**Cypress automation framework:**

**Objective :**

Test ahfarmer calculator functionality

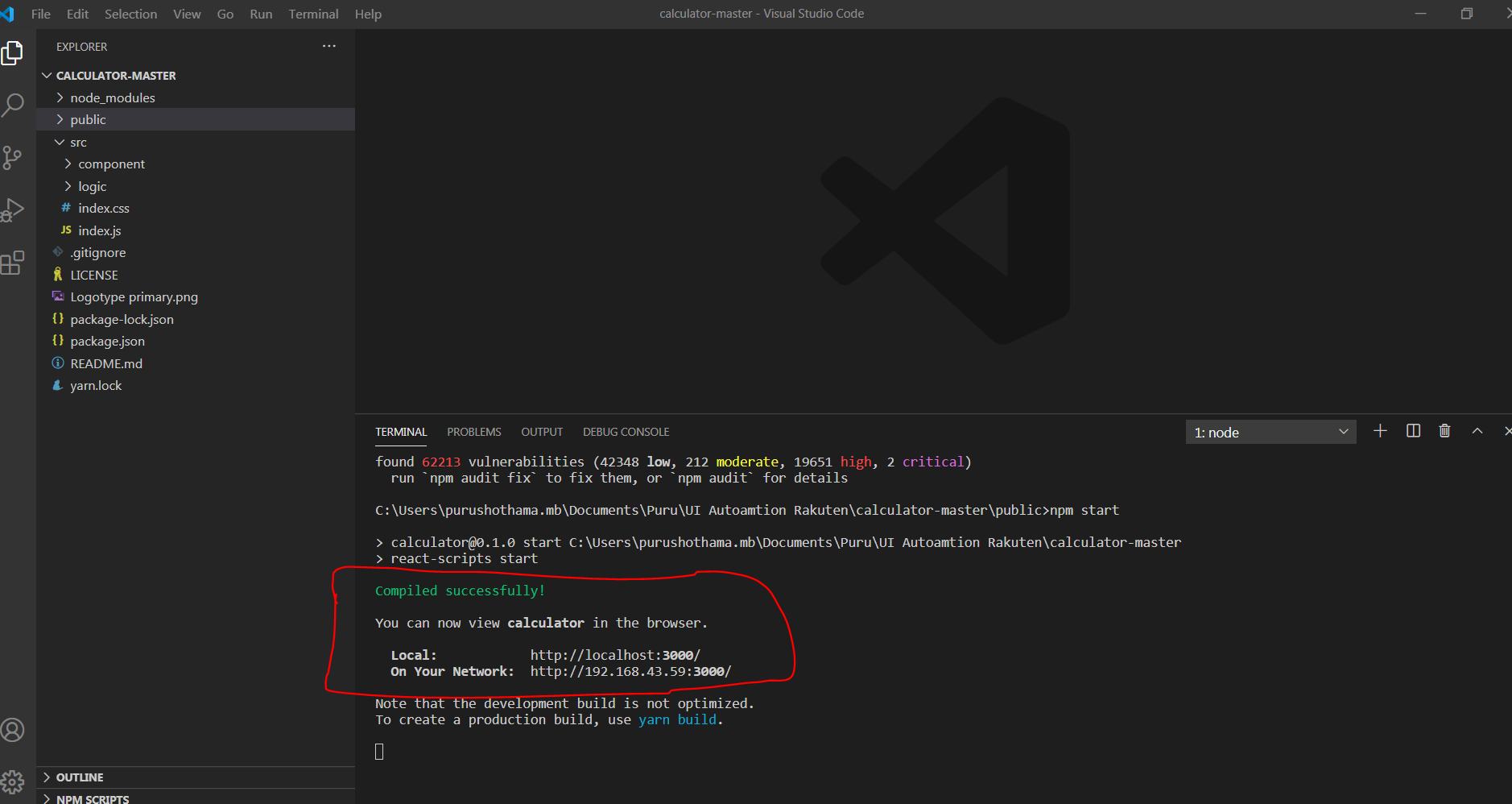
* Addition , subtraction,multiplication, etc and AC and sign functionalities

**Cypress Framework set up nad pre-requisites:**

1. Install node.js
2. Install cypress
3. Install mocha javascript unit testing framework
4. Install chai.js to use assertion in test script
5. Downlaod and install editor : Vs code editor

