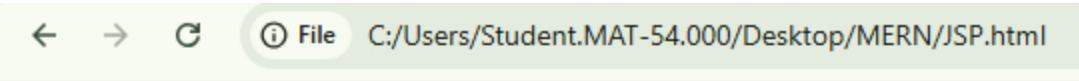
4.PROMISES CHAINING
TASK-5

```
<!DOCTYPE html>
 <html>
    <head>
     <body>
        <script>
function placeOrder(order)
 {
   return new Promise((resolve) => {
       setTimeout(() => {
         document.write(`${order} Order Placed`);
         document.write("<br>");
         resolve(order);
    },5000);
   })
 }
 function prepareFood(order)
{
   return new Promise((resolve) => {
      setTimeout(() => {
         document.write(`${order} is prepared`);
         document.write("<br>");
         resolve(order);
      },5000);
   })
}
function deliverFood(order)
{
   return new Promise((resolve) => {
      setTimeout(() => {
         document.write(`${order} is delivered`);
         document.write("<br>");
         resolve("order completed");
      });
   })
}

async function orderFood(foodItem)
{
   let order = await placeOrder(foodItem);
   let prepare = await prepareFood(order);
   let deliver = await deliverFood(prepare);

}
orderFood('pizza');
```

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

OUTPUT:



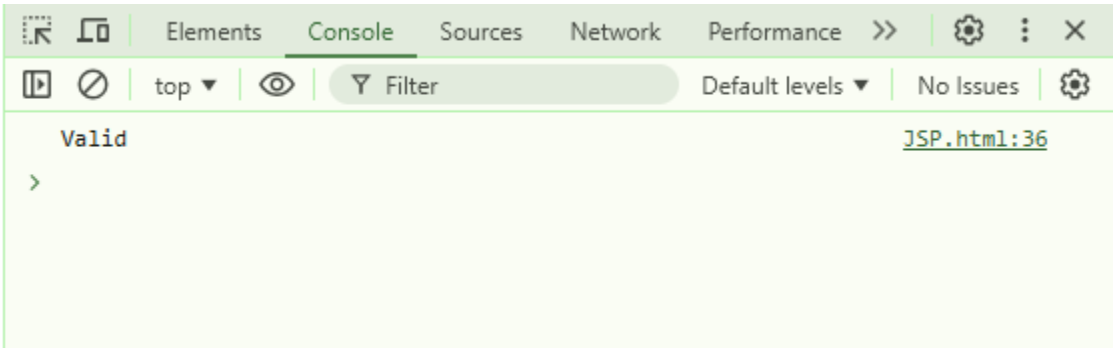
pizza Order Placed
pizza is prepared
pizza is delivered

5.ASYNC/WAIT
TASK-1

```
<!DOCTYPE html>
<html>
  <head>
   <body>
     <script>

   async function fetchData(url)
   {
     if(url == "https://api.com")
   {
     return ("Valid");
   }
   else{
     throw new Error("Not valid");
   }
   }
   async function getData()
   {
     try{
     const result = await fetchData("https://api.com");
     console.log(result);
     }
     catch (error) {
       console.error(error.message);
     }
   }
   getData();
     </script>
   </body>
```

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

OUTPUT:



1.RECURSION AND STACK
TASK-1

```
<!DOCTYPE html>
 <html>
    <head>
     <body>
       <script>

     function factorial(n)
     {
       if(n === 0)
     {
       return 1;
     }
     else{
       return n * factorial(n - 1);
     }
     }
     console.log(factorial(6));
       </script>
     </body>
  </head>
</html>
```

OUTPUT:

TASK-2

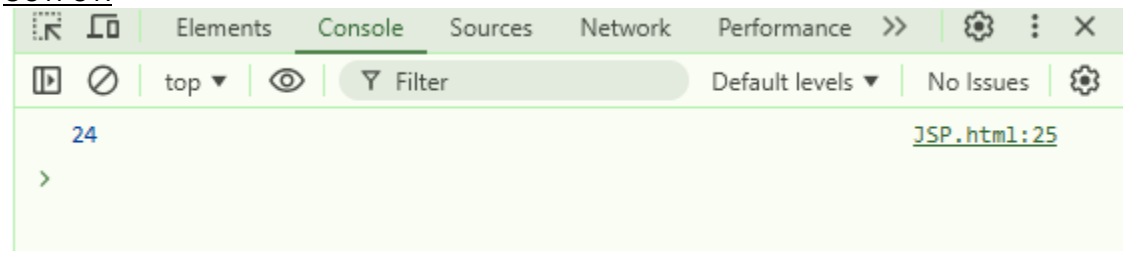
```
<!DOCTYPE html>
<html>
  <head>
    <body>
      <script>
    function fibonacci(n)
    {
      if(n === 0)
    {
      return 0;
    }
    else if(n === 1){
      return 1;
    }
    else
    {
      return fibonacci(n - 1) + fibonacci(n - 2);
    }
    }
    console.log(fibonacci(6));
      </script>
    </body>
  </head>
</html>
```

