Experiment No. - 2

Student Name: Devanand Utkarsh **UID:** 24MCA20454
Branch: MCA **Section/Group:** 6(B)
Semester: II **Date of Performance:** 23/01/25
Subject Name: Software Testing **Subject Code:** 24CAH-654

Aim/Overview of the practical:

Write a test script using unit testing framework for addition, subtraction, multiplication and division of two numbers.

Objective:

Develop a test script using a unit testing framework to validate the functionality of addition, subtraction, multiplication, and division operations for two numbers, ensuring accuracy, edge case handling, and compliance with expected mathematical behavior.

Requirement:

- VS Code as the code editor.
- Jest for JavaScript testing.

Procedure/Algorithm/Code:

math.js

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

```
        function multiply(a, b) {
          return a * b;
        }
        function divide(a, b) {
          if (a !== 0 && b !== 0) {
            return a / b;
          } else {
            return "not divisible by 0";
          }
        }
module.exports = { add, substract, multiply, divide };
```

🗎 math.test.js

```
        const { add, substract, multiply, divide } =
        require('./math');

        describe('Calculator functions', () => {
          test('adds 300+ 3 to equal 303', () => {
            expect(add(300, 3)).toBe(303);
          });

          test('subtracts 300 - 3 to equal 297', () => {
            expect(substract(300, 3)).toBe(297);
          });

          test('multiplies 4 * 3 to equal 12', () => {
            expect(multiply(4, 3)).toBe(12);
```

```
    });


    test('divides 126 / 2 to equal 0', () => {
        expect(divide(126, 2)).toBe(63);
    });


    test('returns error message when dividing by 0', ()
=> {
        expect(divide(8, 0)).toBe("not divisible by 0");
        expect(divide(0, 8)).toBe("not divisible by 0");
    });
});
```

TEST CASES:

Test Case ID: TC001 **Status = passed**

Module Name: Addition Module

Test Designed By: Devanand Utkarsh

Test Designed Date: 2025-01-23

Test Executed By: Devanand Utkarsh

Test Title/Name: Addition Test

Test Steps: Enter 300 and 3 as inputs.

Expected Result: Result: 303

Actual Result: Result: 303

Test Case ID: TC002 **Status = Passed**

Module Name: Subtraction Module

Test Designed By: Devanand Utkarsh

Test Designed Date: 2025-01-23

Test Executed By: Devanand Utkarsh

Test Title/Name: Subtraction Test

Test Steps: Enter 300 and 3 as inputs.

Expected Result: Result: 297

Actual Result: Result: 297

Test Case ID: TC003 **Status = passed**

Module Name: Multiplication Module

Test Designed By: Devanand Utkarsh

Test Designed Date: 2025-01-23

Test Executed By: Devanand Utkarsh

Test Title/Name: Multiplication Test

Test Steps: Enter 4 and 3 as inputs.

Expected Result: Result: 12

Actual Result: Result: 12

Test Case ID: TC004 **Status = passed**

Module Name: Division Module

Test Designed By: Devanand Utkarsh

Test Designed Date: 2025-01-23

Test Executed By: Devanand Utkarsh

Test Title/Name: Division Test

Test Steps: Enter 126 and 2 as inputs.

Expected Result: Result: 63

Actual Result: Result: 63

CHANDIGARH UNIVERSITY Discover. Learn. Empower. NAAC GRADE A+ Accredited University

Test Case ID: TC005 **Status = passed**

Module Name: Divide by Zero Test Module

Test Designed By: Devanand Utkarsh

Test Designed Date: 2025-01-23

Test Executed By Devanand Utkarsh

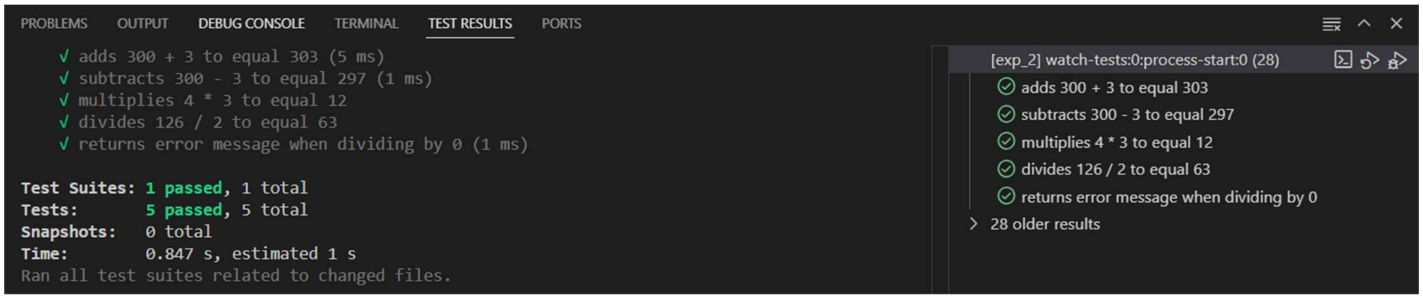
Test Title/Name: Divide by Zero Test

Test Steps: Enter 8 and 0 as inputs.

Expected Result: Result: Error: Division by zero is not allowed.

Actual Result: Result: Error: Division by zero is not allowed.

Output:



```
    √ adds 300 + 3 to equal 303 (5 ms)
    √ subtracts 300 - 3 to equal 297 (1 ms)
    √ multiplies 4 * 3 to equal 12
    √ divides 126 / 2 to equal 63
    √ returns error message when dividing by 0 (1 ms)

Test Suites: 1 passed, 1 total
Tests:       5 passed, 5 total
Snapshots:   0 total
Time:        0.847 s, estimated 1 s
Ran all test suites related to changed files.
```

```
[exp_2] watch-tests:0:process-start:0 (28)
  ✓ adds 300 + 3 to equal 303
  ✓ subtracts 300 - 3 to equal 297
  ✓ multiplies 4 * 3 to equal 12
  ✓ divides 126 / 2 to equal 63
  ✓ returns error message when dividing by 0
> 28 older results
```

Learning outcomes (What I have learnt):

1. Understanding the use of a unit testing framework for validating code functionality.
2. Writing test cases for mathematical operations (addition, subtraction, multiplication, division).
3. Structuring and organizing test scripts for reusability and clarity.
4. Interpreting test results to identify and resolve potential issues in code logic.