Chapter 11: Classes and Packages

Exercises :

1. Try importing Java libraries in Scala—e.g., Java's date libraries—and try using them in your program.
2. If you are using Linux, try executing the shell script from your Scala program. You will have to import a specific package to do so.
3. Research the maven repository. What is it, what does it contain, and how do Scala and Java programmers use it?

Answers :

- **Package**

Package in Scala is a mechanism to encapsulate a group of classes, sub packages, traits and package objects. It basically provides namespace to put our code in a different files and directories. Packages is a easy way to maintain our code which prevents naming conflicts of members of different packages. Providing access control to members of packages like private, protected, package specific controlling scope restricts the access of member to other packages, while the members with no modifier can be used inside any other package with some referenc

- **Maven**

Maven is a popular open-source build tool developed by the Apache Group to build, publish, and deploy several projects at once for better project managment, The tool provides allows developers to build and document the lifecycle framework.

Maven is written in Java and is used to build projects written in C#, Scala, Ruby, etc. Based on the Project Object Model (POM), this tool has made the lives of Java developers easier while developing reports, checks build and testing automation setups.

Maven focuses on the simplification and standardization of the building process, taking care of the following:

Builds

Documentation

Dependencies

Reports

SCMs

Distribution

Releases

Mailing list

Maven was created to simplify Jakarta Tribune project building processes. Many of the projects had slightly different ANT files, so Apache developed Maven to handle building multiple projects together, including publishing project information, facilitating team collaboration, deploying projects, and sharing JARS among several projects.

Maven's purpose is to provide developers with:

- A comprehensive, maintainable, reusable, and simple model for projects
- A set of tools and plug-ins that can interact with the declarative model.

Maven repositories

Maven repositories refer to the directories of packaged JAR files that contain metadata. The metadata refers to the POM files relevant to each project. This metadata is what enables Maven to download dependencies.

There are three types of repositories:

- Local repository refers to the developer's machine, which is where all the project material is saved. This repository contains all the dependency jars.
- The remote repository refers to the repository present on a web server, which is used when Maven needs to download dependencies. This repository works the same as the central repository. Whenever anything is required from the remote repository, it is first downloaded to the local repository, and then used.
- Central repository refers to the Maven community that comes into action when there is a need for dependencies, and those dependencies cannot be found in the local repository. Maven downloads the dependencies from here whenever needed.