OUTPUT:

TASK-3

```
<!DOCTYPE html>
 <html>
   <head>
    <body>
      <script>
    function climbStairs(n)
    {
      if(n === 0)
    {
      return 1;
    }
    else if(n === 1){
      return 1;
    }
    else if(n === 2)
    {
      return 2;
    }
    else{
      return climbStairs(n - 1) +climbStairs(n - 2) + climbStairs(n - 3);
    }
    }
    console.log(climbStairs(6));
      </script>
    </body>
  </head>
</html>
```

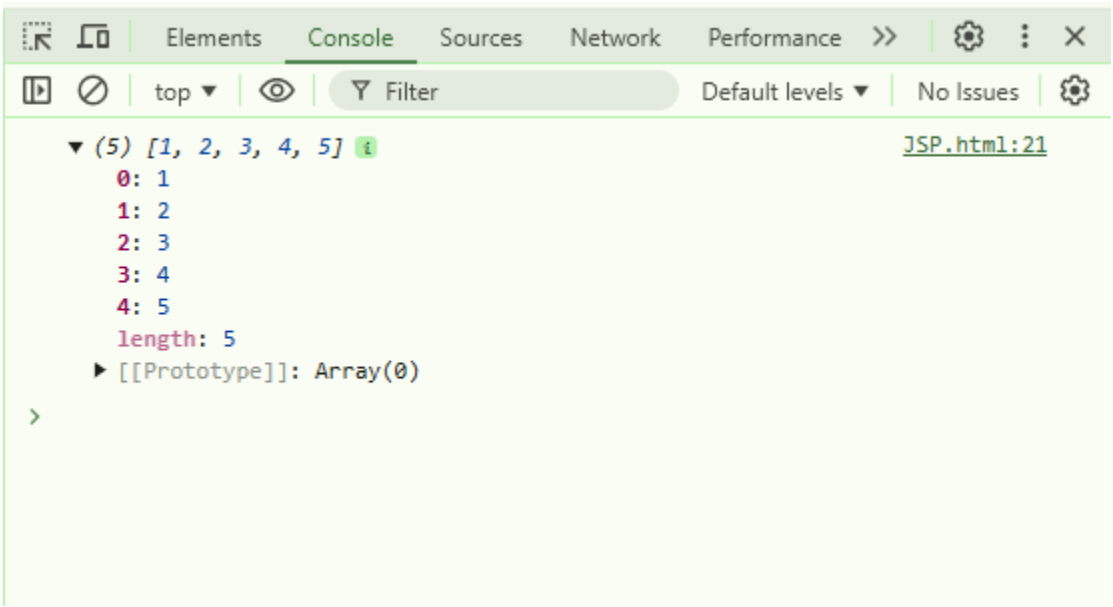
OUTPUT:



