Equality operator(== and !=)

We can apply equality opertors for every primitive type including boolean type also.

System.***out***.println(10==20);

System.***out***.println(**'a'** == **'b'**);

System.***out***.println(**'a'** == 97.0);

System.***out***.println(**false** == **false**);

We can apply equality operator for object types also for object references r1, r2

r1 == r2 returns true if any only if both references pointing to the same object(reference comparison or address comparison)

Thread t1 = new Thread();

Thread t2 = new Thread();

Thread t3 = t1;

sout(t1 == t2) //false

sout(t1 == t3) //true

If we apply equality operator for object types then compulsary there should be some relation between argument types either child to parent or parent to child or same type otherwise we will get compile time error saying in-comparable types.

Thread t1 = **new** Thread();

Object obj = **new** Object();

String str = **new** String();

System.***out***.println(t1 == obj);

System.***out***.println(str == obj);

System.***out***.println(str == t1); //CE: incomparable types: java.lang.String and java.lang.Thread

Difference between == operator and .equals method

In general we can use == operator for reference comparison(address comparison) and .equals method for content comparison

Note: For any object reference r,

r == null is always false

null == null is true