**Project set up (locally ):**

1. Fork the project from github
2. Usage :
   * 1. Go to project root folder
     2. Init the npm
     3. Npm start – server should start and project should be able to run locally on 3000 port (refer below screenshot)



1. Create project and open in VS code editor
2. Install cypress and necessary libraried like chai and mocha.

**Test case:**

**Pre-requisites:**

* Application should be set up locall to run localhost:3000

Launch the calcualtor application: url- <http://localhost:3000/>

Verify the Arithmentic functionality: Add,Sub,Mul,Div and AC functionalities

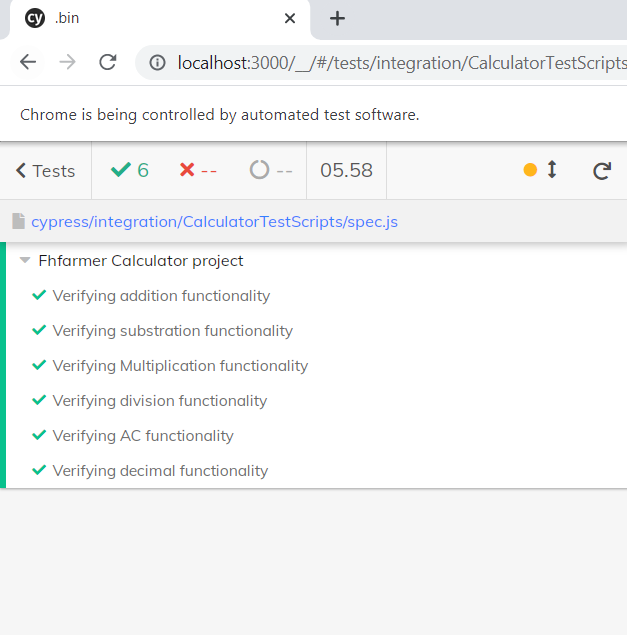
Perform Add,Sub,Mul,Div and AC functionalities and validate the correct results

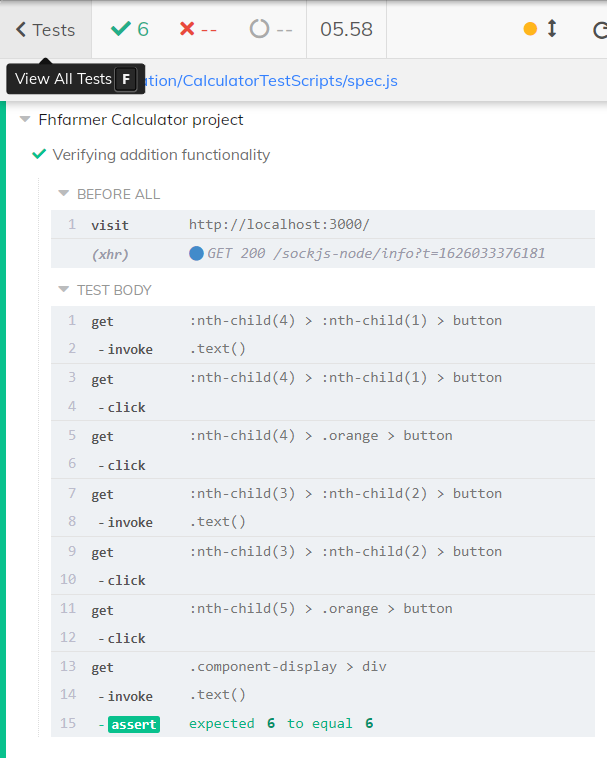
AC functionality : Choose any integer or decimal number

