**EXPERIMENT-6**

**DEVOPS**

**Aim:** To understand Jenkins Master-Slave Architecture and Scale your Jenkins standalone implementation by implementing slave nodes

**LO:4** – Examine the importance of Selenium and Jenkins to test software Applications.

**THEORY:**

Jenkins is one of the most important tools in [DevOps](https://www.edureka.co/devops-certification-training). Jenkins is used in the Continuous Integration stage of DevOps. In this blog, I am going to talk about the Jenkins Master and Slave architecture. The pointers that I will cover are as follows:

* [What is Jenkins?](https://www.edureka.co/blog/jenkins-master-and-slave-architecture-a-complete-guide/#jenkins)
* [Jenkins Architecture](https://www.edureka.co/blog/jenkins-master-and-slave-architecture-a-complete-guide/#arch)
* [How does Jenkins Master Slave architecture work?](https://www.edureka.co/blog/jenkins-master-and-slave-architecture-a-complete-guide/#working)
* [Setting up Slaves with Jenkins Master](https://www.edureka.co/blog/jenkins-master-and-slave-architecture-a-complete-guide/#setup)

**What is Jenkins?**

* Jenkins is an open-source automation tool written in Java with plugins built for Continuous Integration purposes. Jenkins is used to build and test your software projects continuously making it easier for developers to integrate changes to the project, and making it easier for users to obtain a fresh build. It also allows you to continuously deliver your software by integrating with a large number of testing and deployment technologies.
* With Jenkins, organizations can accelerate the software development process through automation. Jenkins integrates development life-cycle processes of all kinds, including build, document, test, package, stage, deploy, static analysis and much more.

**Advantages of Jenkins include:**

* It is an open-source tool with great community support.
* Too easy to install.
* It has 1000+ plugins to ease your work. If a plugin does not exist, you can code it and share it with the community.
* It is free of cost.
* It is built with Java and hence, it is portable to all the major platforms.

**Jenkins Master:**

Your main Jenkins server is the Master. The Master’s job is to handle:

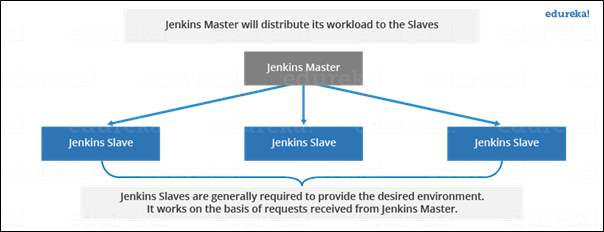
* Scheduling build jobs.
* Dispatching builds to the slaves for the actual execution.
* Monitor the slaves (possibly taking them online and offline as required).
* Recording and presenting the build results.
* A Master instance of Jenkins can also execute build jobs directly.

**Jenkins Slave:**

A Slave is a Java executable that runs on a remote machine. Following are the characteristics of Jenkins Slaves:

