/\*1. Create one typescript application which contains one class named as Arithmetic.

Arithmetic class contains three characteristics (Class data members) as Number1, Number2.

Create one parametrised constructor which accept two values and assign it to Number1 and

Number2.

In Arithmetic class we have to write four methods (Behaviours) as Addition, Subtraction ,

Multiplication and Division.

Addition method will add Number1 , Number2 & return result.

Subtraction method will subtract Number1 , Number2 & return result.

Multiplication method will multiply Number1 , Number2 & return result.

Division method will divide Number1 , Number2 & return result.

After designing the class create two objects of that class by providing some hardcoded value.

Call all the methods by using both the objects.

AUTHOR :- RAMESH GHULE

\*/

class Arithmetic{

Number1:number;

Number2:number;

constructor(Number1:number, Number2:number){

this.Number1 = Number1;

this.Number2 = Number2;

}

Addition():number{

return this.Number1+this.Number2;

}

Subtraction():number{

return this.Number1-this.Number2;

}

Multiplication():number{

return this.Number1\*this.Number2;

}

Division():number{

return this.Number1/this.Number2;

}

}

var obj1 = new Arithmetic(20,10);

var obj2 = new Arithmetic(200,50);

console.log("var obj1 = new Arithmetic(20,10)");

console.log("obj1.Addition is= "+obj1.Addition());

console.log("obj1.Subtraction is= "+obj1.Subtraction());

console.log("obj1.Multiplication is= "+obj1.Multiplication());

console.log("obj1.Division is= "+obj1.Division());

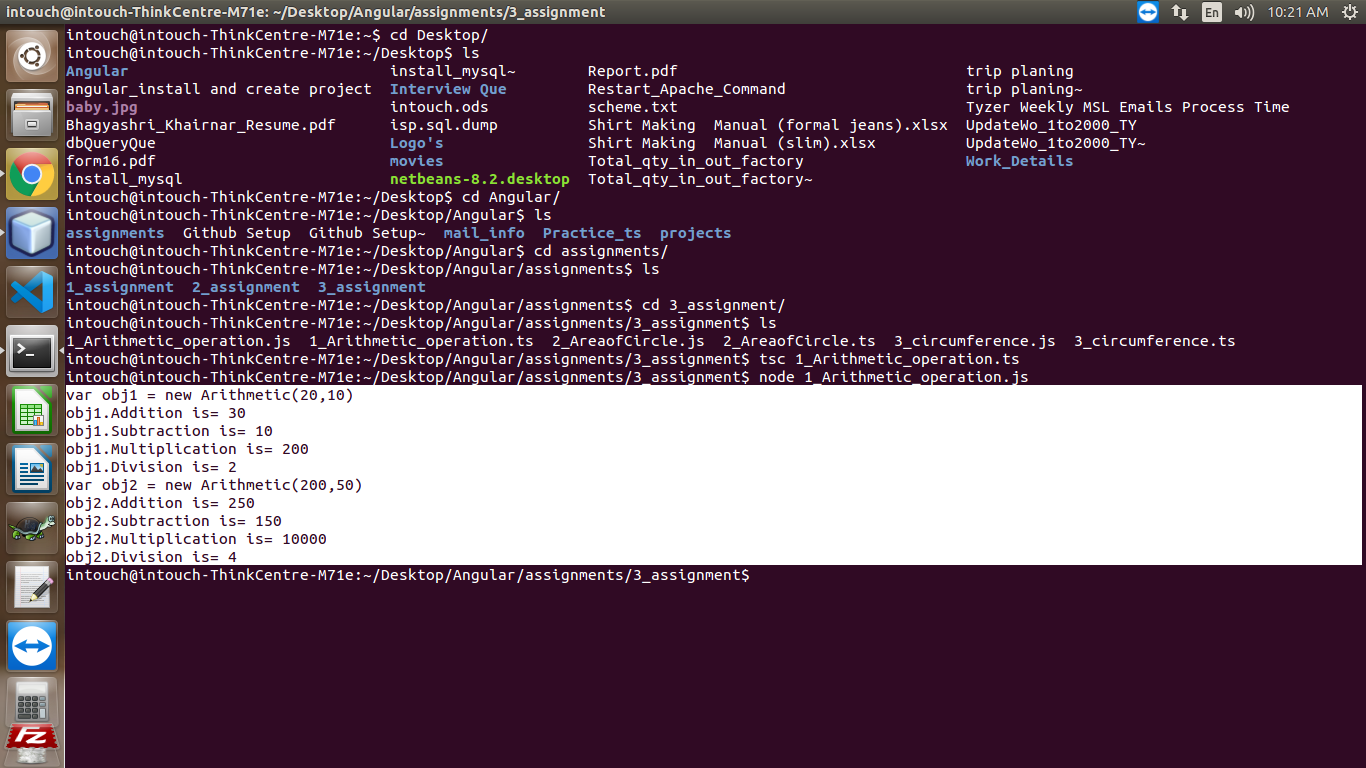
console.log("var obj2 = new Arithmetic(200,50)");

console.log("obj2.Addition is= "+obj2.Addition());

console.log("obj2.Subtraction is= "+obj2.Subtraction());

console.log("obj2.Multiplication is= "+obj2.Multiplication());

console.log("obj2.Division is= "+obj2.Division());



/\*2. Create one typescript application which contains one class named as Circle.

Circle class contains two characteristics (Class data members) as Radius, PI.