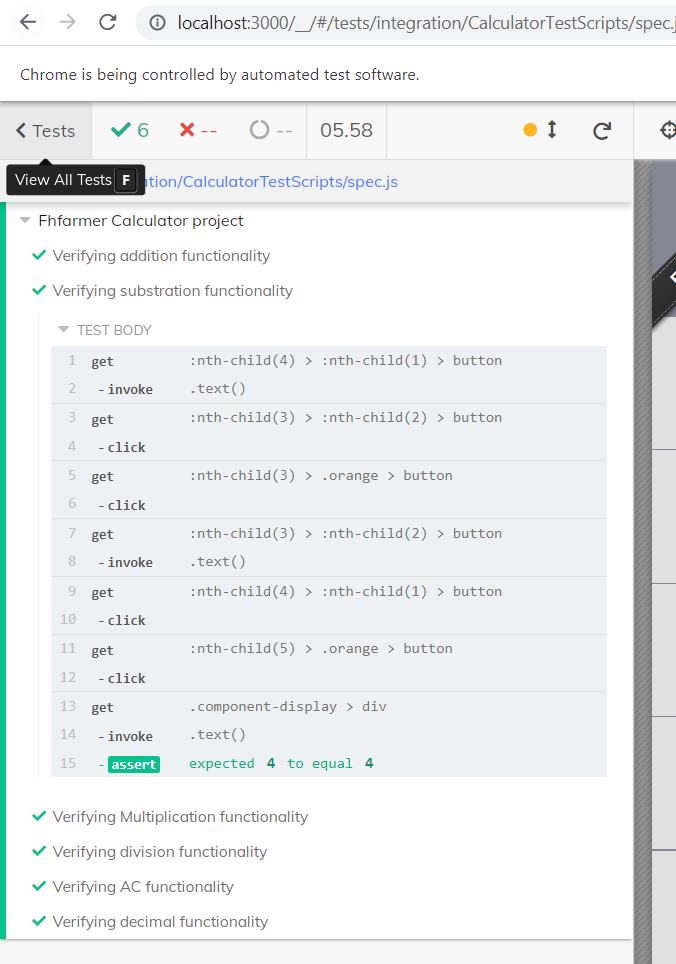
Click on AC button , value in the result text box should set to zero

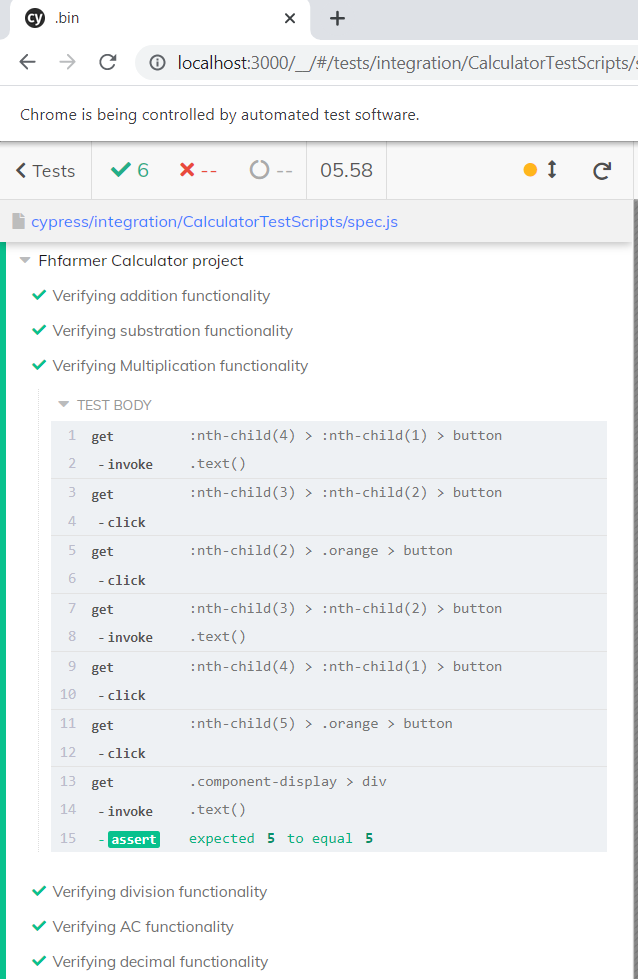
Decimal functionality : Take any two flaoting point numbers and sum with each other and verify the decimal/floating point results

