OBJECT EQUALITY IN JAVA

A SIMPLE RECTANGLE CLASS

```
public class MyRectangle {
                                         TWO MEMBER VARIABLES
   private int length;
                                         FOR THE LENGTH AND BREADTH
   private int breadth;
   public MyRectangle(int length, int breadth) {
       this.setLength(length);
       this.setBreadth(breadth);
                               A CONSTRUCTOR...
   public int getLength() {
       return length;
                                      ..AND GETTERS AND
   public void setLength(int length) {
       this.length = length;
                                      SETTERS
   public int getBreadth() {
       return breadth;
   public void setBreadth(int breadth) {
       this.breadth = breadth;
```

NOTHING TOO COMPLICATED.

NOW LET'S SAY WE INSTANTIATE TWO RECTANGLE OBJECTS WITH THE SAME LENGTH AND BREADTH

```
// two rectangle objects, with the same length and breadth
MyRectangle rectangle1 = new MyRectangle(5,10);
MyRectangle rectangle2 = new MyRectangle(5,10);

// are these two objects 'equal'
if(rectangle1 == rectangle2) {
    System.out.println("These two rectangles are 'equal'");
} else {
    System.out.println("These two rectanges are 'not equal'");
}
```

ARE THE TWO RECTANGLES EQUAL?

TO ANSWER THIS, WE NEED TO UNDERSTAND THE IDEA OF OBJECT IDENTITY IN JAVA

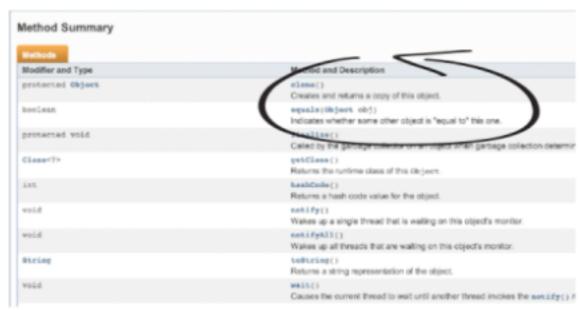
REMEMBER THAT ALL CLASSES IMPLICITLY DERIVE FROM THE OBJECT BASE CLASS

public class Object Class Object Class Object Class Object Class Object is the root of the class hierarchy. Every class has Object as a superclass. All objects, including arrays, implement the methods of this class Since: JDK1.9 See Also: Class Constructor Summary Constructor and Description Object [] Method Summary Method Summary Include: Modifier and Type protected Object Class of this object organic Object o

Class Object

THE OBJECT BASE CLASS HAS A METHOD CALLED ".EQUALS()" WHICH IS USED TO TEST IF TWO OBJECTS ARE EQUAL





BY DEFAULT, THIS
".EQUALS" METHOD
WILL ACTUALLY
ONLY RETURN TRUE
IF THE TWO OBJECTS
ARE LITERALLY THE SAME
OBJECT

("LITERALLY THE SAME":

OCCUPY THE SAME MEMORY
IN THE COMPUTER)

ARE THE TWO RECTANGLES EQUAL?

TO ANSWER THIS, WE NEED TO UNDERSTAND THE ANSWER IS NO! THE IDEA OF OBJECT IDENTITY IN JAVA

IF WE WOULD LIKE TWO RECTANGLES TO BE 'EQUAL' IF THEY HAVE THE SAME LENGTH AND BREADTH, WE WOULD NEED TO OVERRIDE THE EQUALS METHOD OF THE OBJECT BASE CLASS

NOW, WILL THE TWO OBJECTS BE CONSIDERED EQUAL? // two rectangle objects, MyRectangle rectangle1 =

STILL NO!

```
// two rectangle objects, with the same length and breadth
MyRectangle rectangle1 = new MyRectangle(5,10);
MyRectangle rectangle2 = new MyRectangle(5,10);

// are these two objects 'equal'
if(rectangle1 == rectangle2) {
    System.out.println("These two rectangles are 'equal'");
} else {
    System.out.println("These two rectanges are 'not equal'");
}
```

THAT'S BECAUSE THE "OPERATOR ALWAYS GOES WITH REFERENCE EQUALITY!

TO GET SEMANTIC EQUALITY, EXPLICITLY USE THE .EQUALS() METHOD

```
if rectangle1.equals(rectangle2) {
   System.out.println("These two rectangles are 'equal'");
} else {
   System.out.println("These two rectanges are 'not equal'");
}
```

NOW ARE THE TWO OBJECTS EQUAL?

FINALLY - YES!