```
##################
Sonatype Nexus
##################

-> Nexus is an Open Source Software

-> It is an Artifact Repository Server

-> It is used to store and retrieve build artifacts

-> Nexus software developed using Java

Note: To install Nexus s/w we need to install java first

-> Currently people are using Nexus 3.x


Java : jar, war and ear

Docker : Docker images

Node JS : NPM package


Q) What is difference between Nexus and GitHub ?

-> Github is a SCM software which is used to store source code of the project

-> Nexus is Artifact Repository which is used to store build artifacts

##################
Nexus Setup
##################

-> Take t2.medium instance

-> Java s/w is required to install Nexus

-> Connect to t2.medium instance using mobaxterm


#  Nexus S/w Installation Process in Linux OS

$ sudo su -

$ cd /opt

# install java 1.8v
$ sudo yum install java-1.8.0-openjdk

Links to download : https://help.sonatype.com/repomanager3/product-information/download

# latest version
$ wget https://download.sonatype.com/nexus/3/nexus-3.40.1-01-unix.tar.gz

$ tar -zxvf nexus-3.40.1-01-unix.tar.gz

$ mv /opt/nexus-3.40.1-01 /opt/nexus


#As a good security practice, Nexus is not advised to run nexus service as a root user, so create a
new user called nexus and grant sudo access to manage nexus services as follows.

$ useradd nexus
```

```
#Give the sudo access to nexus user

$ visudo
nexus ALL=(ALL) NOPASSWD: ALL

#Change the owner and group permissions to /opt/nexus and /opt/sonatype-work directories.

$ chown -R nexus:nexus /opt/nexus
$ chown -R nexus:nexus /opt/sonatype-work
$ chmod -R 775 /opt/nexus
$ chmod -R 775 /opt/sonatype-work

# Open /opt/nexus/bin/nexus.rc file and  uncomment run_as_user parameter and set as nexus user.

$ vi /opt/nexus/bin/nexus.rc
run_as_user="nexus"

# Create nexus as a service

$ ln -s /opt/nexus/bin/nexus /etc/init.d/nexus

# Switch as a nexus user and start the nexus service as follows.

$ su - nexus

# Enable the nexus services
$ sudo systemctl enable nexus

# Start the nexus service
$ sudo systemctl start nexus

#Access the Nexus server from Laptop/Desktop browser.

URL : http://IPAddess:8081/

Note: Enable this 8081 port number in Security Group

# Default Username
User Name: admin

# we can copy nexus password using below command
$ sudo cat /opt/sonatype-work/nexus3/admin.password


-> We can change nexus default properties

        /opt/nexus/etc/nexus.properties


####################################
Integrate Maven with Nexus
####################################

-> Create Repositories in Nexus to store build artifacts

-> We will create 2 types of repositories in Nexus

        1) snapshot

        2) release

-> If project is under development then that project build artifacts will be stored into snapshot
repository
```

-> If project development completed and released to production then that project build artifacts will be stored to release repository

Snanpshot Repo URL : http://13.233.238.64:8081/repository/ashokit_snapshot_repo/

Release Repo URL : http://13.233.238.64:8081/repository/ashokit_release_repo/

Note: Based on <version/> name available in project pom.xml file it will decide artifacts should be stored to which repository

-> Nexus Repository details we will configure in project pom.xml file like below

```
  <distributionManagement>
        <repository>
                <id>nexus</id>
                <name>Ashok IT Releases Nexus Repo</name>
                <url>http://15.207.19.102:8081/repository/ashokit-release-repository/</url>
        </repository>

        <snapshotRepository>
                <id>nexus</id>
                <name>Ashok IT Snapshots Nexus Repo</name>
                <url>http://15.207.19.102:8081/repository/ashokit-snapshot-repository/</url>
        </snapshotRepository>
</distributionManagement>
```

-> Nexus Server Credentials will be configured in Maven "settings.xml" file

-> Maven Location : C:\apache-maven-3.8.5\conf

-> In settings.xml file, under <servers> tag add below <server> tag

```
        <server>
                <id>nexus</id>
                <username>admin</username>
                <password>admin@123</username>
        </server>
```

-> Once these details are configured then we can run below maven goal to upload build artifacts to Nexus Server

```
        $ mvn clean deploy
```

##################
Remote Repository
##################

-> Remote repository used for shared libraries

-> If we want to use few jar files in multiple projects in the company then we will use Remote Repository

-> Remote repository is specific to our company projects

-> Create remote repo in nexus and upload a jar file

-> Take dependency details of uploaded jar file and add in project pom.xml as a dependency

-> We need to add Remote Repository Details in pom.xml above <dependencies/> tag

```
<repositories>
<repository>
```

```
        <id>nexus</id>
        <url>repo-url</url>
</repository>
</repositories>
```

-> After adding the remote repository details do maven package and see dependency is downloading from nexus repo or not.

-> We will create users and will give access for users for our repositories