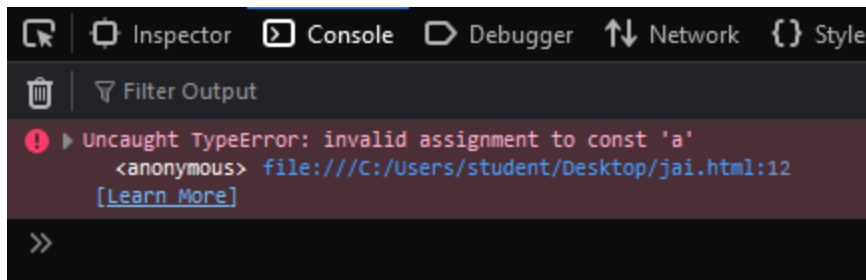
VAR : We can redeclare the variable we can reassign the value.

CONST:It is a constant value we cannot able to alter the value.

TASK 17

```
<script>
    const a=10;
    a=20;
    console.log(a);

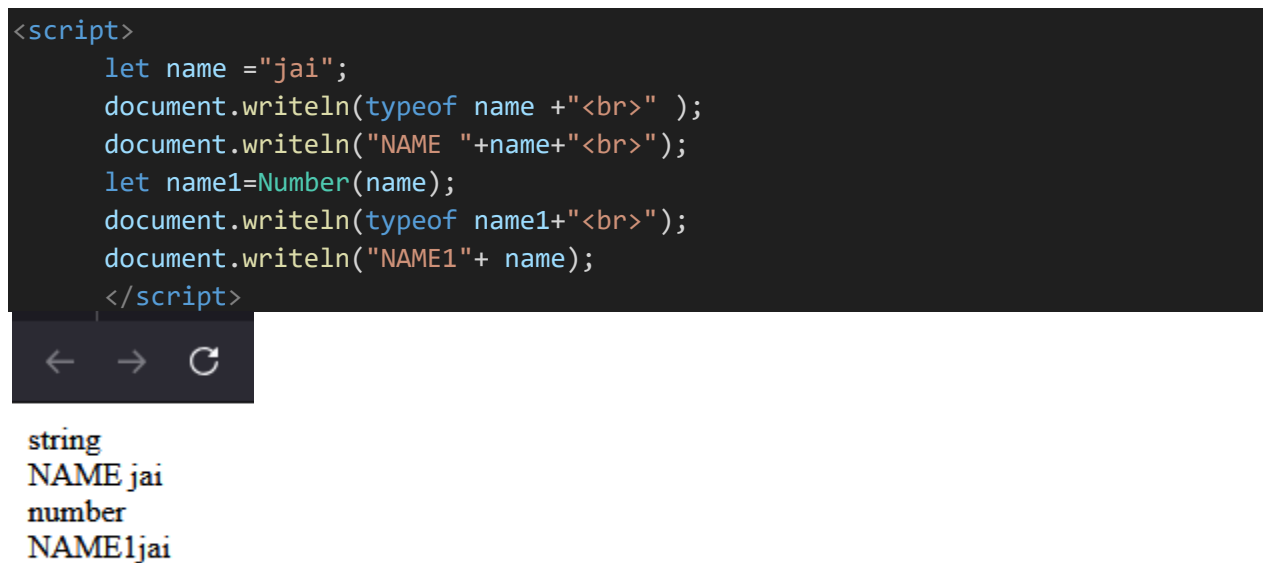
</script>
```



TASK 18



TASK 19



Task 20



```
console.log(userName);
```

```
let name = "Alice";  
console.log(name);
```

```
</script>
```

Inspector Console

Filter Output

Alice

Alice