

Lab 2

Title: Introduction to Test-Driven Development (TDD) and Behavior-Driven Development (BDD)

Objective:

To understand and apply the principles of Test-Driven Development (TDD) and Behavior-Driven Development (BDD) in a Node.js environment using Jest.

Theory:

Test-Driven Development (TDD):

TDD is a software development approach where tests are written before the actual code. The process follows a short, repeatable cycle:

1. Write a failing test.
2. Write just enough code to pass the test.
3. Refactor the code while ensuring the test still passes.

Behavior-Driven Development (BDD):

BDD builds upon TDD by writing tests in a human-readable, natural language format (e.g., Given-When-Then). It focuses more on the behavior and expectations of the system from the user's perspective.

Code

Step 1: Initialize a Node.js project

```
npm init -y
```

Step 2: Install Jest testing library

```
npm install --save-dev jest
```

Step 3 : Configure test script in package.json

```
"scripts": { "test": "jest" }
```

Step 4: Create a function file sum.js

```
function sum(a, b) {  
  return a+b;  
}  
  
module.exports = sum;
```

Step 5 : Create a test file sum.test.js

```
const sum = require('./sum');  
test('adds 2 + 3 to equal 5', () => {  
  expect( sum(2, 3)), tobe(5);  
});
```

Step 6 : Run the test

```
npm test
```

Conclusion

In this lab, we practiced writing tests before implementation (TDD) and describing behaviors in test cases (BDD). TDD emphasizes unit testing, while BDD focuses on system behavior and collaboration between developers and non-technical stakeholders.