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;

    }

    getDetails(){

        console.log("Title : "+this.title);

        console.log("Studio : "+this.studio);

        console.log("Rating : "+this.rating);

    }

}

function getPG(arr)

{

    let sol=[];

    for(let i=0;i<arr.length;i++)

    {

        if(arr[i].rating==="PG")

        {

            sol.push(arr[i]);

        }

    }

    console.log(sol);

}

let obj1=new Movie("Casino Royale","Eon Productions","PG­13");

let obj2=new Movie("Coco","Pixar Animation Studios","R");

let obj3=new Movie("Ferdinand","Blue Sky Studios");

let obj4=new Movie("Gray Man"," Anthony and Joe Russo Productions");

let g=[obj1,obj2,obj3,obj4];

obj1.getDetails();

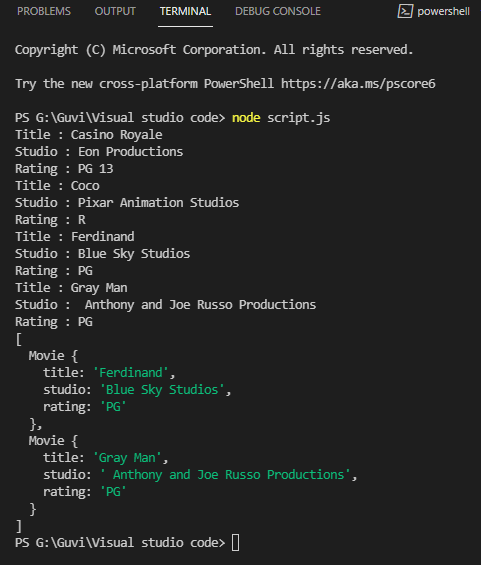
obj2.getDetails();

obj3.getDetails();

obj4.getDetails();

getPG(g);

OUTPUT:



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

class Circle{

    constructor(radius=1.0,color="red")

    {

        this.radius=radius;

        this.color=color;

    }

    getRadius()

    {

        return this.radius;

    }

    setRadius(r)

    {

        this.radius=r;

    }

    getColor()

    {

        return this.color;

    }

    setColor(c)

    {

        this.color=c;

    }

    getArea()

    {

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

    }

    getCircumference()

    {

        return 2\*Math.PI\*this.radius;

    }

    toString()

    {

        return "Radius : "+this.radius+" Color : "+this.color;

    }

}let obj1=new Circle();

obj1.setColor("Blue");

obj1.setRadius("2.5");

console.log("Radius : "+obj1.getRadius());

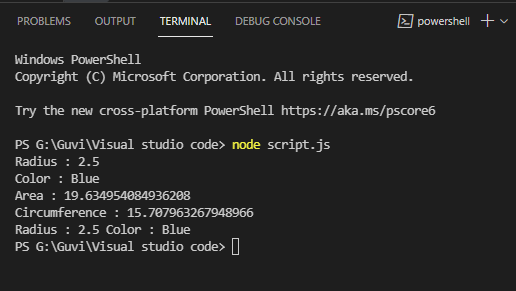
console.log("Color : "+obj1.getColor());

console.log("Area : "+obj1.getArea());

console.log("Circumference : "+obj1.getCircumference());

console.log(obj1.toString());

OUTPUT:



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

class Person{

    constructor(name,age,email,phoneNo)

    {

        this.name=name;

        this.age=age;

        this.email=email;

        this.phoneNo=phoneNo;

    }

    getName()

    {

        return this.name;

    }

    setName(n)

    {

        this.name=n;

    }

    getAge()

    {

        return this.age;

    }

    setAge(a)

    {

        this.age=a;

    }

    getEmail()

    {

        return this.email;

    }

    setEmail(e)

    {

        this.email=e;

    }

    getPhoneNo()

    {

        return this.phoneNo;

    }

    setPhoneNo(p)

    {

        this.phoneNo=p;

    }

    toString()

    {

        return "Name : "+this.name+" Age : "+this.age+" Email : "+this.email+" PhoneNo : "+this.phoneNo;

    }

}

let obj1=new Person();

obj1.setName("Jaga");

obj1.setAge("22");

obj1.setEmail("jaga@gmail.com");

obj1.setPhoneNo("9999999999");

console.log("Name : "+obj1.getName());

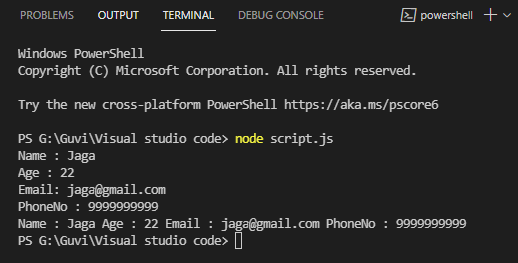
console.log("Age : "+obj1.getAge());

console.log("Email: "+obj1.getEmail());

console.log("PhoneNo : "+obj1.getPhoneNo());

console.log(obj1.toString());

OUTPUT:



4.write a class to calculate uber price.

JS:

class Uber {

    constructor(distance,model)

    {

      this.distance = distance;

      this.model = model;

      this.amount=0;

      this.rate=0;

      this.baseAmount=30;

    }

    calculatePrice(){

        if(this.model==="Uber black")

        {

            this.rate=12;

        }

        else if(this.model==="Uber green")

        {

            this.rate=16;

        }

        else if(this.model==="Uber comfort")

        {

            this.rate=22;

        }

        this.amount=this.amount+this.baseAmount;

        if(this.distance<=20)

        {

            this.amount=this.amount+(this.distance\*(this.rate\*1.5));

        }

        else if(this.distance>20)

        {

            this.amount=this.amount+(this.distance\*this.rate);

        }

        console.log("Bill amount : "+this.amount);

    }

  }

  const uberDrive1 = new Uber(30,"Uber black");

  const uberDrive2 = new Uber(10,"Uber green");

  const uberDrive3 = new Uber(116,"Uber comfort");

uberDrive1.calculatePrice();

uberDrive2.calculatePrice();

uberDrive3.calculatePrice();

OUTPUT:

