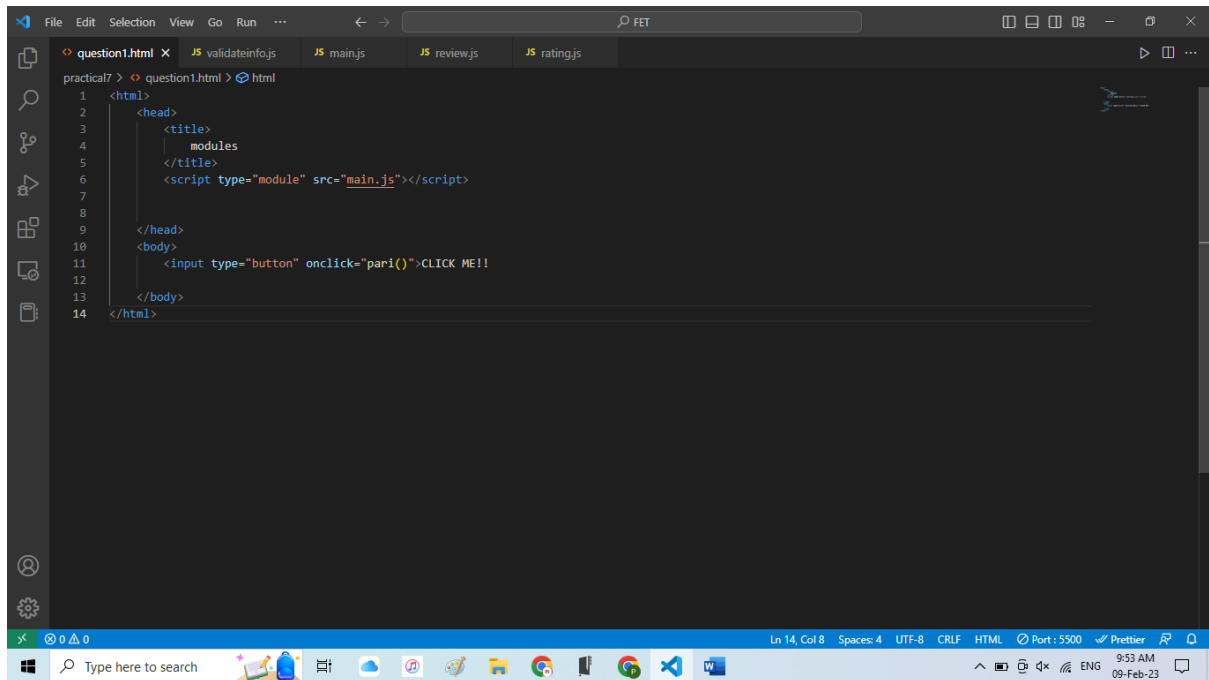


FET PRACTICAL-7

Objective: To understand the usage of modules in EcmaScript.

