

## Constructors and Constructor Overloading and Instance Methods:

```
=====
class MyTest
{
int rollno;
String sname;
int marks[]={10,20,30};
//int x1;
//create constructor
//default constructors
    MyTest()
    {
        rollno=1215;
        sname="Ashok";
        //marks[0]=10;
        //marks[1]=20;
        //marks[2]=30;
    }
//user defined cons
    MyTest(int a)
    {
        System.out.println("1-arg cons");
    }
    MyTest(int a,int b)
    {
        System.out.println("2-arg cons");
    }
    void myMethodTestArray()
    {
        int []x={10,20,30,40,50,60};
        System.out.println("Length of Array x="+x.length);
        for(int i=0;i<x.length;i++)
        {
            System.out.println(x[i]);
        }
    }

    void twoDArray()
    {
        int [][]x={{10,20,30},{40,50}};
    }
}
```

```

        System.out.println("Length of Array x="+x.length);
        for(int i=0;i<x.length;i++)
        {
            for(int j=0;j<x[i].length;j++)
            {
                System.out.println(x[i][j]);
            }
        }
    }

    void enhancedForTest()
    {
        //To print elements of one dimensional Array using enhanced for loop
        int []y={10,20,30,40,50,60};
        for(int x1:y)
        {
            System.out.println("Result of Enhanced for Loop");
            System.out.println(x1);
        }
    }

    void conditionalTest()
    {
        //ternary operator
        //?:
        //exp1=exp2?exp3:exp4;

        int exp=(10>20)?30:40;
        System.out.println("Ternary Operator usage");
        System.out.println(exp);
    }

    void display()
    {
        System.out.println("Rollno:"+rollno);
        System.out.println("Name:"+sname);
        System.out.println("Marks:"+marks[0]);
        System.out.println("Marks:"+marks[1]);
        System.out.println("Marks:"+marks[2]);
        //System.out.println("Marks:"+marks[3]);
    }

```

```

    }
    public static void main(String[] args)
    {
        MyTest t=new MyTest();

        //MyTest t1=new MyTest(60);
        //MyTest t2=new MyTest(40,50);
        //t1.MyTest(30);

        t.display();

        t.myMethodTestArray();
        t.twoDArray();
        t.enhancedForTest();
        t.conditionalTest();

    }
}

```

---

Example 2:

```

class Test

{
    String name;
    int rollno;
    float totalmarks;

    /* Case2:User is defined 0(zero-argument)constructor

```

OUTPUT:

Name=Yellaswamy

RollNo=1215

TotalMarks=50.0

problem:if we create object for Test for another students information we will get same info for another student also

output:

Name=Yellaswamy

RollNo=1215

TotalMarks=50.0

Name=Yellaswamy

RollNo=1215

TotalMarks=50.0

To overcome this problem parameterized constructor need to create

```
*/  
Test()  
{  
    name="Yellaswamy";  
    rollno=1215;  
    totalmarks=50;  
}
```

/\*case 3:

Parameterized constructor

if we are not use this key word we will get the following OUTPUT

Name=Yellaswamy

RollNo=1215

TotalMarks=50.0

Name=Yellaswamy

RollNo=1215

TotalMarks=50.0

Name=null

RollNo=0

TotalMarks=0.0

To overcome above problem this keyword we must use

After usage of this keyword we will get the following OUTPUT

```
Name=Yellaswamy
RollNo=1215
TotalMarks=50.0
Name=Yellaswamy
RollNo=1215
TotalMarks=50.0
Name=Ashok
RollNo=1216
TotalMarks=60.0
```

```
*/
```

```
Test(String name,int rollno,float totalmarks)
```

```
{
```

```
    this.name=name;
```

```
    this.rollno=rollno;
```

```
    this.totalmarks=totalmarks;
```

```
}
```

```
void display()
```

```
{
```

```
    System.out.println("Name="+name);
```

```
    System.out.println("RollNo="+rollno);
```

```
    System.out.println("TotalMarks="+totalmarks);
```

```
}
```

```
public static void main(String args[])
```

```
{
```

```
    //case 1 :There is no constructor defined inside a class.compile is generated no
argument constructor
```

```
    /*out put:
```

```
    Name=null
```

```
    RollNo=0
```

```
    TotalMarks=0.0
```

```
    */
```

```
    Test t1=new Test();
```

```
    t1.display();
```

```
    //Case 2
```

```
    Test t2=new Test();
```

```
    t2.display();
```

```
//case 3
Test t3=new Test("Ashok",1216,60);
t3.display();
```

```
}
```

```
}
```