

16. Non-Primitive Array

Syntax:

Classtype[] arrayname=new classtype [size];

Eg:

Employee[] e=new Employee[4];

Primitive array

Syntax

DataType arrayname=new datatype[size];

Eg:

int[] a=new int[3];

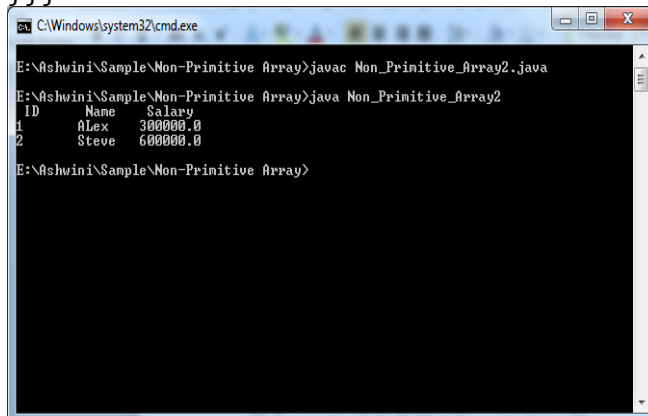
Example for printing a non-Primitive array:

// Non-primitive array example

```
class Employee
{
    int Id;
    String Name;
    double Salary;

    Employee( int a, String b, double c)
    {
        Id=a;
        Name=b;
        Salary=c;
    }
}

class Non_Primitive_Array2
{
    public static void main(String args[])
    {
        Employee e[]=new Employee[2];
        e[0]= new Employee(1, "ALex", 300000);
        e[1]= new Employee(2, "Steve", 600000);
        System.out.println(" ID \t Name \t Salary");
        for(int i=0;i<e.length;i++)
        {
            System.out.println(e[i].Id+"\t"+e[i].Name+"\t"+e[i].Salary);
        }
    }
}
```



```
C:\Windows\system32\cmd.exe
E:\Ashwini\Sample\Non-Primitive Array>javac Non_Primitive_Array2.java
E:\Ashwini\Sample\Non-Primitive Array>java Non_Primitive_Array2
ID      Name      Salary
1       ALex      300000.0
2       Steve     600000.0
E:\Ashwini\Sample\Non-Primitive Array>
```

2. Write a prog to display details of an employee whose salary is in btw 20000 to 400000

```
Employee( int a, String b, double c)
{
    Id=a;
    Name=b;
    Salary=c;
}

}

class Non_Primitive_Array3
{
    public static void main(String args[])
    {

        Employee e[]=new Employee[5];

        e[0]= new Employee(1, "ALex", 300000);
        e[1]= new Employee(2, "Steve", 400000);
        e[2]= new Employee(3, "John", 350000);
        e[3]= new Employee(4, "Dave", 600000);
        e[4]= new Employee(5, "Ava", 200000);

        System.out.println(" ID \t Name \t Salary");

        for(int i=0;i<e.length;i++)
        {

            if((e[i].Salary>=200000)&&(e[i].Salary<=400000))
            {
                System.out.println(e[i].Id+"\t"+e[i].Name+"\t"+e[i].Salary);
            }
        }
    }
}
```

3. Write a prog to create a product array and display only if product price=20000

4. Write a prog to count number of objects

```
class Employee
{

    int Id;
    String Name;
    double Salary;

    Employee( int a, String b, double c)
    {
        Id=a;
        Name=b;
        Salary=c;
    }

}

class Non_Primitive_Array5
```

```

{
public static void main(String args[])
{
int count=0;
Employee e[]=new Employee[2];

e[0]= new Employee(1, "ALex", 300000);
e[1]= new Employee(2, "Steve", 600000);

System.out.println(" ID \t Name \t Salary");

for(int i=0;i<e.length;i++)
{
count++;
}

System.out.println("Count of objects \t"+count);
}

}

```

Final Variable:

```

Class A
{
Static final int x; \error final variable should be initialised else it will throw an error
Public static void main()
{
System.out.println(x);
}
}

```

Static final variable can be initialised else compiler throws an error.
Static final variable can be initialise two places in a program:

1. At declaration
2. Inside static block

Non-static final variable can be initialised in three places:

1. At declaration
2. Inside non-static block
3. inside constructor

1. Write a prog to initialise final static and non-static variable at declaration, static block, non-static block and constructor.

```

class ClassObjects3
{
static final int a=10;
static final int b;
final int c=30;
final int d;
final int e;
static
{
b=20;
}
}

```

```

{
d=40;
}
ClassObjects3()
{
e=50;
}
public static void main(String args[])
{
ClassObjects3 c1=new ClassObjects3();
System.out.println(" Value of static declared value "+c1.a);
System.out.println(" Value of static declared value in static block "+c1.b);
System.out.println(" Value of static declared value"+c1.c);
System.out.println(" Value of static declared value in non-static block"+c1.d);
System.out.println(" Value of static declared value in constructor"+c1.e);
}
}

```

```

C:\Windows\system32\cmd.exe
E:\Ashwini\Sample\ClassObjects>java ClassObjects3
Value of static declared value 10
Value of static declared value in static block 20
Value of static declared value 30
Value of static declared value in non-static block 40
Value of static declared value in constructor 50
E:\Ashwini\Sample\ClassObjects>_

```

Final variable cannot be initialised inside the method