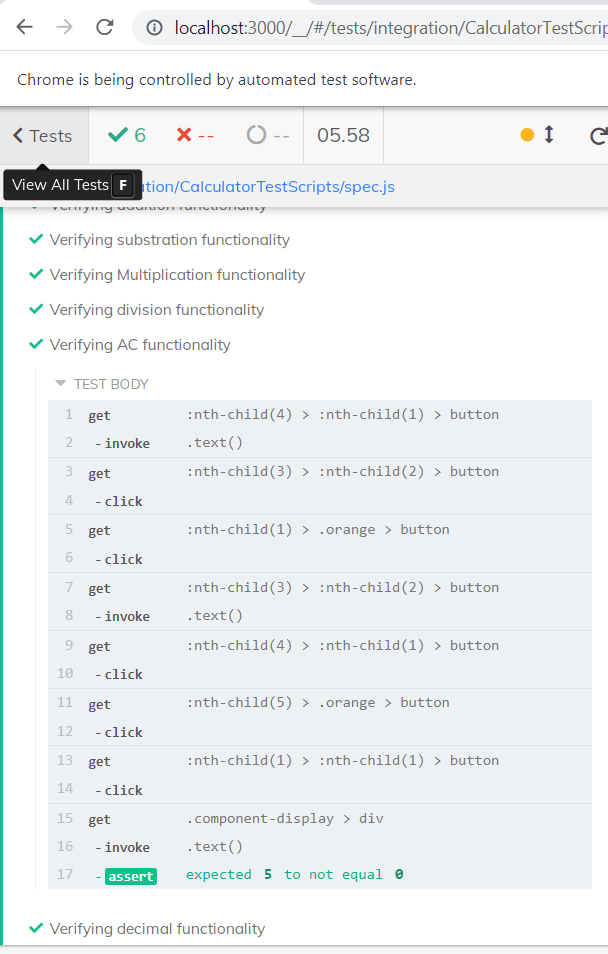
**Test Scripts :**













**Source Code for refrence :**

1. **Cypress lib commands :visit,click etc**
2. **Used chai assertion for validation**
3. **Invoke() to retrieve the text of an element**
4. **Cypress test runner to run the test on chrome browser**

/// <reference types="Cypress" />

var text1;

var text2;

var outputVal;

var outputVal2;

var decValBefAdin1;

var decValBefAdin2;

describe("Fhfarmer Calculator project", () => {

    before("Launch Application", () => {

        const strURL = "http://localhost:3000/";

        cy.visit(strURL)

    })

    it("Verifying addition functionality", () => {

        console.log("addition functionality")

        //Clicking on number and getting its text

        cy.get(':nth-child(4) > :nth-child(1) > button')

            .invoke('text')

            .then((text1) => {

                cy.get(':nth-child(4) > :nth-child(1) > button').click()

                console.log(text1)

                //Clicking on Plus sign

                cy.get(':nth-child(4) > .orange > button').click()

                //Clicking on number and getting its text

                cy.get(':nth-child(3) > :nth-child(2) > button')

                    .invoke('text')

                    .then((text2) => {

                        cy.get(':nth-child(3) > :nth-child(2) > button').click();

                        console.log(text2)

                        //Clicking on equals sign

                        cy.get(':nth-child(5) > .orange > button').click();

                        //Retieving the out put value after addition

                        cy.get('.component-display > div')

                            .invoke('text')

                            .then((outputVal) => {

                                //cy.get('.component-display > div')

                                console.log(outputVal)

                                //Asserting the output value

                                expect((parseInt(text2) + parseInt(text1))).to.equal(parseInt(outputVal))

                            })

                    })

            })

    })

    it("Verifying substration functionality", () => {

        console.log("substration functionality")

        //Clicking on number and getting its text

        cy.get(':nth-child(4) > :nth-child(1) > button')

            .invoke('text')

            .then((text1) => {

                cy.get(':nth-child(3) > :nth-child(2) > button').click();

                console.log(text1)

                //Clicking on minus sign

                cy.get(':nth-child(3) > .orange > button').click()

                //Clicking on number and getting its text

                cy.get(':nth-child(3) > :nth-child(2) > button')

                    .invoke('text')

                    .then((text2) => {

                        cy.get(':nth-child(4) > :nth-child(1) > button').click()

                        console.log(text2)

                        //Clicking on equals sign

                        cy.get(':nth-child(5) > .orange > button').click();

                        //Retieving the out put value after addition

                        cy.get('.component-display > div')

                            .invoke('text')

                            .then((outputVal) => {

                                //cy.get('.component-display > div')

                                console.log(outputVal)

                                //Asserting the output value

                                expect((parseInt(text2) - parseInt(text1))).to.equal(parseInt(outputVal))

                            })

                    })

            })

    })

    it("Verifying Multiplication functionality", () => {

        console.log("Multiplication functionality")

        //Clicking on number and getting its text

        cy.get(':nth-child(4) > :nth-child(1) > button')

            .invoke('text')

            .then((text1) => {

                cy.get(':nth-child(3) > :nth-child(2) > button').click();

                console.log(text1)

                //Clicking on multiplication sign

                cy.get(':nth-child(2) > .orange > button').click();

                //Clicking on number and getting its text

                cy.get(':nth-child(3) > :nth-child(2) > button')

                    .invoke('text')

                    .then((text2) => {

                        cy.get(':nth-child(4) > :nth-child(1) > button').click()

