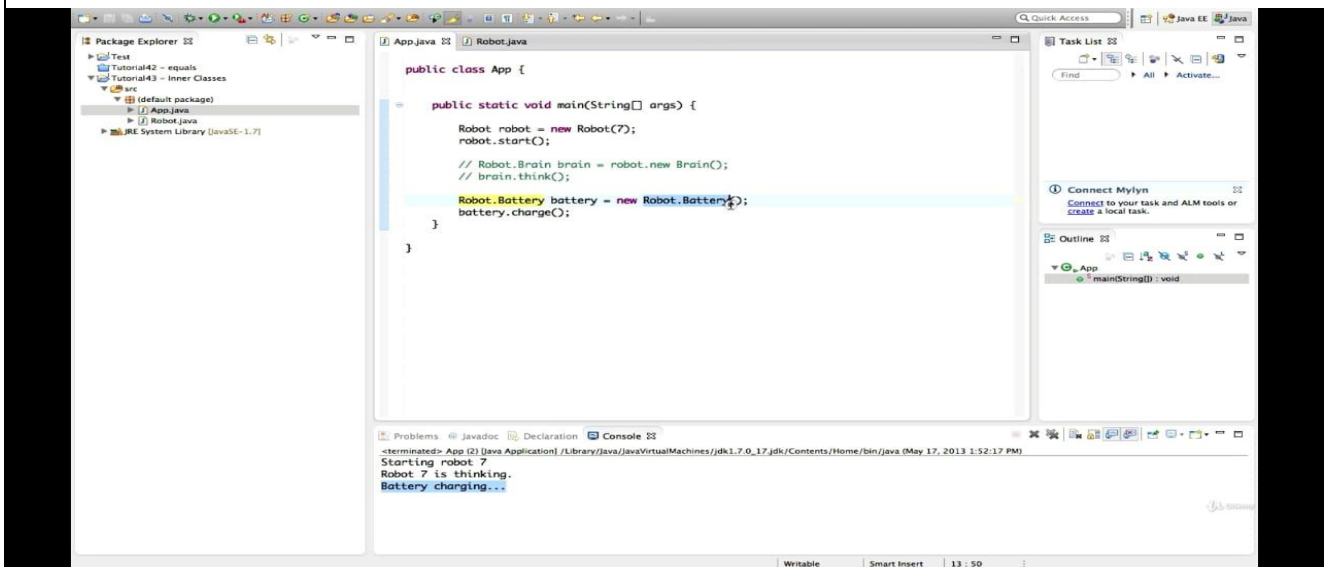
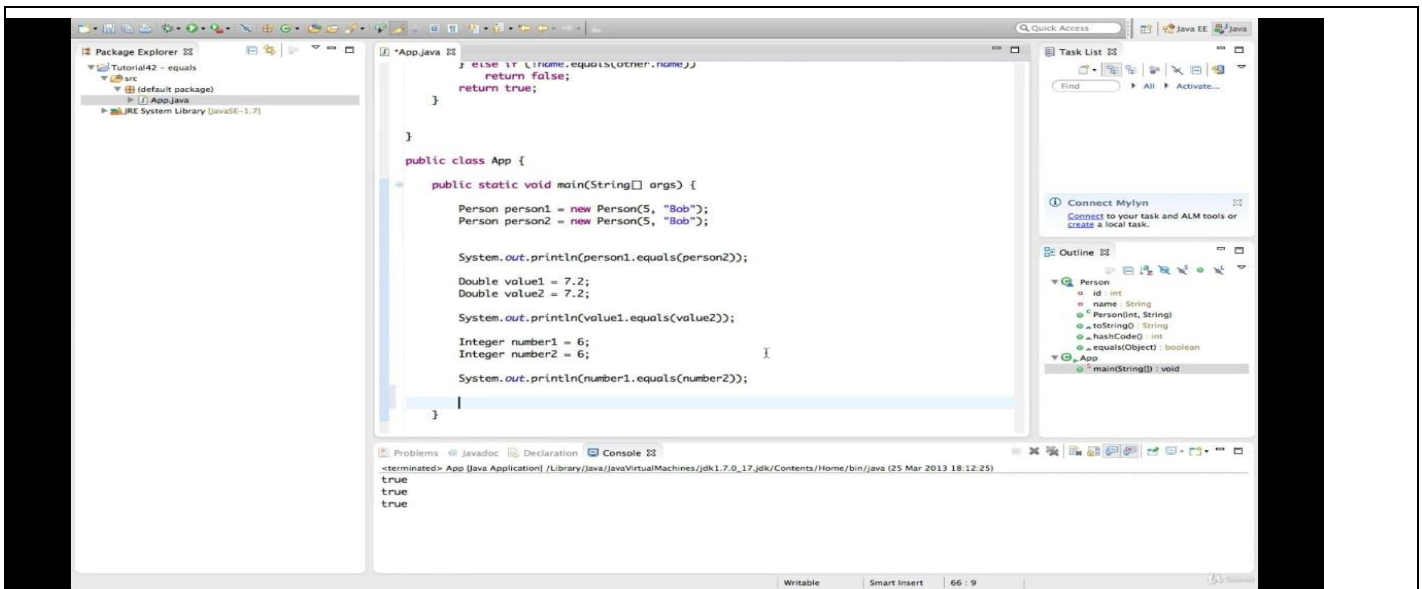


