

Relational Operators

<, <=, >, >=

1. We can apply relational operators for every primitive type except boolean.

Example:



```
int x = 3;
```

```
float y = 2.4f;
```

1. `System.out.println(3<2.4);` **// output: false**
2. `System.out.println('a'> 85);` **// output: true**
3. `System.out.println('a'<'b');` **// output: true**
4. `System.out.println(true>>false);` **// Compile time Error**

Relational Operators

2. We can not use relational operators for Object types.

Example:

1. `System.out.println("Rohit">"Roshan");` **// Compile time Error**

3. Nesting of relational operators for not allowed.

Example:

1. `System.out.println(25<26<27);` **// Compile time Error**

Equality Operators



1. We can apply equality operators for every primitive type including boolean.

Example:

1. `System.out.println(3 == 5.6);` **// output: false**
2. `System.out.println('a' != 85);` **// output: true**
3. `System.out.println('a' == 'b');` **// output: false**
4. `System.out.println(false == false);` **// output: true**

Equality Operators

2. We can use Equality operators for Object types.

Example:

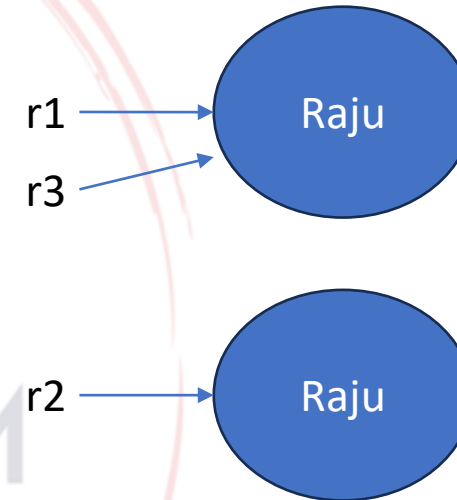
```
String r1 = new String("Raju");
```

```
String r2 = new String("Raju");
```

```
System.out.println(r1 == r2); // output: false
```

```
String r3 = r1;
```

```
System.out.println(r3 == r1); // output: true
```



Equality Operators

Difference between == operator and .equals() method.

Example:

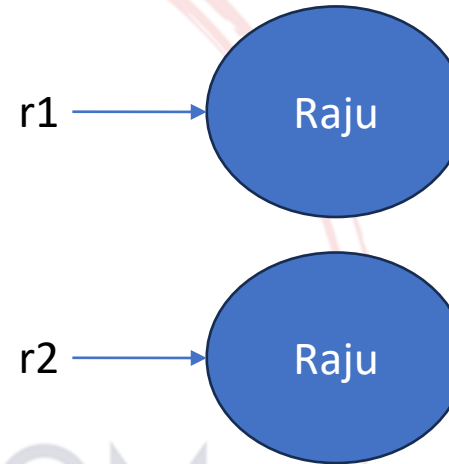
```
String r1 = new String("Raju");
```

```
String r2 = new String("Raju");
```

```
System.out.println(r1 == r2); // output: false
```

```
System.out.println(r1.equals(r2)); // output: true
```

hence, it is clear that '==' operator is use for refrence comparision and .equals() method use for value.



Equality Operators

Important :

To use the equality operators between object type compulsory there should be some relation between argument types (child to parent, parent to child), Otherwise we will get **Compiletime error incompatible types**.

Example:

```
String r1 = new String("Raju");
```

```
Thread r2 = new Thread();
```

```
System.out.println(r1==r2); // Compile time Error
```

```
Object o = new Object();
```

```
System.out.println(r1 == o); // output: false
```



CLASSROOM