Example 1: Varibale Dec

var a:number;

var fristname:string;

fristname="Rajesh Patel";

console.log("Name is:"+fristname);

Example: 2 Class Dec

class MyClass

{

 public a:number;

 fristname:string="Rajesh Patel";

isTest:boolean=true;

}

var obj1=new MyClass();

console.log("Name is:"+obj1.fristname);

console.log(obj1.isTest);

Example-3 Variable Scope

var MyGlobalVariable=100;

class MyClass

{

 public instanceVariable:number=300;

 static myStaticVaribale=200;

}

console.log("Global Variable: "+MyGlobalVariable);

console.log("Static Variable: "+MyClass.myStaticVaribale);

var obj1=new MyClass();

console.log("Instance Variable: "+obj1.instanceVariable);

Example 4-Constant

class MyClass

{

    myMethod1()

    {

        const Num1=100;//Constant Variable

        //Num1=200;//Cannot reassign value ot constant variable

        return Num1;

    }

}

var obj1=new MyClass();

console.log("Constant Variable Value is: "+obj1.myMethod1());

Example 5: If Else

class MyClass

{

    a:number=-100;

    myMethod1()

    {

       if(this.a>0)

       {

        console.log("Number is positive");

       }

       else

       console.log("Number is Negative");

    }

}

var obj1=new MyClass();

console.log(obj1.myMethod1());

For Loop:

class MyClass

{

    myMethod1()

    {

      for(var i=0;i<=10;i++)

      {

        console.log(i);

      }

    }

}

var obj1=new MyClass();

console.log(obj1.myMethod1());

Example: Constructor

class MyClass

{

    Id:number;

    Name:string;

    constructor()

    {

        this.Id=101;

        this.Name="Test1";

    }

    myMethod1():void

    {

      console.log("Id: "+this.Id +" & "+"Name:"+this.Name);

    }

}

var obj1=new MyClass();

console.log(obj1.myMethod1());

class MyClass

{

    Id:number;

    Name:string;

    constructor(id:number,name:string)

    {

        this.Id=id;

        this.Name=name;

    }

    myMethod1():void

    {

      console.log("Id: "+this.Id +" & "+"Name:"+this.Name);

    }

}

var obj1=new MyClass(111,"Name1");

console.log(obj1.myMethod1());

Multiple Constructor:

class MyClass

{

    Id:number;

    Name:string;

    constructor(id:number);

    constructor(id:number,name:string);

    constructor(id:number,name?:string)

    {

        if(name)

        {

            this.Id=id;

            this.Name=name;

        }

        else

        {

            this.Id=id;

        }

    }

 }

let obj1=new MyClass(110);

console.log(obj1);

obj1=new MyClass(111,"Name1");

console.log(obj1);

Example: Property

class MyClass

{

    Id:number;

    Name:string;

    private \_salary:number;

    constructor(id:number,name:string)

    {

          this.Id=id;

          this.Name=name;

    }

    //getter

    get getSalary():number

    {

        return this.\_salary;

    }

    //setter

    set setSalary(salary:number)

    {

        this.\_salary=salary;

    }

    displayData():void{

        console.log("Id: "+this.Id+" Name: "+this.Name+" Salary: "+this.\_salary);

    }

 }

let obj1=new MyClass(110,"testName1");

obj1.setSalary=5000;

obj1.displayData();

Example:Inheritance

class MyClass

{

    \_id:number;

    \_name:string;

    constructor(id:number,name:string)

    {

          this.\_id=id;

          this.\_name=name;

    }

 }

 class MyClass1 extends MyClass

 {

    \_salary:number;

    constructor(id:number,name:string,salary:number)

    {

        super(id,name);

        this.\_salary=salary;

    }

 }

let obj1=new MyClass1(110,"testName1",12000);

console.log(obj1);

Interface:

interface ITest1

{

    dispaly();

}

class MyClass implements ITest1

{

    \_id:number;

    \_name:string;

    constructor(id:number,name:string)

    {

          this.\_id=id;

          this.\_name=name;

    }

    dispaly() {

        console.log("Emp Id: "+this.\_id+" & "+"Emp Name: "+this.\_name);

    }

 }

let obj1=new MyClass(110,"testName1");

obj1.dispaly();