//3.Write a “person” class to hold all the details.

class Person {

  constructor(name, email, phone, city) {

    this.name = name;

    this.email = email;

    this.phone = phone;

    this.city = city;

  }

  displayPersonDetails () {

    console.log(`${this.name} is from ${this.city}`)

  }

}

const p1 = new Person("raji","raji@gmail.com",9476542109,"Coimbatore");

const p2 = new Person("lakshmi","lakshmi@gmail.com",9872312548,"Chennai");

console.log(p1);

console.log(p2);

p1.displayPersonDetails();

p2.displayPersonDetails();

//4.write a class to calculate the Uber price.

class UberCostCalculator {

  constructor(basePrice = 50, costPerKilometer = 13, costPerMinute = 2) {

    this.basePrice = Number(basePrice);

    this.costPerKilometer = Number(costPerKilometer);

    this.costPerMinute = Number(costPerMinute);

  }

  costCalculation(distanceInKilometers, timeInMinutes) {

    const totCost = this.basePrice + (this.costPerKilometer \* Number(distanceInKilometers)) + (this.costPerMinute \* Number(timeInMinutes)) ;

    console.log(`Uber Total cost is Rs.${totCost}`);

  }

}

const uberPrice1 = new UberCostCalculator(40,15,3);

uberPrice1.costCalculation(10.7, 15);

const uberPrice2 = new UberCostCalculator();

uberPrice2.costCalculation(12, 10);

//1.https://github.com/rvsp/typescript-oops/blob/master/Practice/Movie.md

class Movie {

  constructor(title, studio, rating = "PG") {

    this.title = title;

    this.studio = studio;

    this.rating = rating;

  }

  getPG(movieArray) {

    movieArray.forEach((element, index) => {

      console.log(`Movie name${index+1}: ${element}`);

    });

  }

  getMovieDetails() {

    console.log(`Movie Name: ${this.title}, Productions: ${this.studio}, Rating: ${this.rating} `)

  }

}

const film1 = new Movie("Casino Royale", "Eon Productions", "PG13");

const film2 = new Movie("Casino Royale", "Eon Productions");

film1.getPG(['Irugapatru', 'joe', 'sabanayagan']);

film2.getMovieDetails();

//2.https://github.com/rvsp/typescript-oops/blob/master/Practice/class-circle.md

class Circle{

  constructor(radius, color){

      this.radius = radius;

      this.color = color;

  }

  get radiusCircle(){

      return this.radius;

  }

  set radiusCircle(radius){

          this.radius = radius;

  }

  get colorCircle(){

      return this.color;

  }

  set colorCircle(color){

      this.color = color;

  }

  get areaCircle(){

      return (Math.PI \* this.radius \* this.radius).toFixed(2);

  }

  get circumferenceCircle(){

      return (2 \* Math.PI \* this.radius).toFixed(2);

  }

}

var obj1 = new Circle(2,"blue");

console.log(obj1.radiusCircle);

obj1.radiusCircle = 2.2

console.log(obj1.radiusCircle);

console.log(obj1.colorCircle);

obj1.colorCircle = "blue";

console.log(obj1.colorCircle);

console.log(obj1.areaCircle);

console.log(obj1.circumferenceCircle);