Lesson 05 Demo 07

Working with Babel

Objective: To demonstrate the process of configuring and using Babel to compile and transform modern JavaScript code for enhanced compatibility across various browser environments

Tools required: Visual Studio Code and Node.js

Prerequisites: None

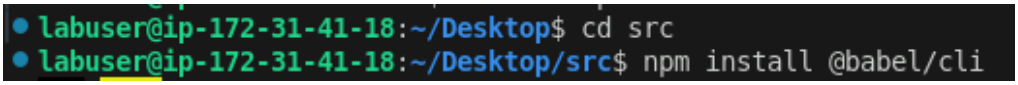
Steps to be followed:

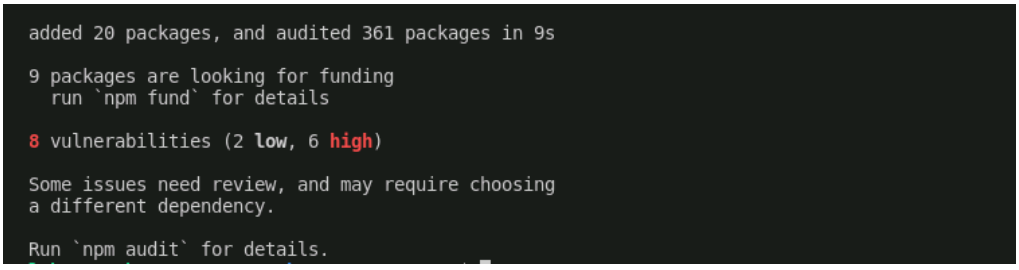
1. Write a program in JavaScript for Babel
2. Execute the program and verify the implementation of Babel

Step 1: Write a program in JavaScript for Babel

1.1 Create a folder named **src** and write the following command in the terminal to install Babel CLI:

npm install @babel/cli





1.2 Write the following command to install **babel-core** modules:

npm install @babel/core

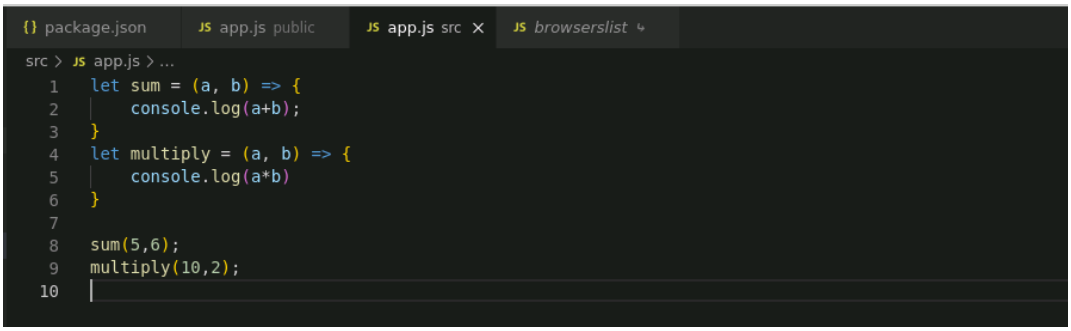
1.3 Open **package.json** and add the following build script:

"scripts": {
x"build": "node_modules/.bin/babel"
}

```
"scripts": {
  "test": "echo \"Error: no test specified\" && exit 1",
  "version": "babel --version",
  "build": "node_modules/.bin/babel"
},
```

1.4 Create a file named **app.js** under the **src** folder and enter the following code:

```
let sum = (a, b) => {
   console.log(a+b);
}
let multiply = (a, b) => {
   console.log(a*b)
}
sum(5,6);
multiply(10,2);
```

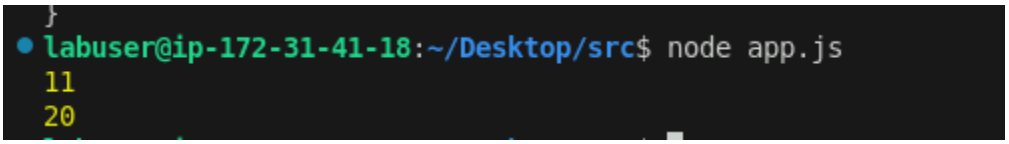


Step 2: Execute the program and verify the implementation of Babel

2.1 Write the following command to execute the **app.js** file:
npm run build src/app.js --out-file=public/scripts/app.js --presets=env

```
labuser@ip-172-31-41-18:~/Desktop/src$ npm run build src/app.js --out-file=public/scripts/app.js --presets=env
```

2.2 Write the following command to execute the **app.js** file:
node app.js



By following the above steps, you have successfully configured and used Babel to compile and transform modern JavaScript code, ensuring enhanced compatibility across various browser environments while maintaining code efficiency and readability.