Gradle Repository

Sometimes dependencies come in the form of modules. We have to specify the location of modules so that the build can consume them. The **location for storing modules** is called a repository. By specifying the repositories for a project, Gradle can find and retrieve modules. Repositories can be in different forms, such as a local directory or a remote repository.

At runtime, Gradle will discover the declared dependencies required for operating a specific task. Once a dependency is resolved, the resolution mechanism stores the essential files of dependency in the local cache memory also stated as the dependency cache. Future builds reuse the files saved in the cache to skip unnecessary network calls.

Gradle supports the following repository formats:

* Ivy repositories
* Maven repository
* Flat directory repository

Adding repositories to Gradle project

We can add different repository formats to our project by adding that particular code snippet to the **'build.gradle'** file.

Adding Ivy repository to Gradle Build

We can add an Ivy repository to our project by declaring its url address or its location in the 'build.gradle' file.

To add an Ivy repository by using its url address, add the following code snippet to the **'build.gradle'** file:

1. repositories {
2. ivy {
3. url 'http://ivy.petrikainulainen.net/repo'
4. }
5. }

To add an Ivy repository by using its location in the file system, add the following code snippet to the **'build.gradle'** file:

1. repositories {
2. ivy {
3. url '../ivy-repo'
4. }
5. }

Adding Maven Repositories to Gradle Build

We can add a Maven repository to our project by using its url address or its location by adding that particular code snippet in our Gradle build script.

To add a Maven repository by using its url, add the following code snippet to the **'build.gradle'** file:

1. repositories {
2. maven {
3. url 'http://maven.petrikainulainen.net/repo'
4. }
5. }

To add a Maven repository by using its location in the file system, add the following code snippet to the 'build.gradle' file:

1. repositories {
2. maven {
3. url '../maven-repo'
4. }
5. }

The "aliases" in Gradle are used in case of adding Maven repositories to our project build. These aliases are as following:

**The mavenCentral():** This alias stands for the dependencies that are fetched from the central Maven 2 repository.

**The jcenter():** This alias stands for the dependencies that are fetched from the Bintray's JCenter Maven repository.

**The mavenLocal():** This alias stands for the dependencies that are fetched from the local Maven repository.

To add the central Maven repository in our project, add the following code snippet to our **'build.gradle'** file:

1. repositories {
2. mavenCentral()
3. }

Adding Flat Directory Repository to our build

To use a flat directory repository, add the below code snippet to **'build.gradle'** file:

1. repositories {
2. flatDir {
3. dirs 'lib'
4. }
5. }

As we have described directory as **'lib'**, this means that dependencies are searched from the lib directory. Also, we can specify multiple directories; to do so, add the following code snippet to **'build.gradle'** file:

1. repositories {
2. flatDir {
3. dirs 'libA', 'libB'
4. }
5. }