Libraries

Package is a collection of classes and interface. There are of two types of packages:
1. User defined libraries
2. Pre-defined/ inbuilt libraries

User defined packages:

The packages which are created by user is known as user defined packages.

Pre-defined/inbuilt packages:

Packages provided by language developers are called as pre-defined developers.
Eg: java.lang,java.Io, java.sql

The classes/interfaces belongs to java.lang packages can be accessed directly without import statements.

JDK->Src->Java

Object class

Object is a class in java. Object class is defined in java.lang package. Object is a root class for all the classes, hence object class properties will be available in all the classes.

Object class methods:

1. Public String toString();
2. Public int hashCode();
3. public Boolean equals();
4. public void wait();
5. public void wait(long arg);
6. public void notify();
7. public void notifyAll();
9. protected void finalize();
10. protected object clone();

Public String toString()

```
        public class LibrarytoString
{

}
public class LibrariesMain2 {
        public static void main(String[] args)
        {
                LibrarytoString l=new LibrarytoString();
                String s=l.toString();
                System.out.println(s);
        }
}
```

```
Libraries2.LibrarytoString@15db9742
```

Package.class@hexdecimalvalue

Fully qualified class name is printed if only object reference is printed.

Overriding toString()

```
public class Student
{
        int id;
        String name;
        double marks;

        Student(int id, String name, double marks)
        {
                this.id=id;
                this.name=name;
                this.marks=marks;
        }

        @Override
        public String toString()
        {
                return "id:"+id+"\n Name:"+name+"\n Marks:"+marks+" ";
        }

}
public class LibrariesMain1
{
        public static void main(String[] args)
        {
                Student s=new Student(10,"Alex",65.5);
                String str=s.toString();
                System.out.println(str);
        }

}
```

toString method is declared in object class. toString method returns string values. The value contains fullyqualified class name and hexadecimal address value generated by JVM.
toString can be overridden to change implementation of the method.

Public int hashCode()

Hashcode is non-static method declared in object class, Hascode method returns unique integer number, generated by JVM. Hashcode number is a hexadecimal address of object converted into an integer.

```
package Libraries5;

public class LibrarieMain5
{
        public static void main(String[] args)
        {
        LibraryHashCode l=new LibraryHashCode();
        int x=l.hashCode();
        System.out.println(x);
        System.out.println(l);
        }
}
public class LibraryHashCode
{

}
```

Overriding public int hashCode()

```
public class LibHashCodeOverride
{
        int x;
        public int hashCode()
        {
                return x;
        }
}
public class LibrariesMain6 {

        public static void main(String[] args)
        {
        LibHashCodeOverride l=new LibHashCodeOverride();
        l.x=200;
        int y=l.hashCode();
        System.out.println(y);
        }
}
```

Public Boolean equals(object obj)

Equal method is used to compare two objects. Equals method compare two object based on hashcode equals can be overridden to compare two object based on its states.

```
public class LibrarieMain7 {
        public static void main(String[] args)
        {
                LibrarieEqual7 l1=new LibrarieEqual7();
                LibrarieEqual7 l2=new LibrarieEqual7();

                boolean y=l1.equals(l2);
                System.out.println(y);
        }
}
public class LibrarieEqual7
{

}
```

Overriden method for hashcode:

```
public class LibraryEquals
{
        String brand;
        double price;

        LibraryEquals(String brand,double price)
        {
```

```
		this.brand=brand;
		this.price=price;
	}
	
	@Override
	public boolean equals(Object obj)
	{
		LibraryEquals l=(LibraryEquals) obj;
		return this.price==l.price;
	}
}

public class LibrarieMain3 {

	public static void main(String[] args)
	{
		LibraryEquals l=new LibraryEquals("Sony",9899.098);
		LibraryEquals l1=new LibraryEquals("Sony",9899.098);
		
		Boolean x=l.equals(l1);
		System.out.println(x);
	}

}
```