

Task-26:

Code:

Tasks.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Tasks</title>
</head>
<body>
  <script type="module">
    import { greet, Person, today } from './Task.js';
    console.log(greet("alan"));
    var person=new Person("alan",20)
    console.log(person.details());
    console.log(today);

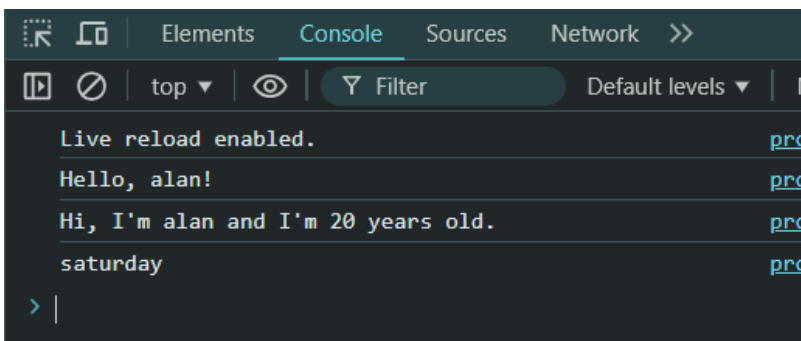
  </script>

</body>
</html>
```

Task.js:

```
export function greet(name) {
  return `Hello, ${name}!`;
}
export class Person {
  constructor(name, age) {
    this.name = name;
    this.age = age;
  }
  details() {
    return `Hi, I'm ${this.name} and I'm ${this.age} years old.`;
  }
}
export var today ="saturday";
```

Output:



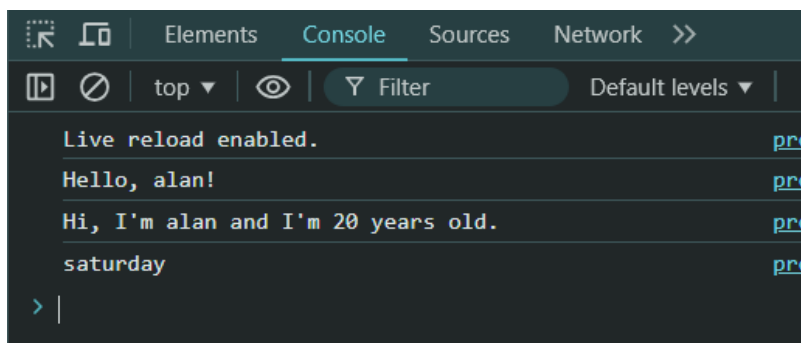
Task-27:

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Task</title>
</head>
<body>
  <script type="module">
    import { greet, Person, today } from './Task.js';
    console.log(greet("alan"));
    var person=new Person("alan",20)
    console.log(person.details());
    console.log(today);
  </script>
</body>
</html>
Task.js:
```

```
export function greet(name) {
  return `Hello, ${name}!`;
}
export class Person {
  constructor(name, age) {
    this.name = name;
    this.age = age;
  }
  details() {
    return `Hi, I'm ${this.name} and I'm ${this.age} years old.`;
  }
}
export var today ="saturday";
```

Output:



Task-28:

Code:

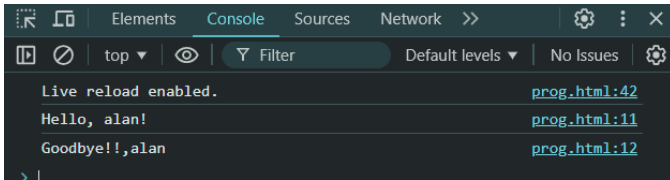
```
<!DOCTYPE html>
```

```
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Task</title>
</head>
<body>
  <script type="module">
    import { greet, sendoff } from './Task.js';
    console.log(greet("alan"));
    console.log(sendoff("alan"));
  </script>
</body>
</html>
```

Task.js:

```
function greet(name) {
  return `Hello, ${name}!`;
}
function sendoff(name){
  return `Goodbye!!,$ {name}`
}
export {greet,sendoff}
```

Output:



Task-29:

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Task</title>
</head>
<body>
  <script type="module">
    import { add, sub } from './Task.js';
    console.log(add(5,6));
    console.log(sub(5,6));
  </script>
</body>
</html>
```

Task.js:

```
function add(a,b) {
  return a+b;
}
function sub(a,b){
  return a-b;
}
```

```
}  
export {add,sub}
```

Output:

```
Live reload enabled.  
11  
-1  
> |
```

Task-30:

Code:

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
  <meta charset="UTF-8">  
  <meta name="viewport" content="width=device-width, initial-scale=1.0">  
  <title>Task</title>  
</head>  
<body>  
  <script type="module">  
    import mul,{sub} from './Task.js';  
    console.log(mul(5,6));  
    console.log(sub(5,6));  
  </script>  
</body>  
</html>
```

Task.js:

```
export default function add(a,b) {  
  return a+b;  
}  
export function sub(a,b){  
  return a-b;  
}
```