TASK-4

```
<!DOCTYPE html>
 <html>
    <head>
      <body>
         <script>

    function nestedArray(arr) {
   let result = [];

   arr.forEach(item => {
      if (Array.isArray(item)) {
         result = result.concat(nestedArray(item));
      } else {
         result.push(item);
      }
   });

   return result;
}

console.log(nestedArray([1, [2, [3, [4]], 5]]));

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

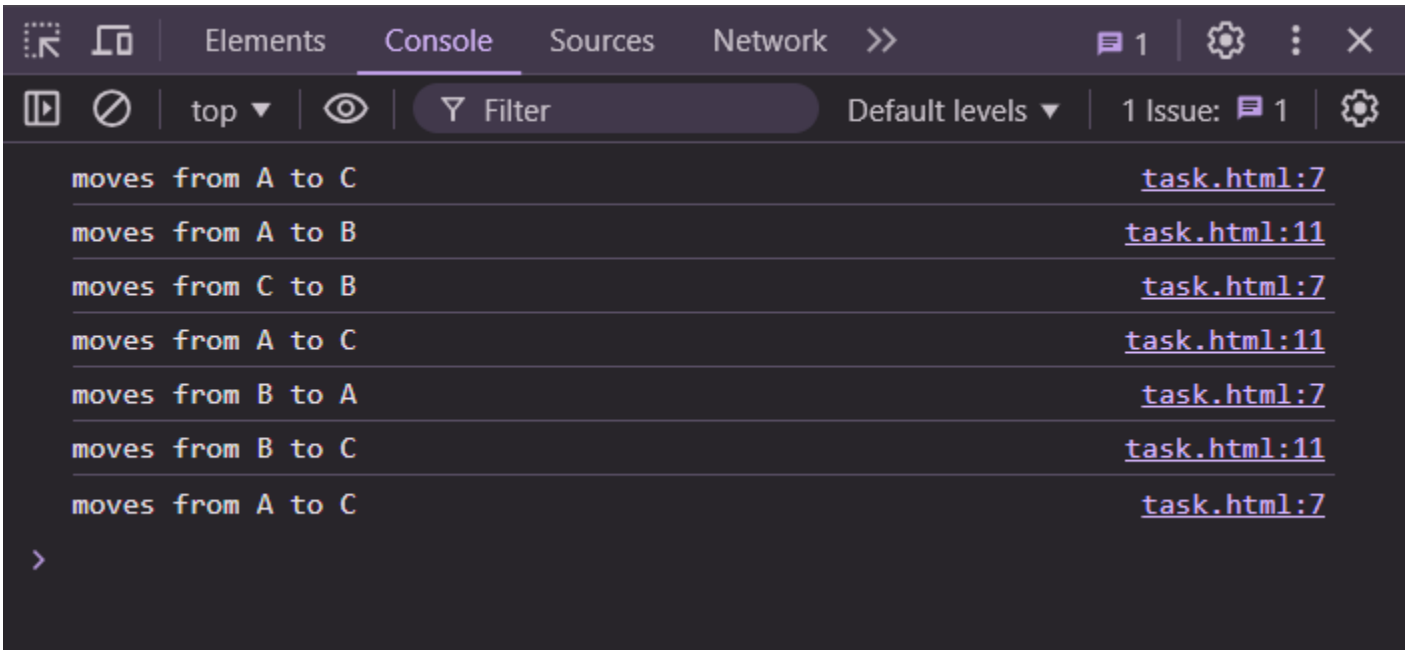
OUTPUT:



TASK-5

```
<html>
 <body>
  <script>
     function towerofHanoi(n,source,destination,auxillary)
     {
        if(n === 1){
           console.log(`moves from ${source} to ${destination}`);
           return;
        }
        towerofHanoi(n - 1,source,auxillary,destination);
        console.log(`moves from ${source} to ${destination}`);

        towerofHanoi(n - 1,auxillary,destination,source);
     }
     const n = 3;
        towerofHanoi(n,'A','C','B');
  </script>
 </body>
</html>
```

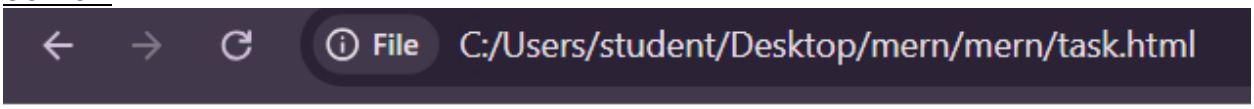
OUTPUT:



7.BROWSER:DOM BASICS
TASK-1

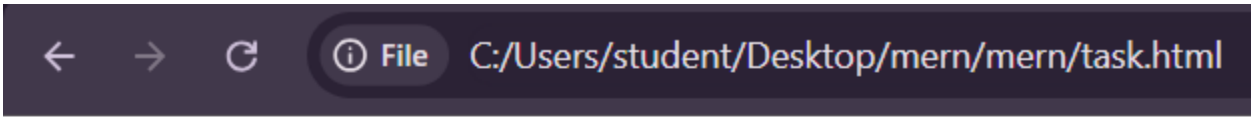
```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Document</title>
</head>
<body>
   <h1 id="header">WELCOME TO KCE</h1>
   <button onclick="changecontent()">Change</button>
   <script>
      function changecontent(){
         document.getElementById("header").textContent = "HAPPY TO SEE YOU ALL";
      }
   </script>
```

OUTPUT:



WELCOME TO KCE

Change



HAPPY TO SEE YOU ALL

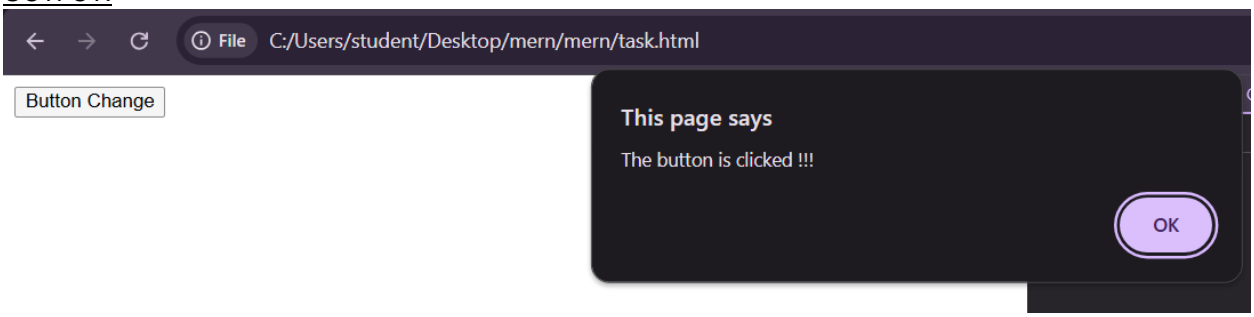
Change

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>
</head>
<body>
    <button  id="clickbutton">Button Change</button>
    <script>
        document.getElementById("clickbutton").addEventListener("click",function(){
            alert("The button is clicked !!!")
        });
    </script>
</body>
</html>
```

