NAME :- Aman Singh
BATCH :- T11
ROLL NO :- 128

Assignment No 4

AIM: To understand Continuous integration, Jenkins installation.

LO1: To understand the fundamentals of DevOps engineering and be fully proficient with DevOps terminologies, concepts, benefits, and deployment options to meet your business requirements.

LO3: To understand the importance of Jenkins to Build and deploy Software Applications on server environment.

THEORY:

Jenkins is open source automation server. With Jenkins, organizations can accelerate the software development process by automating it. Jenkins manages and controls software delivery processes throughout the entire lifecycle, including build, document, test, package, stage, deployment, static code analysis and much more.

You can set up Jenkins to watch for any code changes in places like GitHub, Bitbucket or GitLab and automatically do a build a with tools like Maven and Gradle. You can utilize container technologies such as Docker and Kubernetes, initiate tests and then take actions like rolling back or rolling forward in production.

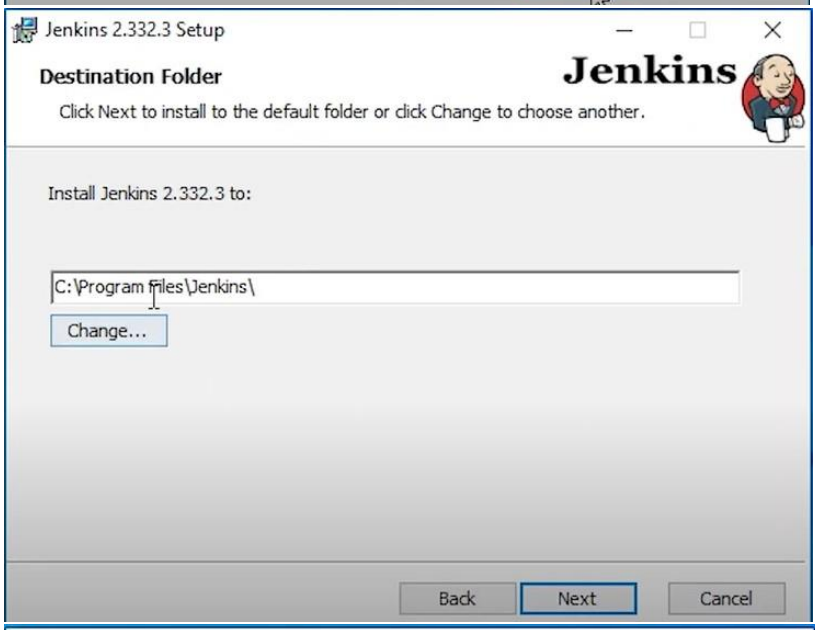
Originally developed by Kohsuke for continuous integration (CI), today Jenkins orchestrates the entire software delivery pipeline – called continuous delivery. For some organizations automation extends even further, to continuous deployment. Continuous delivery (CD), coupled with a DevOps culture, dramatically accelerates the delivery of software.

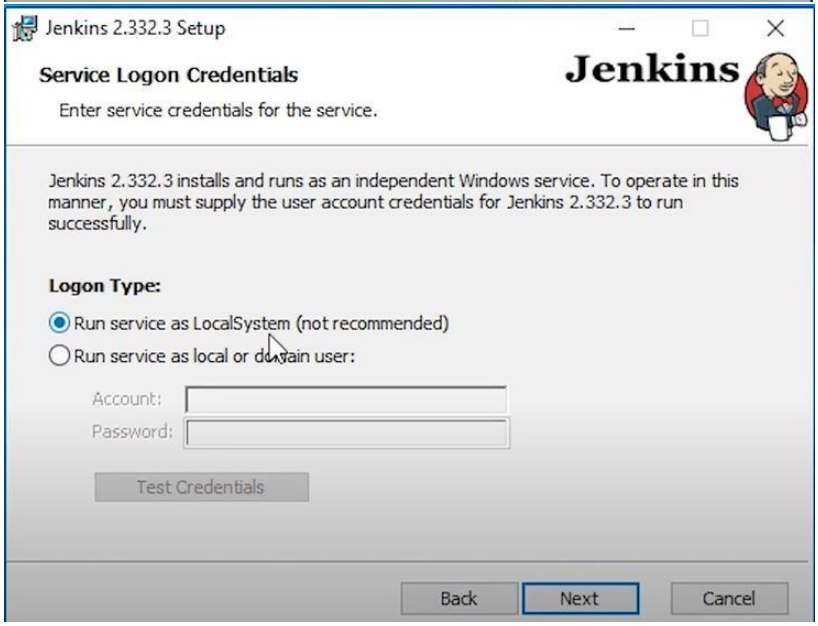
Jenkins is the most widely adopted solution for continuous delivery, thanks to its extensibility and a vibrant, active community. The Jenkins community offers more than 1,700 plugins that enable Jenkins to integrate with virtually any tool, including all of the best-of-breed solutions used throughout the continuous delivery process. Jenkins continues to grow as the dominant solution for software process automation, continuous integration and continuous delivery and, as of February 2018, there are more than 165,000 active installations and an estimated 1.65 million users around the world.