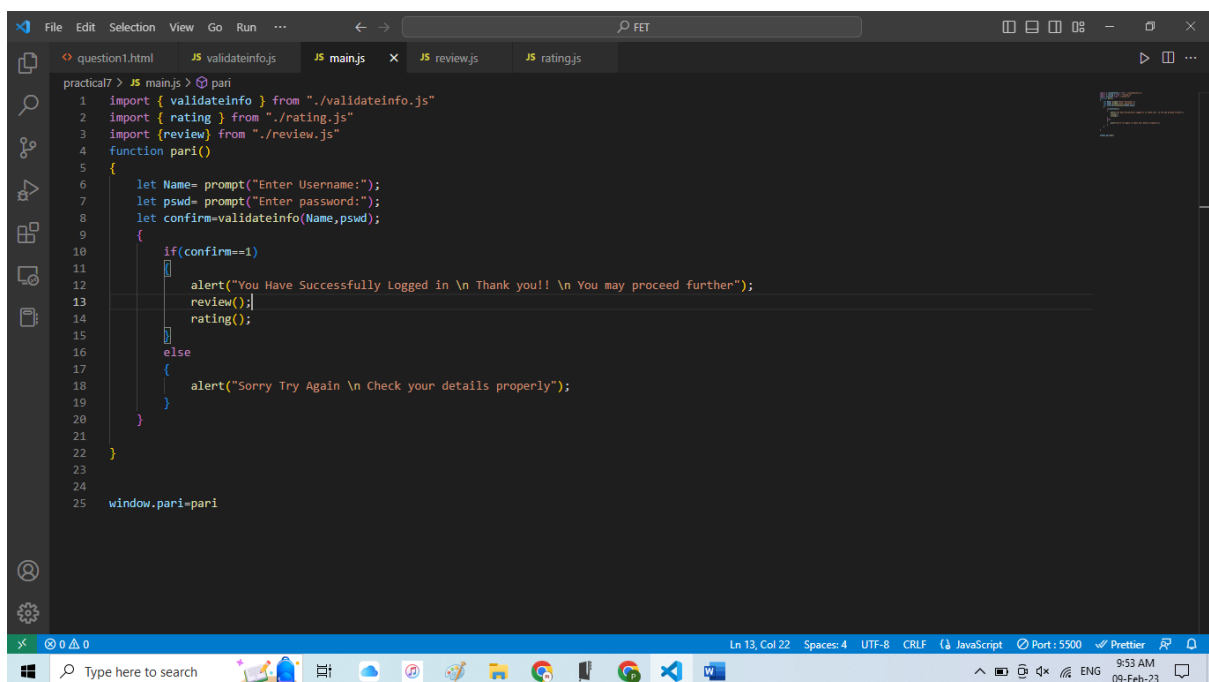


The screenshot shows the FET IDE interface with a file named 'question1.html' open. The code is as follows:

```
1 <html>
2   <head>
3     <title>
4       modules
5     </title>
6     <script type="module" src="main.js"></script>
7   </head>
8   <body>
9     <input type="button" onclick="pari()">CLICK ME!!
10  </body>
11 </html>
```

The status bar at the bottom indicates 'Ln 14, Col 8', 'Spaces: 4', 'UTF-8', 'CRLF', 'HTML', 'Port: 5500', and 'Prettier'.