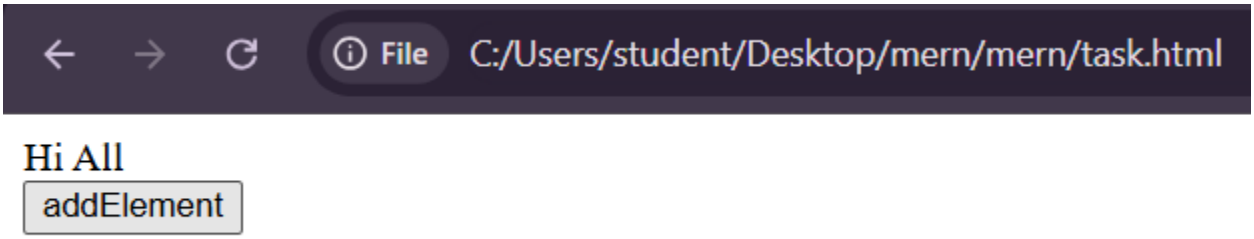
OUTPUT:

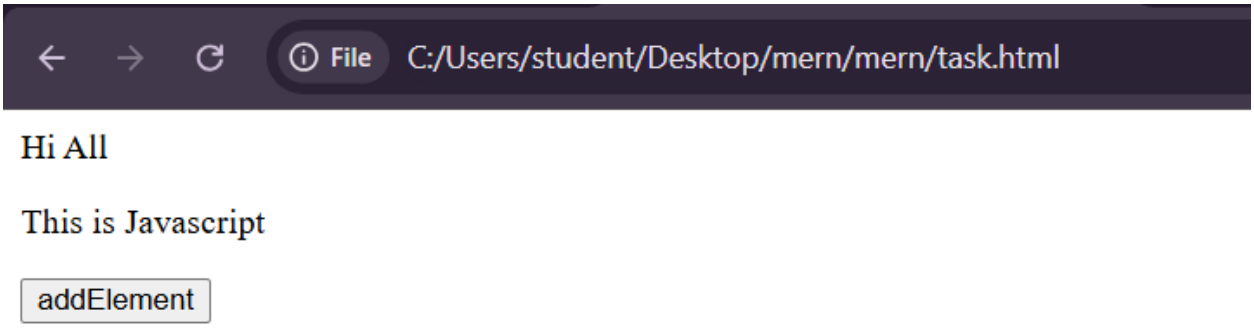


TASK-3

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <div id="container">Hi All</div>
    <button onclick="changeElement()">addElement</button>
    <script>
        function changeElement(){
            let newelement = document.createElement("p");
            newelement.textContent = "This is Javascript";

            document.getElementById("container").appendChild(newelement);
        }
    </script>
</body>
</html>
```

OUTPUT:

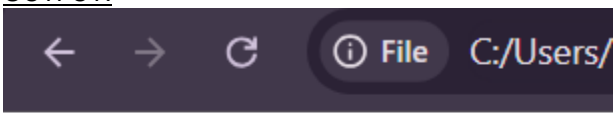




TASK-4

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Task 4</title>
</head>
<body>
  <div id="toggleElement">This is a toggled element!</div>
  <button onclick="toggleVisibility()">Toggle Visibility</button>

  <script>
    function toggleVisibility() {
      let element = document.getElementById("toggleElement");

      if (element.style.display === "none") {
       document.write("It is Visible")  ;
      } else {
        document.write("It is not Visible")  ;
      }
    }
  </script>
</body>
```

OUTPUT:



This is a toggled element!

Toggle Visibility

It is not Visible