3.10 Modern JavaScript: Working with Babel

This section will guide you to:

* Create a JavaScript project in your IDE
* Write a program in JavaScript to implement Babel

This lab has three subsections, namely:

* + 1. Writing a program in JavaScript to verify the implementation of Babel
    2. Executing the program and verifying the implementation of Babel
    3. Pushing the code to your GitHub repositories

**Step 3.10.1:** Writing a program in JavaScript to verify the implementation of Babel

* Create the folder with name as demo.
* Create a folder as public folder named *public*. The *public* folder contains a file named *index.html*.
* Then create the src folder and create the app.js file.
* Run the following command to install Babel:

*npm install -g babel-cli@6.24.1*

You can run *babel –help* to check all the Babel options.

* Run the following command to set up your project:

*yarn init*

Leave all options as default.

* Run the following command to install Babel presets:

*yarn add babel-preset-es2015 babel-preset-env@1.5.2*

* *In app.js* file which contains inside are src folder. This will contain the JSX code.

*let* sum = (*a*, *b*) *=>* {

   return *a*+*b*;

}

*let* sub = (*a*, *b*) *=>* {

    return *a*-*b*;

}

*let* mul = (*a*,*b*)*=>* {

    return *a*\**b*;

}

*let* div =(*a*,*b*)*=>*{

    return *a*/*b*;

}

console.log("sum of two number is "+sum(10,20));

console.log("sub of two number is "+sub(100,20));

console.log("mul of two number is "+mul(1,20));

console.log("div of two number is "+div(100,20));

**Step 3.10.2:** Executing the program and verifying implementation of Babel

Before you execute the program, check for syntactical corrections. If no errors are found, follow the steps mentioned below:

* Create dist folder inside a project.
* Run the following command in the terminal:

*babel src/app.js --out-file=dist/app.js --presets=env,es2015*

You can see inside a dist folder created ES5 types files.

"use strict";

*var* sum = *function* sum(*a*, *b*) {

    return *a* + *b*;

};

*var* sub = *function* sub(*a*, *b*) {

    return *a* - *b*;

};

*var* mul = *function* mul(*a*, *b*) {

    return *a* \* *b*;

};

*var* div = *function* div(*a*, *b*) {

    return *a* / *b*;

};

console.log("sum of two number is " + sum(10, 20));

console.log("sub of two number is " + sub(100, 20));

console.log("mul of two number is " + mul(1, 20));

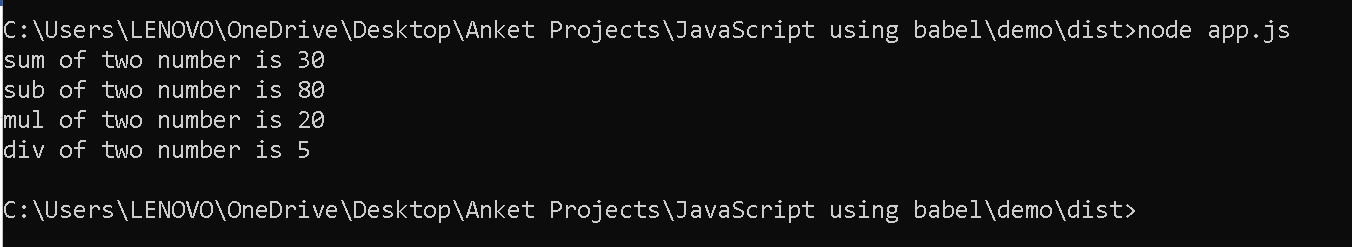
console.log("div of two number is " + div(100, 20));

* Open a new terminal and run the following commands:

*cd dist*

*node app.js*

*output*



In index.html page link to this file

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <script src="../dist/app.js"></script>

</head>

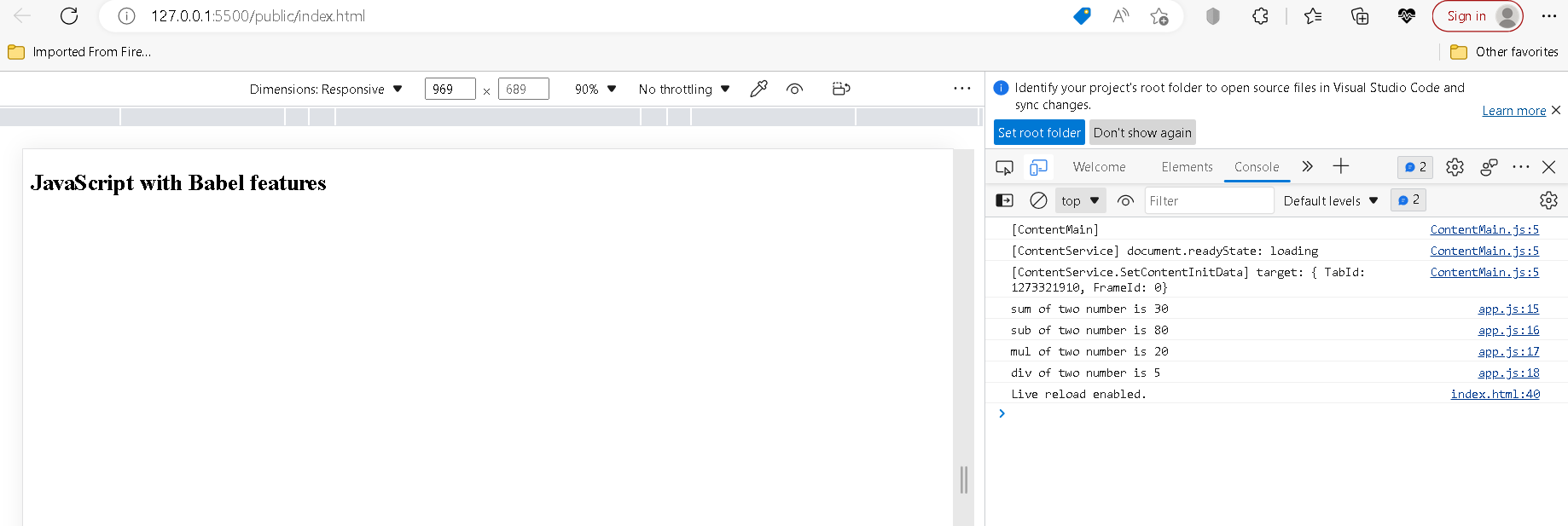
<body>

    <h2>JavaScript with Babel features</h2>

</body>

</html>

Run index.html page on browser



**Step 3.10.3:** Pushing the code to your GitHub repositories

Open your command prompt and navigate to the folder where you have created your files.

cd <folder path>

Initialize your repository using the following command:

git init

Add all the files to your git repository using the following command:

git add . 

Commit the changes using the following command:

git commit . -m “Changes have been committed.”

Push the files to the folder you initially created using the following command:

git push -u origin master