Create one parametrised constructor which accept one value and assign it to Radius. Value of

PI member is set to 3.14.

In Circle class we have to one method (Behaviours) as Area which will return area of Circle.

After designing the class create two objects of that class by providing some hardcoded value.

Call the method Area by using both the objects.

AUTHOR :- RAMESH GHULE

\*/

class Circle{

Radius:number;

PI:number;

constructor(rad:number){

this.Radius = rad;

this.PI = 3.14;

}

Area():number{

return this.PI\*this.Radius\*this.Radius;

}

}

var obj1 = new Circle(5);

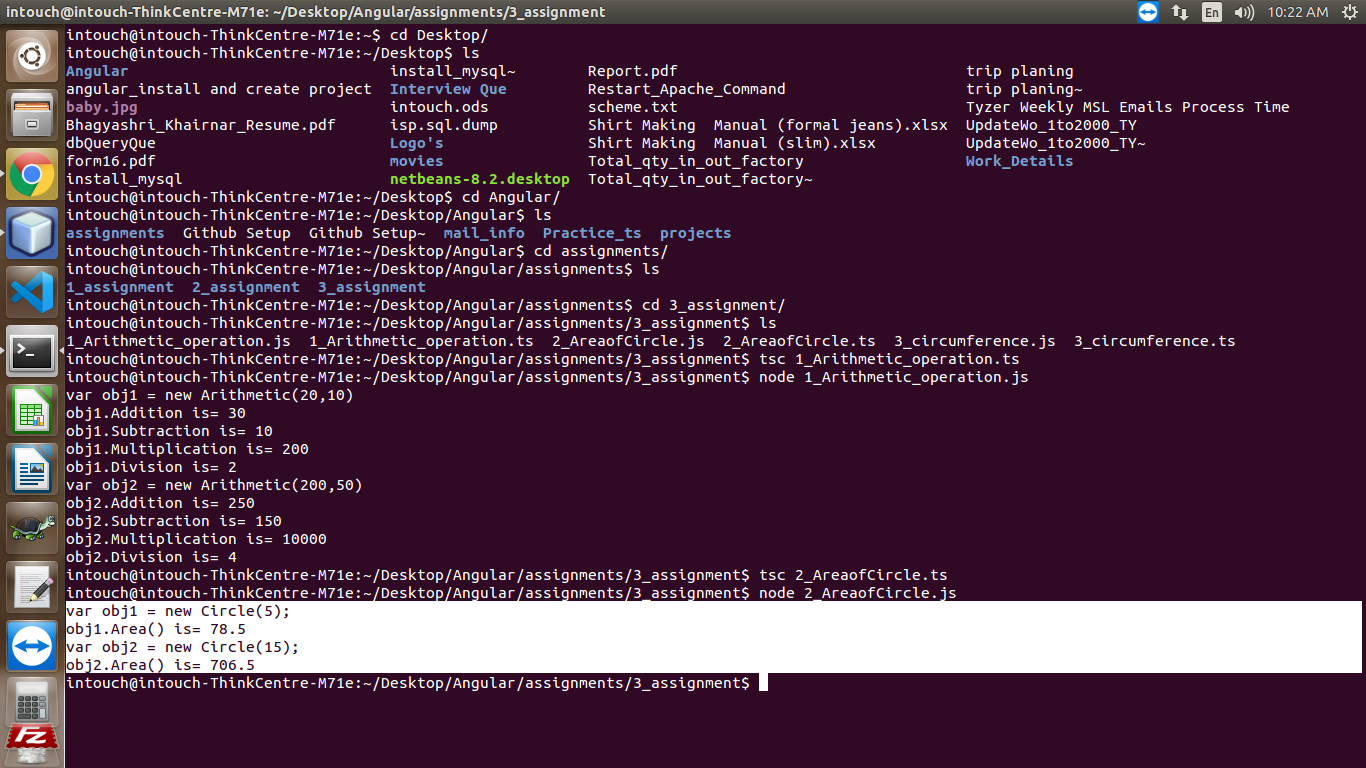
var obj2 = new Circle(15);

console.log("var obj1 = new Circle(5);");

console.log("obj1.Area() is= "+obj1.Area());

console.log("var obj2 = new Circle(15);");

console.log("obj2.Area() is= "+obj2.Area());



/\*3. Create one typescript application which contains one class named as CircleX which sill

inherits above Circle class.

In CircleX class we have to write one method (Behaviours) as Circumference which will return

circumference of circle.

After designing the class create two objects of that class by providing some hardcoded value.

Call Circumference and Area methods by using both the objects.

AUTHOR :- RAMESH GHULE

\*/

class Circle{

Radius:number;

PI:number;

constructor(rad:number){

this.Radius = rad;

this.PI = 3.14;

}

Area():number{

return this.PI\*this.Radius\*this.Radius;

}

}

class CircleX extends Circle{ //inheritance

constructor(rad:number){

super(rad);

}

Circumference():number{

return 2\*this.PI\*this.Radius;

}

}

var obj1 = new CircleX(5);

var obj2 = new CircleX(15);

console.log("var obj1 = new CircleX(5);");

console.log("obj1.Area() is= "+obj1.Area());

console.log("obj1.Circumference() is= "+obj1.Circumference());

console.log("var obj2 = new CircleX(15);");

console.log("obj2.Area() is= "+obj2.Area());

console.log("obj2.Circumference() is= "+obj2.Circumference());

