Object Class:
==============

1. It is the super most class in the entire java hierarchy.
2. It is available in java.lang package.
3. It has no argument constructor only.

```
Object o1 = new Object();
```

Methods of Object Class:
===========================

```
public String toString()
public int hashCode()
public boolean equals(Object obj)
public final void wait()
public final void wait(long ms)
public final void wait(long ms,iny ns)
public final void notify()
public final void notifyAll()
public class getClass()
protected void finalize()
protected Object clone()
```

Overridden Methods:
=====================

1. public String toString():
=============================

It returns string representation of an object. String representation includes fully qualified name of the class along with the hashcode in hexa-decimal in the following format

packageName.className@hashCode in hexa-decimal

Usually we override this method in the subclass to display states of an object (contents of an object).

2. public int hashCode():
============================

HashCode is a unique integrated integer number associated with an object. This method generates the hashcode based on hexa-decimal address.

Usually we override this method in the subclass to generates the hashcode based on the unique attribute of an object.

3. public boolean equals(Object obj):
=======================================

This method is used to compare the current object with the given object based on the hashcode. If two objects are having same hashcode then it returns true otherwise false.

Usually we override this method in the subclass to compare the current object with the given object based on the state of the object.

Note: When we use reference inside the System.out.println() internally it calls toString().

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

1. equals operator is used to compare two references based on actual addresses.

2. equals() method default implementation is still to compare objects based on actual addresses.

3. If we do not override equals() method then there is no difference between equals operator and equals() method.

4. Its recommended practice to override equals method and hashcode method together.

getClass():
============

It returns an object of Class class. Using that object we can invoke getName() method which returns the fully qualified name of the object.

```
class Class
{
    getName()
    {

    }
}
class Object
{
    public Class getClass()
    {
        return new Class();
    }
}

-----------------------------------------------------------------
class Employee
{

}
class Test
{
    public static void main(String[] args)
    {
        Employee e = new Employee();


        Class c = e.getClass();
        String s = c.getName();
        System.out.println(s);

            or

        System.out.println(e.getClass().getName());

    }
}
```