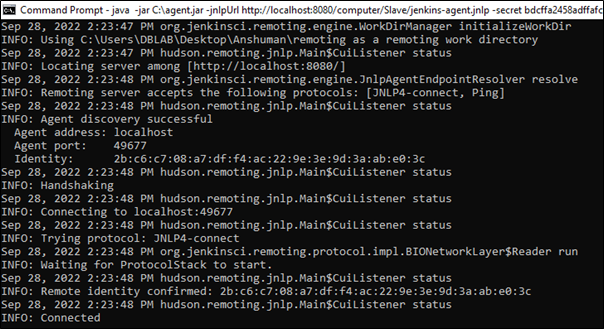
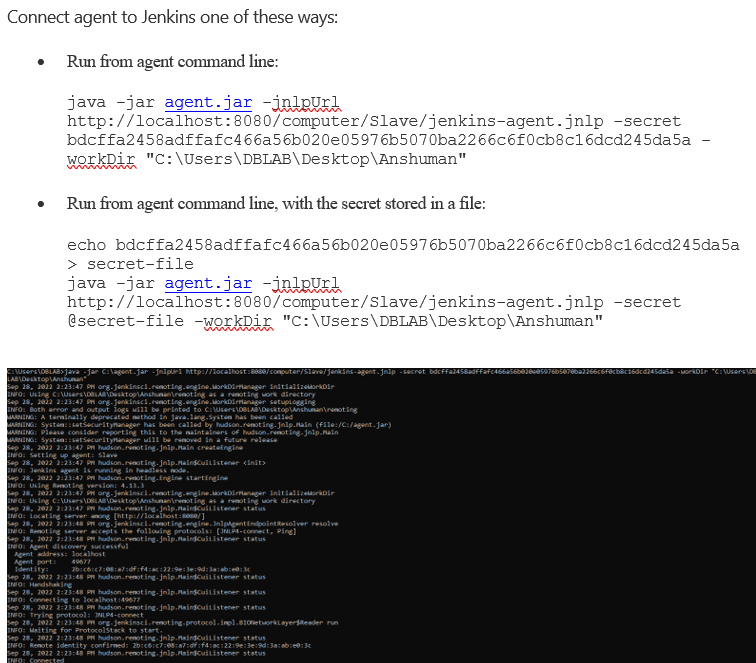
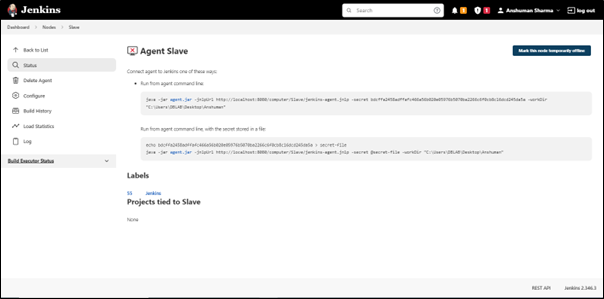
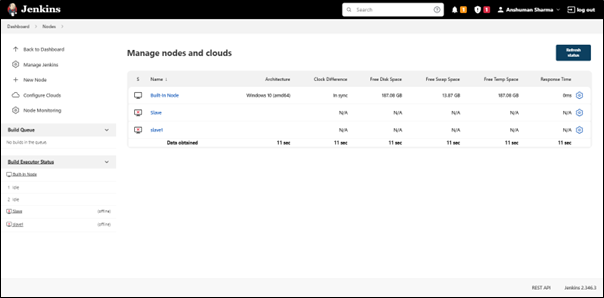
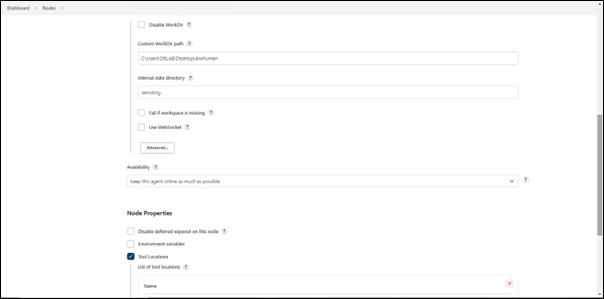
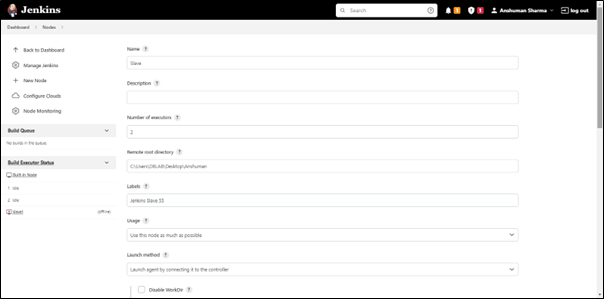
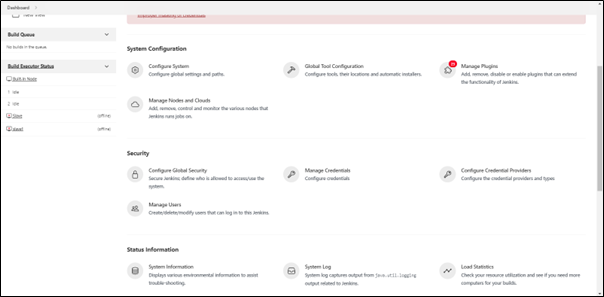
* It hears requests from the Jenkins Master instance.
* Slaves can run on a variety of operating systems.
* The job of a Slave is to do as they are told to, which involves executing build jobs dispatched by the Master.
* You can configure a project to always run on a particular Slave machine or a particular type of Slave machine, or simply let Jenkins pick the next available Slave.

The diagram below is self-explanatory. It consists of a Jenkins Master which is managing three Jenkins Slave.

