Decrement Operator: It Reduces the Value by 1.

```
public class MyClass2
{

	public static void main(String args[])
	{
		int a=11;
		System.out.println("a is:"+a);

		a--;//Decrement Operator--> a=a-1;
		System.out.println("a is:"+a);//a is 10

		--a;

		System.out.println("a is:"+a);// a is 9

		System.out.println("\n--------Pre-Decrement---------\n");

		int num1=85;
		int num2;

		System.out.println("num1 is:"+num1);

		num2=--num1;// num2=84, num1=84
		/*Pre-Decrement Operator First Decreases the Value by 1 then
		stores into another variable*/

		System.out.println("num2 is:"+num2);//84
		System.out.println("num1 is:"+num1);//84

		int num3;

		num3=--num2;//83

		System.out.println("num3 is:"+num3);//83
		System.out.println("num2 is:"+num2);//83

		System.out.println("\n--------Post-Decrement------\n");

		int value1=100;
```


```
        int value2;

        System.out.println("value1 is:"+value1);//100

                value2=value1--;// value2=100, value1=99

        /* Post Decrement Operator First Stores the Value in another
        variable then Decreases the Value by 1*/


        System.out.println("value2 is:"+value2);//100
        System.out.println("value1 is:"+value1);//99

        int value3;

        value3=value1--;//value1=98
        System.out.println("value3 is:"+value3);//99
        System.out.println("value1 is:"+value1);//98

    }

}
```

OutPut:

```
a is:11
a is:10
a is:9

-----------Pre-Decrement------------

num1 is:85
num2 is:84
num1 is:84

num3 is:83
num2 is:83
```

```
----------Post-Decrement------------

value1 is:100
value2 is:100
value1 is:99
value3 is:99
value1 is:98

*****************************************************
```

Relational Operator

int a = 10, b = 2 for all examples below

Operator	Meaning	Example	Result
==	Equals to	a == b	false
!=	Not equal to	a != b	true
<	Less than	a < b	false
<=	Less than or equal to	a <= b	false
>	Greater than	a > b	true
>=	Greater than or equal to	a >= b	true

Greater then equal to

p=23
q=5

23 22 9 6 5| 4>=5-->false

Less then equal to

0 1 3 5| 6<=5-->false

Pandu=Samsung, Pandi=Panasonic

int m=7; int n=17;

Pandu=Samsung == Pandi=Panasonic-->False

m=7 == n=17-->false

Pandu=Pansonic == Pandi=Panasonic-->true

m=17 == n=17-->true

NOT EQUAL TO !=

Pandu=Samsung != Pandi=Panasonic-->true

m=7 != n=17-->true

Pandu=Panasonic != Pandi=Panasonic-->false

m=17 != n=17-->false


```java
public class MyClass2
{

	public static void main(String args[])
	{

		System.out.println("Relational Operator");

		int p=23;
		int q=5;

		System.out.println("23 is greater then 5:"+(p>q));//Greater
Then Operator
		System.out.println("23 is less then 5:"+(p<q));//Less Then
Operator
		System.out.println("23 is greater the equal to 5:"+(p>=q));
//Greater then equal to
		System.out.println("20 is greater then equal to 5:"+(20>=q));
		System.out.println("5 is greater then equal to 5:"+(q>=q));
		System.out.println("4 is greater then equal to 5:"+(4>=q));
		System.out.println("0 is less then equal to 5:"+(0<=q));
		System.out.println("3 is less then equal to 5:"+(3<=q));
		System.out.println("7 is less then equal to 5:"+(7<=q));

		System.out.println("5 is Equal to 5:"+(q==q));
		// Equal to (ans is true only when values are same)
		System.out.println("23 is Equal to 5:"+(p==q));


		System.out.println("20 is Equal to 20:"+(20==20));
		System.out.println("3 is Equal to 20:"+(3==20));

		System.out.println("23 is Not Equal to 5:"+(p!=q));
		//!= Not Equal to(ans is true only when values are different)
		System.out.println("23 is Not Equal to 23:"+(p!=p));

	}

}
```

Relational Operator

23 is greater then 5:true
23 is less then 5:false
23 is greater then equal to 5:true
20 is greater then equal to 5:true
5 is greater then equal to 5:true
4 is greater then equal to 5:false
0 is less then equal to 5:true
3 is less then equal to 5:true
7 is less then equal to 5:false
5 is Equal to 5:true
23 is Equal to 5:false
20 is Equal to 20:true
3 is Equal to 20:false
23 is Not Equal to 5:true
23 is Not Equal to 23:false