# Chapter 12 - Packages

## Interpreter vs Compiler

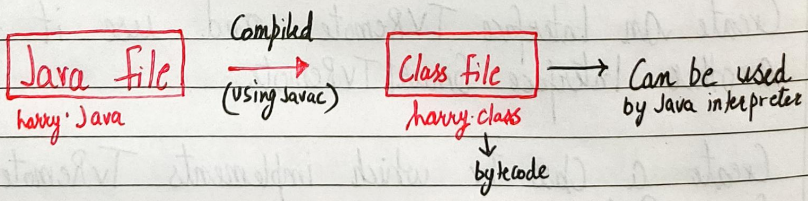
Interpreter translates one statement at a time into machine code.

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)

URBAN EDGE

# Executing a Java Program

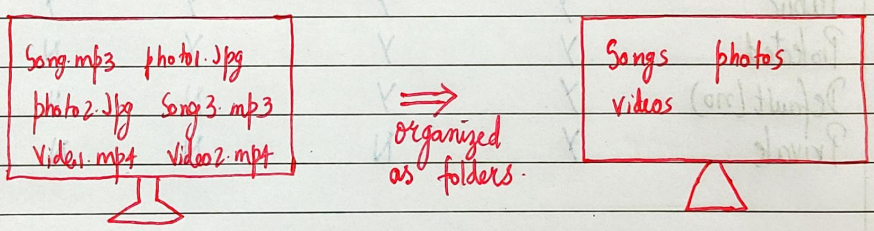
Javac Harry.java → Compiled
java Harry.class → Interpreted (syntax:-java class_name)

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




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

URBAN EDGE

## 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 (package → folder, inner → 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)

| 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 |