Output:

```
Live reload enabled.  
11  
-1  
>
```

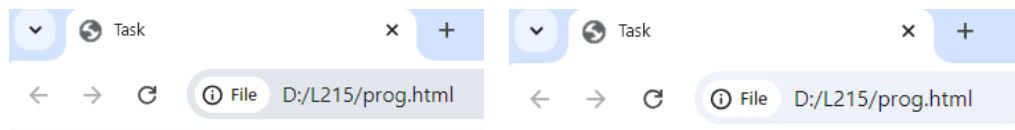
Task-31:

Code:

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

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Task</title>
</head>
<body>
  <h1 id="h" onmouseover="content()">Content</h1>
  <script>
    function content(){
      document.getElementById("h").innerHTML="Content changed";
    }
  </script>
</body>
</html>
```

Output:



Content

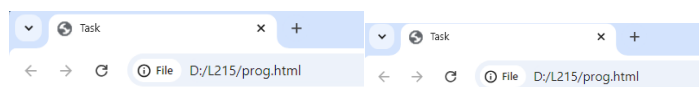
Content changed

Task-32:

Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Task</title>
</head>
<body>
  <h1 id="h">Content</h1>
  <button id="but">Change</button>
  <script>
    document.getElementById("but").addEventListener("click",content);
    function content(){
      document.getElementById("h").innerHTML="Content Changed"
    }
  </script>
</body>
</html>
```

Output:



Content

Content Changed

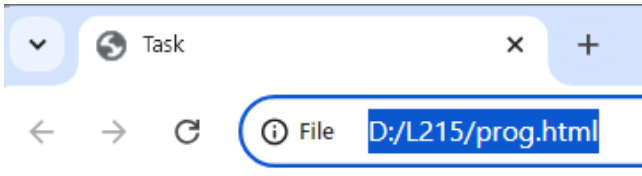
Change

Change

Task-33:**Code:**

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

  <script>
    var h= document.createElement("h1");
    var text=document.createTextNode("child append");
    h.appendChild(text);
    document.body.appendChild(h);
  </script>
</body>
</html>
```

Output:

child append

Task-34:**Code:**

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Task</title>
</head>
<body>
  
  <script>
    document.getElementById("image").addEventListener("mouseover",big)
    document.getElementById("image").addEventListener("mouseout",small)
    function big(){
      document.getElementById("image").style.height="700px";
      document.getElementById("image").style.width="700px";
    }
    function small(){
```

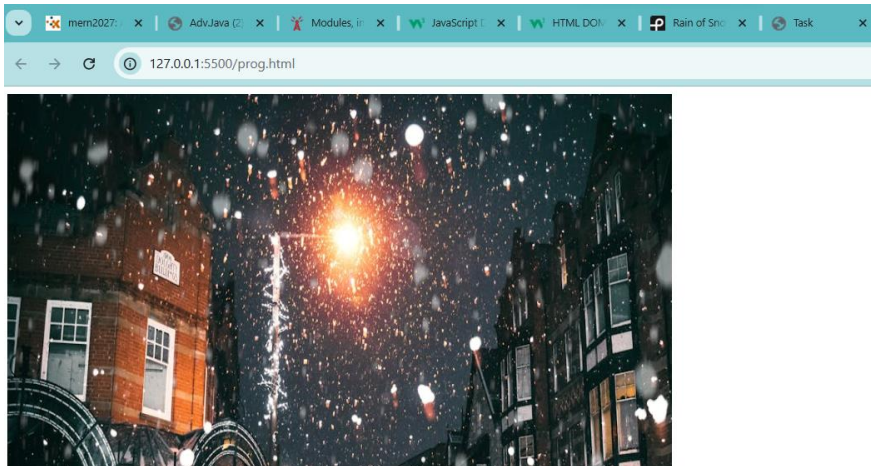
```

    document.getElementById("image").style.height="50px";
    document.getElementById("image").style.width="50px";
}

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

```

Output:



Task-35:

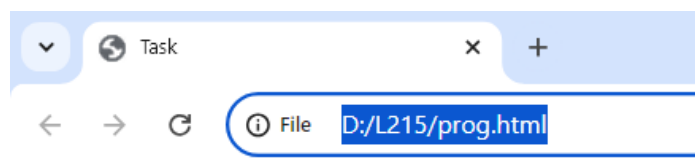
Code:

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Task</title>
</head>
<body>
  <p class="para" id="p-tag">Paragraph tag</p>
  <script>
    var ele=document.getElementById("p-tag");
    var class_name=ele.getAttribute('class');
    var id_name=ele.getAttribute('id');
    document.writeln("before modificaton:"+"<BR>");
    document.writeln('class'+<BR>");
    document.writeln('id'+<BR>")
    document.writeln(" After modificaton:"+"<BR>");
    ele.setAttribute('class','new-clss');
    ele.setAttribute('id','new-id');
    document.writeln(ele.getAttribute('class')+"<BR>");
    document.writeln(ele.getAttribute('id')+"<BR>")
  </script>
</body>
</html>

```

Output:



Paragraph tag

before modificaton:

class

id

After modificaton:

new-clss

new-id