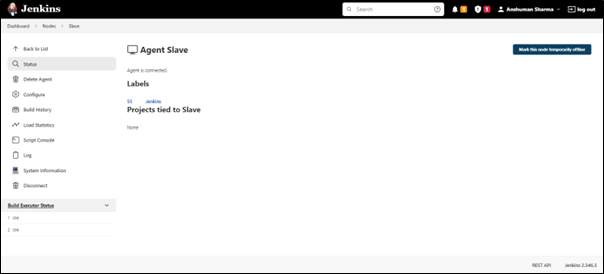


**OUTPUT:**

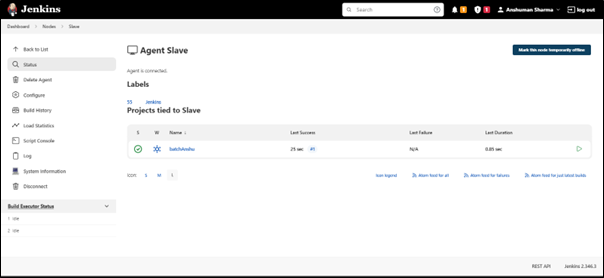
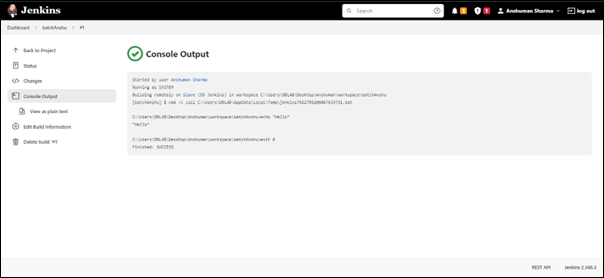
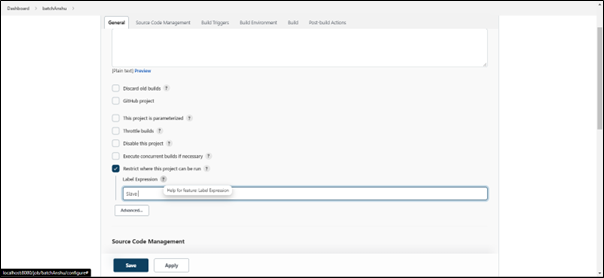
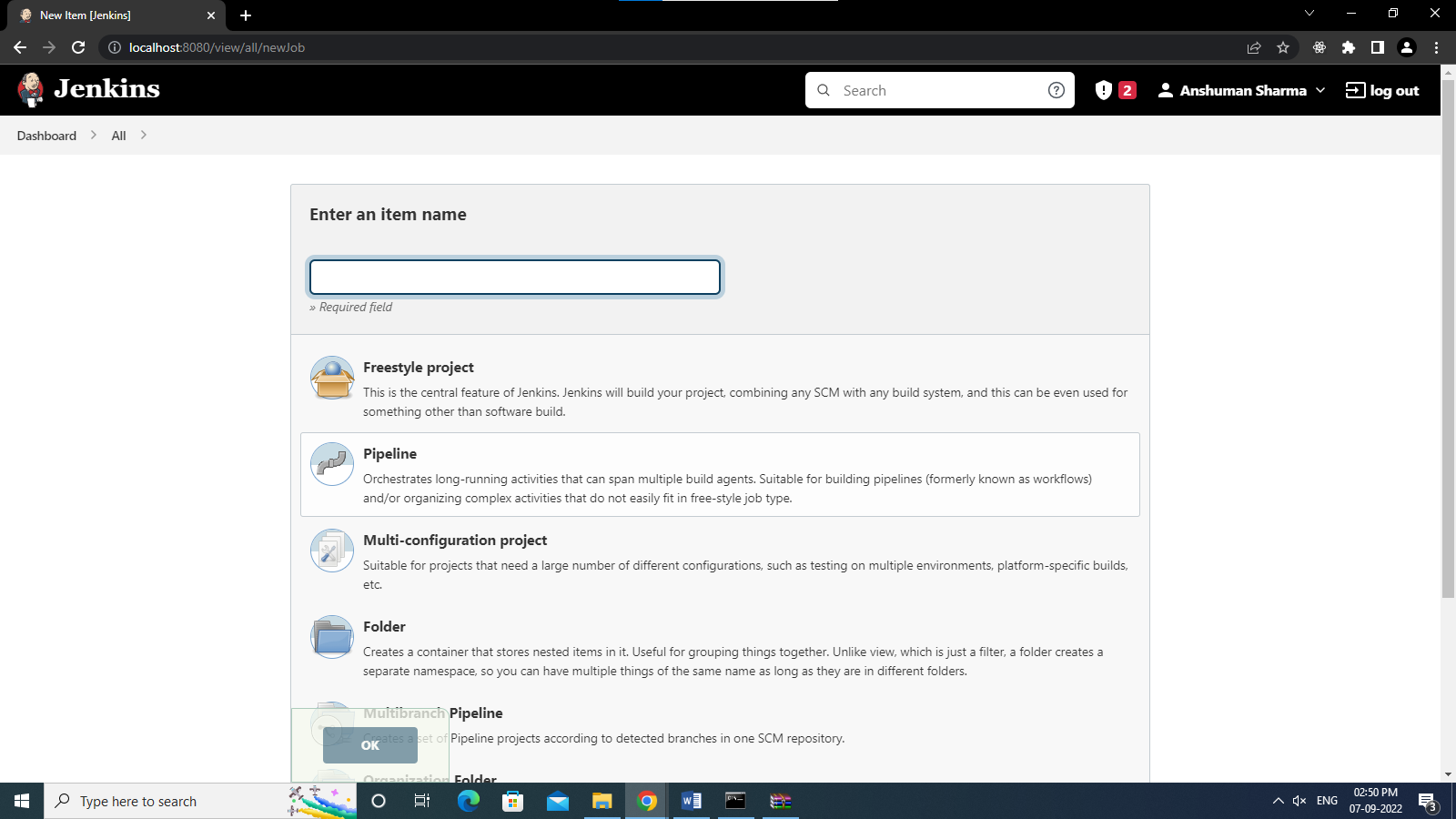
Step 1: Create node by clicking onto manage nodes and clouds