JAVA REPORT

Date:	13/06/2020	Name:	SAFIYA BANU
Course:	JAVA	USN:	4AL16EC061
Topic:	<ol style="list-style-type: none">1. The Equals Method2. Inner Classes3. Enum Types: Basic and Advanced Usage4. Recursion: A Useful Trick Up Your Sleeve5. Serialization: Saving Objects to Files6. Serializing Arrays7. The Transient Keyword and More Serialization8. Passing by Value	Semester & Section:	8 TH B
Github Repository:	Safiya-Courses		



The method determines whether the Number object that invokes the method is equal to the object that is passed as an argument.

Syntax

public boolean equals(Object o)

Parameters

Here is the detail of parameters –

- Any object.

Return Value

- The method returns True if the argument is not null and is an object of the same type and with the same numeric value. There are some extra requirements for Double and Float objects that are described in the Java API documentation.

Example

```
public class Test {  
  
    public static void main(String args[]) {  
        Integer x = 5;  
        Integer y = 10;  
        Integer z = 5;  
        Short a = 5;  
  
        System.out.println(x.equals(y));  
        System.out.println(x.equals(z));  
        System.out.println(x.equals(a));  
    }  
}
```

Nested Classes

In Java, just like methods, variables of a class too can have another class as its member. Writing a class within another is allowed in Java. The class written within is called the nested class, and the class that holds the inner class is called the outer class.

Syntax

Following is the syntax to write a nested class. Here, the class Outer_Demo is the outer class and the class Inner_Demo is the nested class.

```
class Outer_Demo {  
    class Inner_Demo {  
    }  
}
```

Nested classes are divided into two types –

- Non-static nested classes – These are the non-static members of a class.
- Static nested classes – These are the static members of a class.

Java Transient Keyword

Java transient keyword is used in serialization. If you define any data member as transient, it will not be serialized.

Let's take an example, I have declared a class as Student, it has three data members id, name and age. If you serialize the object, all the values will be serialized but I don't want to serialize one value, e.g. age then we can declare the age data member as transient.

Example of Java Transient Keyword

In this example, we have created the two classes Student and PersistExample. The age data member of the Student class is declared as transient, its value will not be serialized.

Object references are passed by value

All object references in Java are passed by value. This means that a copy of the value will be passed to a method. But the trick is that passing a copy of the value also changes the real value of the object. To understand why, start with this example:

```
public class ObjectReferenceExample {
```

```
public static void main(String... doYourBest) {  
    Simpson simpson = new Simpson();  
    transformIntoHomer(simpson);  
    System.out.println(simpson.name);  
}  
  
static void transformIntoHomer(Simpson simpson) {  
    simpson.name = "Homer";  
}  
  
}  
  
class Simpson {  
    String name;  
}
```

What do you think the simpson.name will be after the transformIntoHomer method is executed?

In this case, it will be Homer! The reason is that Java object variables are simply references that point to real objects in the memory heap. Therefore, even though Java passes parameters to methods by value, if the variable points to an object reference, the real object will also be changed.