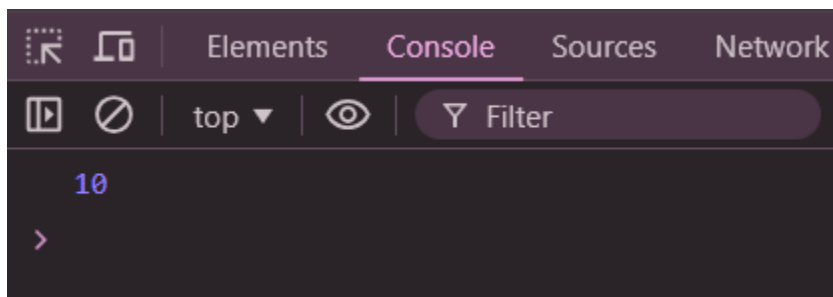


JSON and variable length arguments/spread syntax

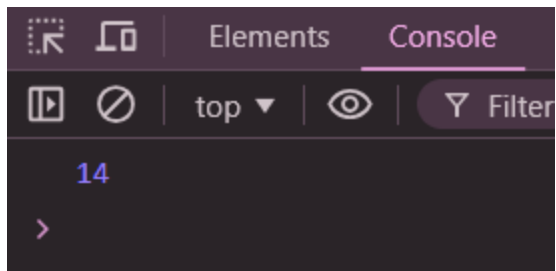
TASK 1

```
<html>
<body>
  <script>
    function sum(...args) {
      let total = 0;
      for (let num of args) {
        total += num;
      }
      return total;
    }
    console.log(sum(1, 2, 3, 4));
  </script>
</body>
</html>
```



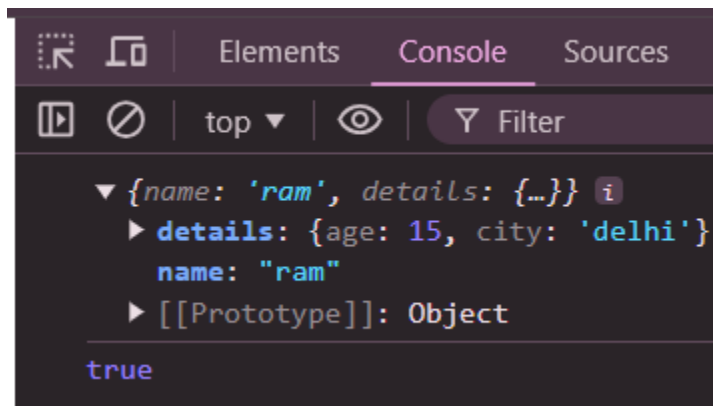
TASK 2

```
<html>
<body>
  <script>
    function sum(...args) {
      let total = 0;
      for (let num of args) {
        total += num;
      }
      return total;
    }
    const arr = [1,6,7];
    console.log(sum(...arr));
  </script>
</body>
</html>
```



TASK 3

```
<html>
<body>
  <script>
    function deepclone(obj){
      return JSON.parse(JSON.stringify(obj));
    }
    const originalobj = { name:"ram",details:{age: 15,city : "delhi"}};
    const clonedobj = deepclone (originalobj);
    console.log(clonedobj);
    console.log(originalobj!=clonedobj);
  </script>
</body>
</html>
```



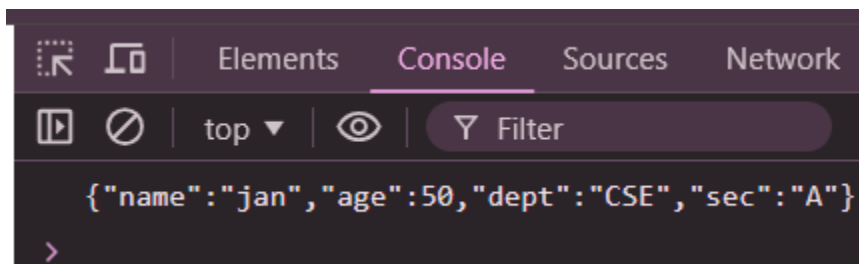
TASK 4

```
<html>
<body>
  <script>
    let student = {
      name : "ram",
      age : 20,
      dept: "CSE",
    };
  </script>
</body>
</html>
```

```

    let school = {
      name : "jan",
      age : 50,
      sec : "A",
    };
    let obj = {...student,...school};
    console.log(JSON.stringify(obj));
  </script>
</body>
</html>

