**Aim:** Demonstrate Access Modifiers example using TypeScript

**Program:** class Person {

  public name: string;

  private age: number;

  protected email: string;

  constructor(name: string, age: number, email: string) {

    this.name = name;

    this.age = age;

    this.email = email;

  }

  public displayAge(): void {

    console.log(`Age: ${this.age}`);

  }

  protected displayEmail(): void {

    console.log(`Email: ${this.email}`);

  }

}

class Employee extends Person {

  private employeeId: number;

  constructor(name: string, age: number, email: string, employeeId: number) {

    super(name, age, email);

    this.employeeId = employeeId;

  }

  public displayDetails(): void {

    console.log(`Name: ${this.name}`);

    this.displayAge();

    console.log(`Employee ID: ${this.employeeId}`);

    this.displayEmail();

  }

}

const person = new Person("RS", 21, "rs@gmail.com");

console.log("Public Property Access:");

console.log(`Name: ${person.name}`);

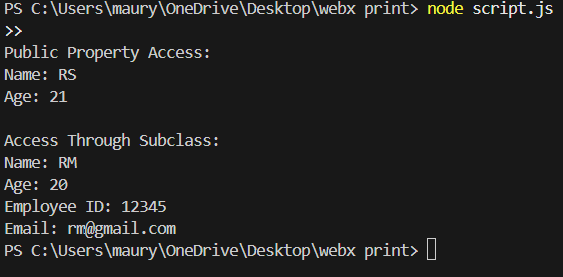
person.displayAge()

console.log("\nAccess Through Subclass:");

const employee = new Employee("RM", 20, "rm@gmail.com", 12345);

employee.displayDetails();

**OUTPUT:**

****