

## Constructor

**Constructor:-** Constructor is used to initialize memory at run time.

### **Properties:-**

- i. It has same name as class name.
- ii. It has no return type.
- iii. It is invoked automatically when object is initialized.
- iv. Every class contains a default constructor. When a user creates one own constructor, then the default constructor's properties get destroyed.

### **Default constructor:-**

#### **Example:-**

```
class myclass
{
    public myclass()
    {
        System.out.print("this is default constructor\n");
    }
}

class ajay
{
    public static void main(String arg[])
    {
        myclass m=new myclass();
    }
}
```

```
new myclass();  
  
}  
  
}
```

### **Output:-**

```
this is default constructor  
this is default constructor
```

### **Parameter constructor:-**

#### **Example:-**

```
class myclass  
{  
    public myclass(int a)  
    {  
        System.out.print("This is parameter constructor "+a);  
        System.out.print("\n");  
    }  
}  
  
class ajay  
{  
    public static void main(String arg[])
```

```
{  
    myclass m=new myclass(6);  
    new myclass(2);  
}
```

### Output:-

```
This is default constructor 6  
This is default constructor 2
```

### Constructor overloading:-

#### Example:-

```
class myclass  
{  
    public myclass()  
    {  
        System.out.print("This is default constructor ");  
        System.out.print("\n");  
    }  
    public myclass(int a)
```

```
{  
    System.out.print("This is constructor "+a);  
    System.out.print("\n");  
}  
public myclass(int a,int b)  
{  
    int c=a+b;  
    System.out.print("Sum is "+c);  
  
}  
}  
class ajay  
{  
    public static void main(String arg[])  
    {  
        myclass m=new myclass();  
        new myclass();  
        myclass m1=new myclass(13);  
        new myclass(3,6);  
    }  
}
```

```
}
```

### Output:-

```
This is default constructor  
This is default constructor  
This is constructor 13  
Sum is 9
```

**Static keyword:-** It is used to create static variable as well as static function.

### Static variable:-

- i. It is followed help of static keyword.
- ii. It contain single memory location.
- iii. it is shared with all objects of particular class.
- iv. It is always assigned with zero when first object is initialized.

**Static fuction:-** It is also followed with the static keyword. It contains only static variable it is invoked with the help of class name.

### Example:-

```
class Bank  
{  
  
    private int accno;  
    private String name;  
    private float bal;  
    private static float bbal;
```

```
public void getinfo()
{
    java.util.Scanner x= new
    java.util.Scanner(System.in);
    System.out.print("Enter Customer Name ");
    name=x.next();
    System.out.print("Enter Customer Account
    number");
    accno=x.nextInt();
    System.out.print("Enter Customer Balance");
    bal=x.nextFloat();
    bbal=bbal+bal;
}
public void putinfo()
{
    System.out.print("Customer Name "+name);
    System.out.print("\nCustomer Account number
    "+accno);
    System.out.print("\nCustomer Balance "+bal);
}
```

```
public static void show()
{
    System.out.print("\nCustomer bank balace
"+bbal);
}
}

class ajay1
{
    public static void main(String arg[])
    {
        Bank m=new Bank();
        m.getinfo();
        m.putinfo();
        m.show();
    }
}
```

**Output:-**

Enter Customer Name sanny

//first time run

Enter Customer Account number 12345678

Enter Customer Balance 234

Customer Name sanny

Customer Account number 12345678

Customer Balance 234.0

Customer bank balace 234.0

Enter Customer Name ajay

//Second time run

Enter Customer Account number 12348

Enter Customer Balance 2319

Customer Name ajay

Customer Account number 12348

Customer Balance 2319.0

Customer bank balace 2319.0

**Object as an argument:-**

**Example:-** Write a program to put two time and find sum .

**Solution:-**

class time



```
{
    private int h,m,s;
    public void getinfo()
    {
        java.util.Scanner x= new java.util.Scanner(System.in);
        System.out.print("Enter Time with second ");
        h=x.nextInt();
        m=x.nextInt();
        s=x.nextInt();
    }
    public void putinfo()
    {
        System.out.print("\nTime="+h+":"+m+":"+s);
    }
    public void add(time t1,time t2)
    {
        h=t1.h+t2.h;
        m=t1.m+t2.m;
        s=t1.s+t2.s;
        m=m+s/60;
        s=s%60;
        h=h+m/60;
        m=m%60;
        h=h%60;
    }
}
```

```
}  
class objasarg  
{  
public static void main(String arg[])  
{  
time a=new time();  
time b=new time();  
time c=new time();  
a.getinfo();  
a.putinfo();  
b.getinfo();  
b.putinfo();  
c.add(a,b);  
c.putinfo();  
}  
}
```

**Output:-**

```
Enter time with second 04  
20  
18  
Time 4:20:18  
Enter time with second 05  
20  
20  
Time 5:20:20  
Time 9:40:38
```