```



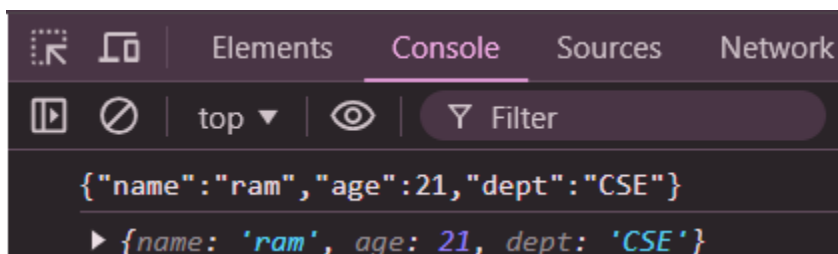
TASK 5

```

<html>
<body>
  <script>
    const obj = {name:"ram",age:21,dept:"CSE"};
    const jsonstring = JSON.stringify(obj);
    console.log(jsonstring);

    const parseint = JSON.parse(jsonstring);
    console.log(parseint);
  </script>
</body>
</html>

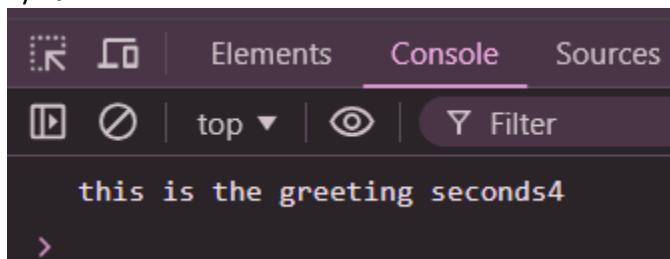
```



Promise, Promises chaining:

TASK 1

```
<html>
<body>
  <script>
    function greetdelay(sec){
      return new Promise((resolve)=>{
        setTimeout(() => {
          resolve("this is the greeting seconds" + sec);
        },sec*1000);
      });
    }
    greetdelay(4).then((greeting)=>console.log(greeting));
  </script>
</body>
</html>
```



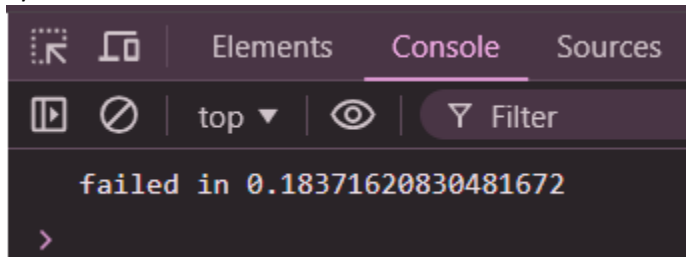
TASK 3

```
<html>
<body>
  <script>
    function randomPromise(){
      return new Promise((resolve,reject)=>{
        const randnum = Math.random();
        if(randnum > 1)
        {
          resolve("success in "+ randnum);
        }
        else if(randnum == 1){
          resolve("success again"+randnum);
        }
        else{
          reject("failed in "+randnum);
        }
      });
    }
  </script>
</body>
</html>
```

```

});
    }
    randomPromise()
    .then((message)=>console.log(message))
    .catch((error)=>console.log(error));
</script>
</body>
</html>

```



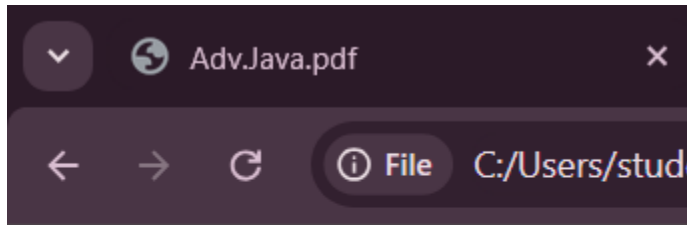
TASK 4

```

<html>
<body>
<script>
    function reachHome(name,timetoreach,issuccessful){
        return new Promise((resolve,reject)=>
        setTimeout(()=>{
            if(issuccessful){
                resolve(`${name} has reached home`);
            }
            else{
                reject(`${name} could not reached home`);
            }
        }),timetoreach)}
    let f1 = reachHome('ram',1000,true);
    let f2 = reachHome('jan',2000,true);
    let f3 = reachHome('feb',3000,false);
    Promise.allSettled([f1,f2,f3]).then
    ((Response)=>{
        Response.forEach((res)=>{
            document.writeln(res.status == 'fulfilled'?res.value:res.reason);
            document.writeln("<br>");
        })
    })
    .catch((error)=>{document.writeln(error.message)}).finally("party over");

```

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



ram has reached home
jan has reached home
feb could not reached home