MAIN FILE:-

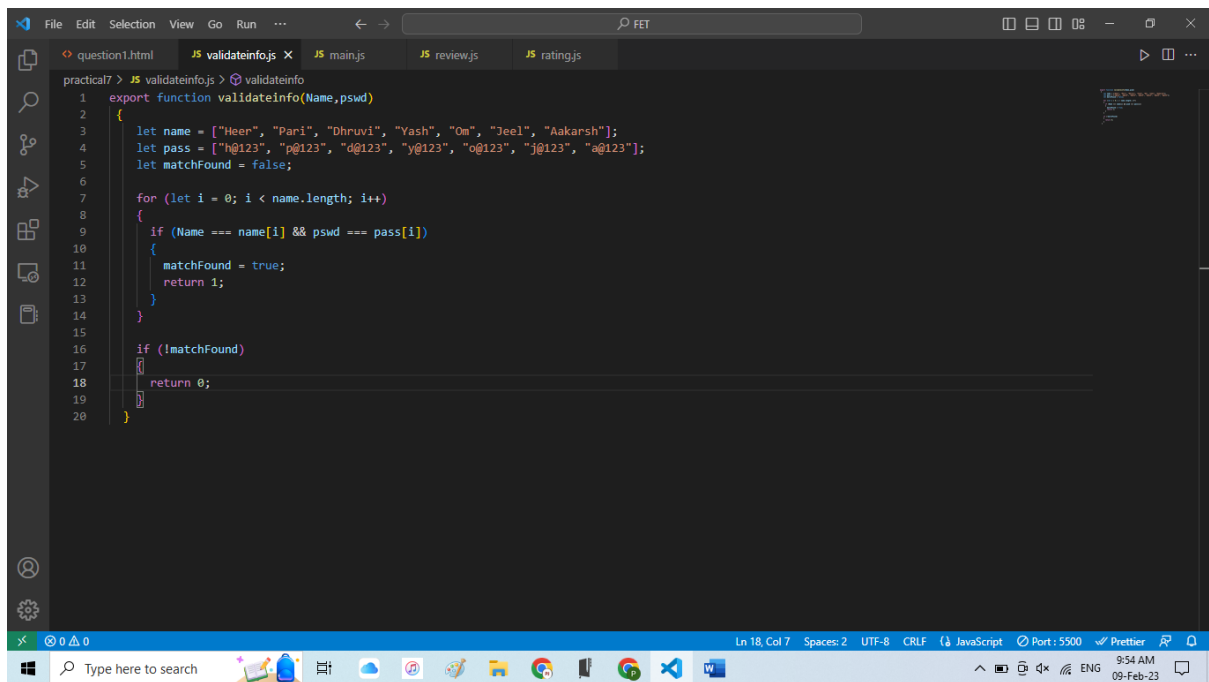


The screenshot shows the FET IDE interface with a file named 'main.js' open. The code is as follows:

```
1 import { validateInfo } from "./validateinfo.js"
2 import { rating } from "./rating.js"
3 import { review } from "./review.js"
4 function pari()
5 {
6   let Name= prompt("Enter Username:");
7   let pswd= prompt("Enter password:");
8   let confirm=validateInfo(Name,pswd);
9   {
10     if(confirm==1)
11     {
12       alert("You Have Successfully Logged in \n Thank you!! \n You may proceed further");
13       review();
14       rating();
15     }
16     else
17     {
18       alert("Sorry Try Again \n Check your details properly");
19     }
20   }
21 }
22
23
24
25 window.pari=pari
```

The status bar at the bottom indicates 'Ln 13, Col 22', 'Spaces: 4', 'UTF-8', 'CRLF', 'JavaScript', 'Port: 5500', and 'Prettier'.

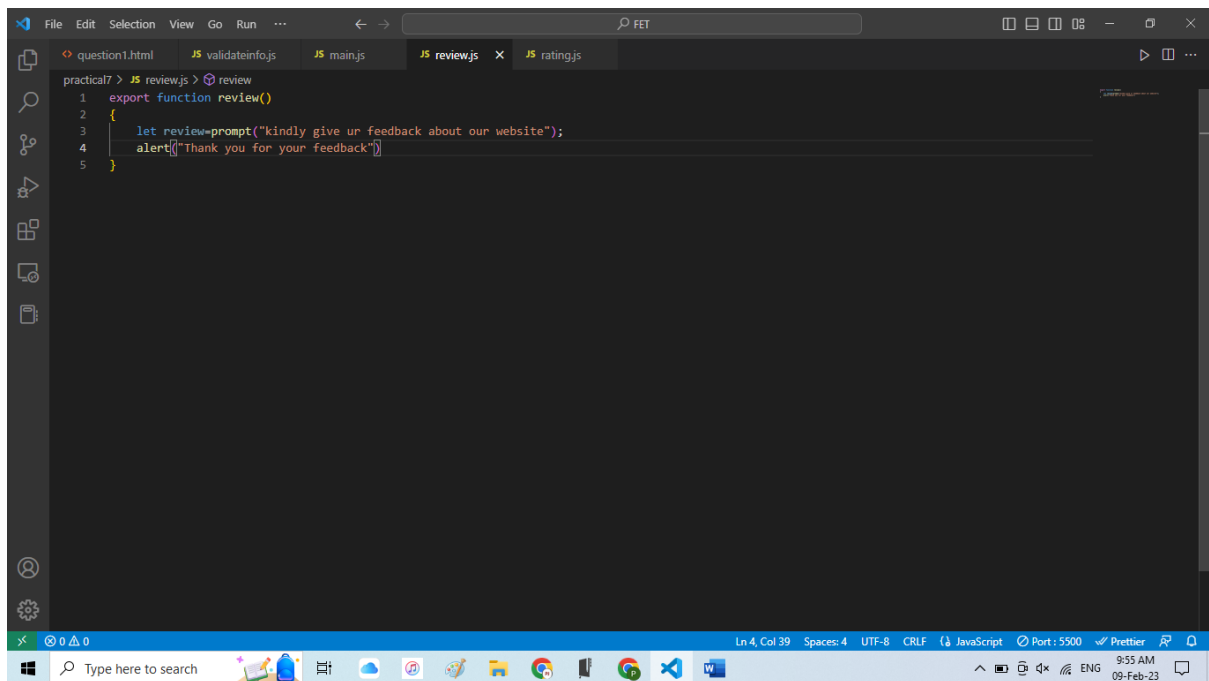
VALIDATE INFO FILE:-



The screenshot shows the VS Code editor with the file `validateinfo.js` open. The code defines a function `validateinfo(Name,pswd)` that checks if a name and password match from predefined arrays. The name array contains ["Heer", "Pari", "Dhruvi", "Yash", "Om", "Jeel", "Aakash"], and the password array contains ["h@123", "p@123", "d@123", "y@123", "o@123", "j@123", "a@123"]. The function returns 1 if a match is found and 0 otherwise.

```
1 export function validateinfo(Name,pswd)
2 {
3   let name = ["Heer", "Pari", "Dhruvi", "Yash", "Om", "Jeel", "Aakash"];
4   let pass = ["h@123", "p@123", "d@123", "y@123", "o@123", "j@123", "a@123"];
5   let matchFound = false;
6
7   for (let i = 0; i < name.length; i++)
8   {
9     if (Name === name[i] && pswd === pass[i])
10    {
11      matchFound = true;
12      return 1;
13    }
14  }
15
16  if (!matchFound)
17  {
18    return 0;
19  }
20 }
```