**Connected to the slave.**



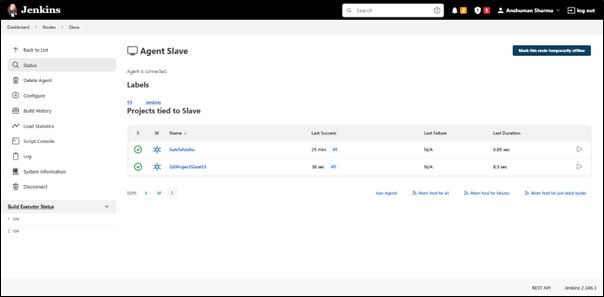
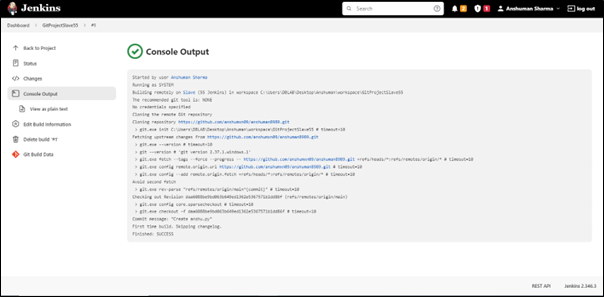
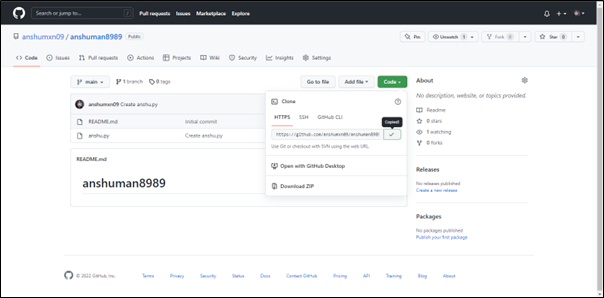
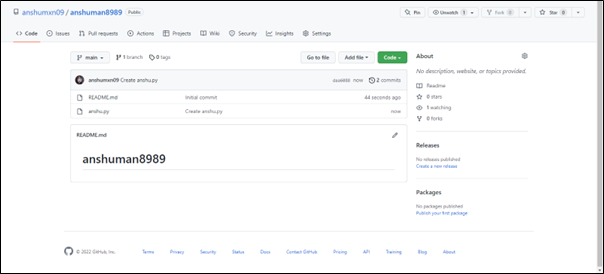
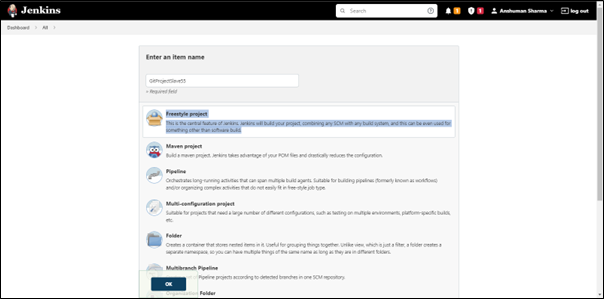
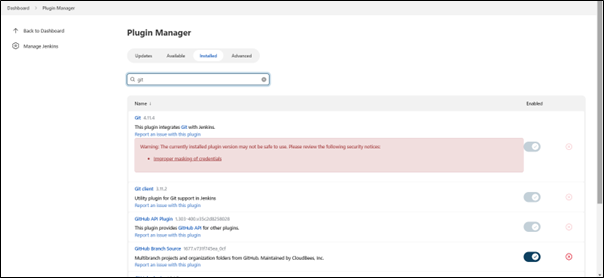
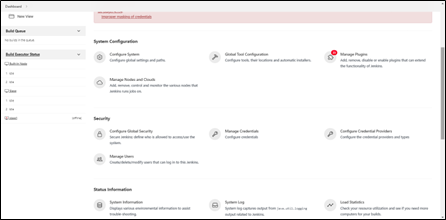
**Now you can assign the task to the slave**



**Assigning the Slave to the git Project:**

<https://www.guru99.com/jenkins-github-integration.html>

**Step1:**Open your dashboard.  
Click on the **Manage Jenkins**button on your Jenkins dashboard:



**CONCLUSION:**

PO1-PO5, PO12.