**EXPERIMENT 2**

**PROGRAMS:**

1. **Calculator in TypeScript**

**Code:**

class Calculator {

private number1: number;

private number2: number;

add(number1: number, number2: number) {

return number1+number2;

}

subtract(number1: number, number2: number) {

return number1-number2;

}

multiply(number1: number, number2: number) {

return number1\*number2;

}

divide(number1: number, number2: number) {

return number1/number2;

}

}

const calculator = new Calculator();

const result1 = calculator.add(5,3)

const result2 = calculator.multiply(5,3)

const result3 = calculator.subtract(5,3)

const result4 = calculator.divide(5,3)

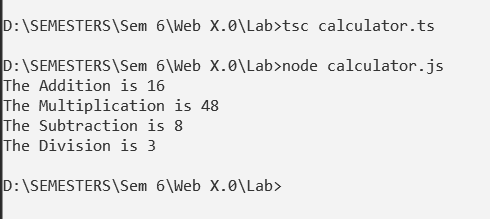
console.log(`The Addition is ${result1}`);

console.log(`The Multiplication is ${result2}`);

console.log(`The Subtraction is ${result3}`);

console.log(`The Division is ${result4}`);

**Output:**



1. **Inheritance using TypeScript**

**Code:**

class Colors {

name: string;

constructor(name: string) {

this.name = name;

}

splash(radius: number = 0) {

console.log(`${this.name} splashed on the wall with a radius of ${radius}m.`);

}

}

class Blue extends Colors{

constructor(name: string) {

super(name);

}

spill() {

console.log('Oops! It spilled');

}

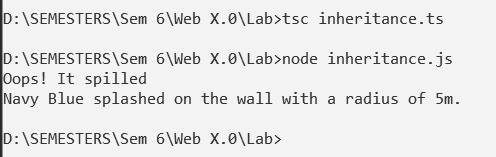
}

const blue = new Blue('Navy Blue');

blue.spill();

blue.splash(5);

**Output:**



1. **Access Modifier in TypeScript**

**Code:**

class Person {

private firstName: string;

protected lastName: string;

public age: number;

constructor(firstName: string, lastName: string, age: number) {

this.firstName = firstName;

this.lastName = lastName;

this.age = age;

}

private getFullName(): string {

return `${this.firstName} ${this.lastName}`;

}

public getGreeting(): string {

return `Hello, my name is ${this.getFullName()} and I am ${this.age} years old.`;

}

}

class Employee extends Person {

private department: string;

constructor(firstName: string, lastName: string, age: number, department: string) {

super(firstName, lastName, age);

this.department = department;

}

public getEmployeeInfo(): string {

return `${this.getGreeting()} I work in the ${this.department} department.`;

}

}

const person = new Person("Sakshi", "Patil", 21);

console.log(person.getGreeting());

const employee = new Employee("Smit", "Sawant", 29, "Sales");

console.log(employee.age);

console.log(employee.getGreeting());

console.log(employee.getEmployeeInfo());

**Output:**

