

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



- * One statement at a time
- * Interpreter is needed everytime
- * Partial execution if error
- * Easy for programmers

Compiler



- * Entire program at a time
- * Once compiled it is not needed
- * No execution if an error occurs
- * Usually not as easy as Interpreted ones

Is Java Compiled or Interpreted?

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

Java file
Harry.java

Compiled

→
(using javac)

Class file
Harry.class

↓
bytecode

→ Can be used
by Java interpreter

- A JVM can be used to Interpret this bytecode
- This bytecode can be taken to any platform (win/Mac/Linux) for execution
- Hence Java is platform independent (write once run everywhere)

Executing a Java Program

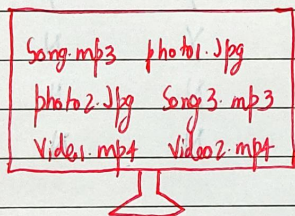
javac Harry.java → Compiled
java Harry.class → Interpreted

So far the execution of our program was being managed by IntelliJ Idea.
We can download a source code editor like VS Code to compile & execute our Java programs.

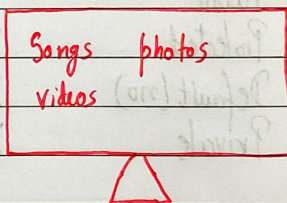
Packages in Java

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



⇒
organized
as folders.



1. class this.java my.mp3 ⇒
Song.java Harry.java organized
as packages

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

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
 These packages once created can be used by other classes.

↓
folder

↓
subfolder

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)

Modifier	Class	Package	Subclass	World
Public	Y	Y	Y	Y
Protected	Y	Y	Y	N
Default (no)	Y	Y	N	N
Private	Y	N	N	N