Chapter 12 - Packages

Interpreter vs Compiler

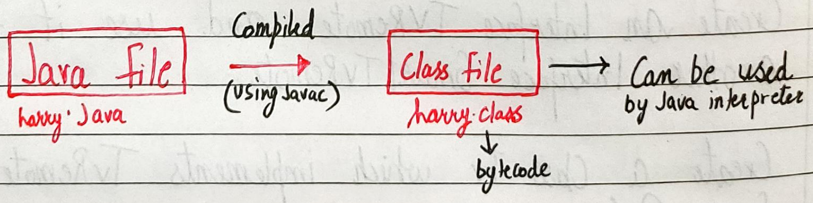
Interpreter translates one statement at a time into machine code.

Compiler scans the entire program and translates whole of it into machine code.

Interpreter	Compiler
* One statement at a time	* Entire program at a time
* Interpreter is needed everytime	* Once compiled it is not needed
* Partial execution if error	* No execution if an error occurs
* Easy for programmers	* Usually not as easy as Interpreted ones

Is Java Compiled or Interpreted?

Java is a hybrid language → both compiled as well as interpreted



→ A JVM can be used to Interprete this bytecode

→ This bytecode can be taken to any platform (Win/Mac/Linux) for execution

→ Hence Java is platform independent (write once run everywhere)

Before executing the program save all the files. If you make any changes then also save the file after making changes.

Executing a Java Program

Javac Harry.Java → Compiled
Java Harry. → Interprcted

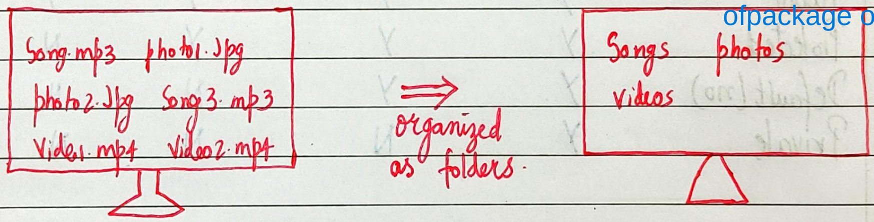
So fare the execution of our program was being managed by intelly Idea.
We can download a Source code editor like VS Code to Compile & execute our Java programs.

Packages in Java

A package is used to group related classes.
Packages help in avoiding name conflicts.
There are two types of packages:

* Built in packages → Java API
* User defined packages → Custom packages

package nameofpackage;
for subpackages
package name.name
ofpackage ofsubpac.



1.class this.Java my.mp3 ⇒ organized as packages
Song.Java harry.Java

Using a Java package

import java.lang.* → import everything from Java.lang
import java.lang.String → import String from java.lang
s = new Java.lang.String("Harry") → Use without importing

Similarly you can use java.util package which has classes like Scanner class and random class...

using javac -d . *.java we can create all the class files in respective packages.

Creating a package

Javac Harry.java → Creates Harry.class

javac -d . Harry.java → creates a package folder

↳ We can keep adding classes to a package like this

We can also create inner packages by adding "package.inner" as package name

folder

subfolder

These packages once created can be used by other classes.

Access Modifiers in Java

Access modifiers determine whether other classes can use a particular field or invoke a particular method Can be public, private, protected or default (no modifier)

applicable for all - classes, methods, and attributes

Modifier	Class	Package	Subclass (in diff. packages)	World
Public	Y	Y	Y	Y
Protected	Y	Y	Y	N
Default (no)	Y	Y	N	N
Private	Y	N	N	N

How to use a package:

import packagename.classname;

// Now in main method make the object of the class imported
//and use can run methods using object.