                        console.log(text2)

                        //Clicking on equals sign

                        cy.get(':nth-child(5) > .orange > button').click();

                        //Retieving the out put value after addition

                        cy.get('.component-display > div')

                            .invoke('text')

                            .then((outputVal) => {

                                //cy.get('.component-display > div')

                                console.log(outputVal)

                                //Asserting the output value

                                expect((parseInt(text2) \* parseInt(text1))).to.equal(parseInt(outputVal))

                            })

                    })

            })

    })

    it("Verifying division functionality", () => {

        console.log("division functionality")

        //Clicking on number and getting its text

        cy.get(':nth-child(4) > :nth-child(1) > button')

            .invoke('text')

            .then((text1) => {

                cy.get(':nth-child(3) > :nth-child(2) > button').click();

                console.log(text1)

                //Clicking on division sign

                cy.get(':nth-child(1) > .orange > button').click();

                //Clicking on number and getting its text

                cy.get(':nth-child(3) > :nth-child(2) > button')

                    .invoke('text')

                    .then((text2) => {

                        cy.get(':nth-child(4) > :nth-child(1) > button').click()

                        console.log(text2)

                        //Clicking on equals sign

                        cy.get(':nth-child(5) > .orange > button').click();

                        //Retieving the out put value after addition

                        cy.get('.component-display > div')

                            .invoke('text')

                            .then((outputVal) => {

                                //cy.get('.component-display > div')

                                console.log(outputVal)

                                //Asserting the output value

                                expect((parseInt(text2) \* parseInt(text1))).to.equal(parseInt(outputVal))

                            })

                    })

            })

    })

    it("Verifying AC functionality", () => {

        console.log("AC functionality")

        //Clicking on number and getting its text

        cy.get(':nth-child(4) > :nth-child(1) > button')

            .invoke('text')

            .then((text1) => {

                cy.get(':nth-child(3) > :nth-child(2) > button').click();

                console.log(text1)

                //Clicking on division sign

                cy.get(':nth-child(1) > .orange > button').click();

                //Clicking on number and getting its text

                cy.get(':nth-child(3) > :nth-child(2) > button')

                    .invoke('text')

                    .then((text2) => {

                        cy.get(':nth-child(4) > :nth-child(1) > button').click()

                        console.log(text2)

                        //Clicking on equals sign

                        cy.get(':nth-child(5) > .orange > button').click();

                        //Click on AC button

                        cy.get(':nth-child(1) > :nth-child(1) > button').click();

                        //Retieving the out put value after addition

                        cy.get('.component-display > div')

                            .invoke('text')

                            .then((outputVal) => {

                                //cy.get('.component-display > div')

                                console.log(outputVal)

                                //Asserting the output value

                                expect((parseInt(text2) \* parseInt(text1))).to.not.equal(parseInt(outputVal))

                            })

                    })

            })

    })

    it("Verifying decimal functionality", () => {

        console.log("decimal functionality")

        //Clicking on number

        cy.get(':nth-child(3) > :nth-child(2) > button').click();

       //Clicking on decimal point

       cy.get(':nth-child(5) > :nth-child(2) > button').click();

       //Clicking on number

       cy.get(':nth-child(4) > :nth-child(3) > button').click();

       //getting decimal value before addtion

       cy.get('.component-display > div')

       .invoke('text')

       .then((decValBefAdin1)=>{

      console.log(decValBefAdin1)

        //Clicking plus sign

        cy.get(':nth-child(4) > .orange > button').click();

         //Clicking on number

        cy.get(':nth-child(3) > :nth-child(2) > button').click();

        //Clicking on decimal point

        cy.get(':nth-child(5) > :nth-child(2) > button').click();

        //Clicking on number

        cy.get(':nth-child(4) > :nth-child(3) > button').click();

        //getting decimal value before addtion

        cy.get('.component-display > div')

            .invoke('text')

            .then((decValBefAdin2)=>{

                console.log(decValBefAdin2)

            //Clicking on equals sign

            cy.get(':nth-child(5) > .orange > button').click();

            //Retieving the out put value after addition

            cy.get('.component-display > div')

                .invoke('text')

                .then((outputVal2) => {

                    //cy.get('.component-display > div')

                    console.log(outputVal)

                    //Asserting the output value

                    expect((parseFloat(decValBefAdin1) + parseFloat(decValBefAdin2))).to.equal(parseFloat(outputVal2))

                })

            })

       })

    })

})