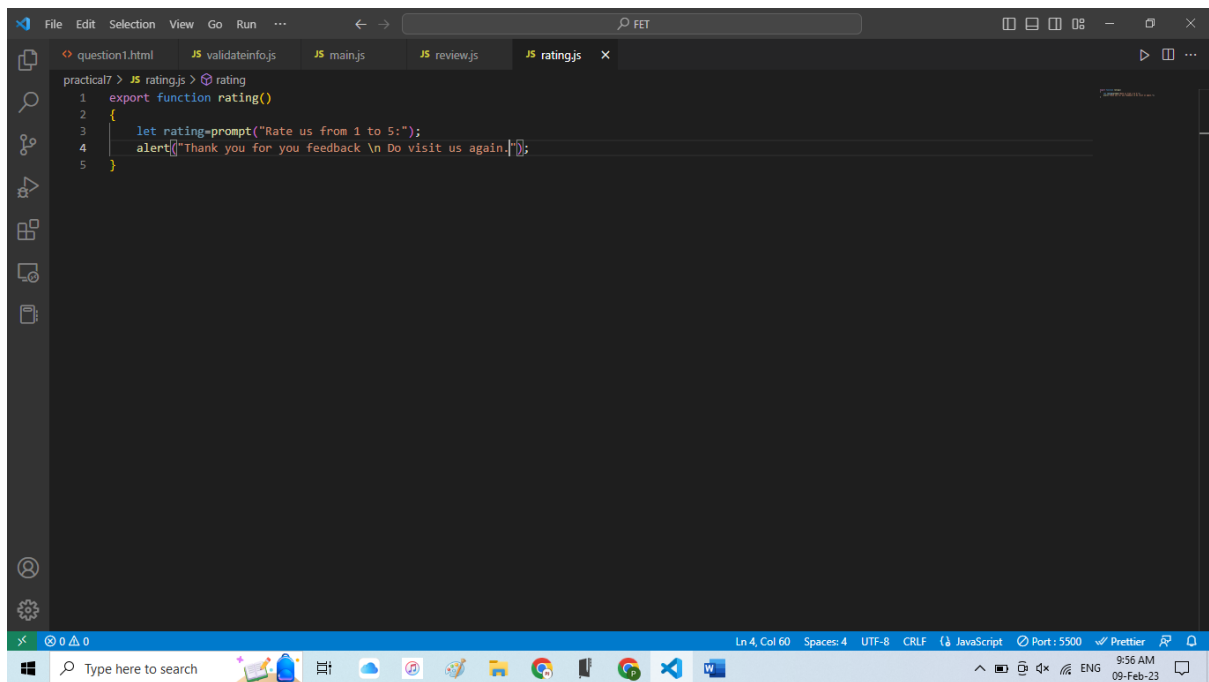
REVIEW FILE:-



The screenshot shows the VS Code editor with the file `review.js` open. The code defines a function `review()` that prompts the user for feedback and displays an alert message.

```
1 export function review()
2 {
3   let review=prompt("kindly give ur feedback about our website");
4   alert("Thank you for your feedback")
5 }
```

RATING FILE:-



A screenshot of the Visual Studio Code editor interface. The 'rating.js' file is open in the editor, showing the following JavaScript code:

```
1 export function rating()
2 {
3   let rating=prompt("Rate us from 1 to 5:");
4   alert("Thank you for you feedback \n Do visit us again.");
5 }
```

The interface includes a sidebar on the left with icons for Explorer, Search, Source Control, and Run and Debug. The bottom status bar shows 'Ln 4, Col 60', 'Spaces: 4', 'UTF-8', 'CRLF', 'JavaScript', 'Port: 5500', 'Prettier', and the system clock '9:56 AM 09-Feb-23'.

OUTPUT:-

