**Assignment 15.2**

**Working of Maven:**

**Maven** is build automation tool used primarily for Java projects. Maven addresses two aspects of building software: first, it describes how software is built, and second, it describes its dependencies. Contrary to preceding tools like Apache Ant, it uses conventions for the build procedure, and only exceptions need to be written down. An XML file describes the software project being built, its dependencies on other external modules and components, the build order, directories, and required plug-ins. It comes with pre-defined targets for performing certain well-defined tasks such as compilation of code and its packaging. Maven is built using a plugin-based architecture that allows it to make use of any application controllable through standard input. Theoretically, this would allow anyone to write plugins to interface with build tools (compilers, unit test tools, etc.) for any other language. In reality, support and use for languages other than Java has been minimal. Currently a plugin for the .NET framework exists and is maintained, and a C/C++ native plugin is maintained for Maven 2.

**Working of Gradle:**

Gradle is an advanced general purpose build management system based on Groovy and Kotlin. Gradle supports the automatic download and configuration of dependencies or other libraries. It supports Maven and Ivy repositories for retrieving these dependencies. This allows reusing the artifacts of existing build systems.

Gradle supports multi-project and multi-artifact builds.

Gradle has the notion of projects and tasks.

**Working of SBT:**

SBT is an open source build tool for Scala and Java Projects, similar to Java’s Maven or Ant. Its main features are native support for compiling Scala code and integrating with many Scala test frameworks.

**Difference between Maven,Gradle and SBT:**

SBT is very simple and it is focused on Scala it relies on Ivy for dependency management. Maven it's a great build tool and it enables to control the entire software lifecycle with XML files. Using the Project Object Model you can intercept all points of the software lifecycle from compile to test, packaging and deploy. Maven has its own dependency manager. Maven is the XML syntax, writing a POM can be annoying and too much expensive. Gradle is built on top of Maven, Ant and Ivy. It uses Maven repositories. Gradle does not use XML, it's a polyglot build tool. It combines the Ant API with the Groovy language to enable developers to write a build script with an intuitive DSL. With a few lines of code we can write a Gradle build script that can do the same things that Maven can do. With Gradle we can define your own task with the Groovy language and intercept programmatically your build execution. This functional approach is not for all developers, infact Maven it's good if you don't want this behavior in your build environment. Both Maven and Gradle have plugins to integrate your build with technologies used in your projects.