Jenkins offers a simple way to set up a continuous integration or continuous delivery (CI/CD) environment for almost any combination of languages and source code repositories using pipelines, as well as automating other routine development tasks. While Jenkins doesn't eliminate the need to create scripts for individual steps, it does give you a faster and more robust way to integrate your entire chain of build, test, and deployment tools than you can easily build yourself.

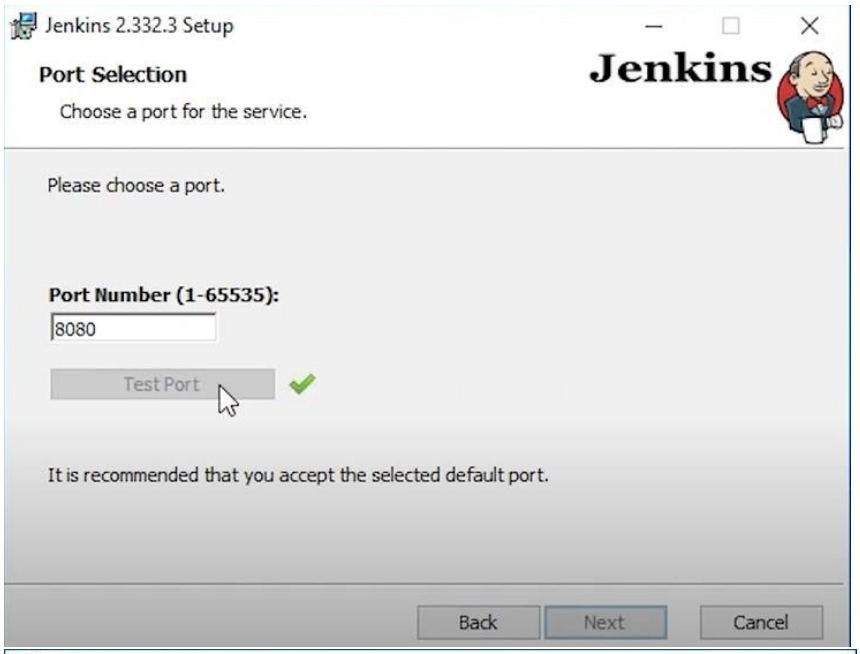
STEPS FOR INSTALLATION:

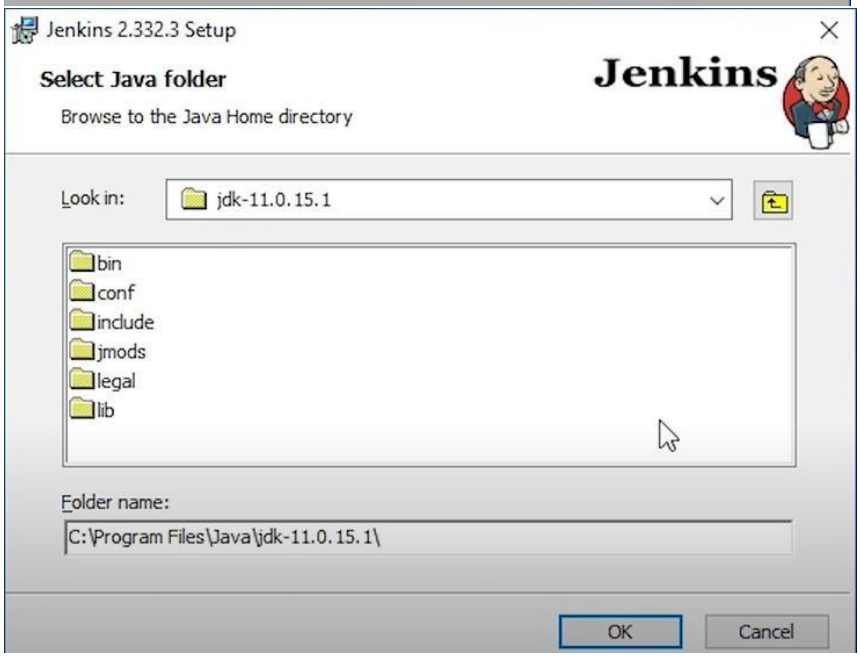
NAME :- Aman Singh
BATCH :- T11
ROLL NO :- 128



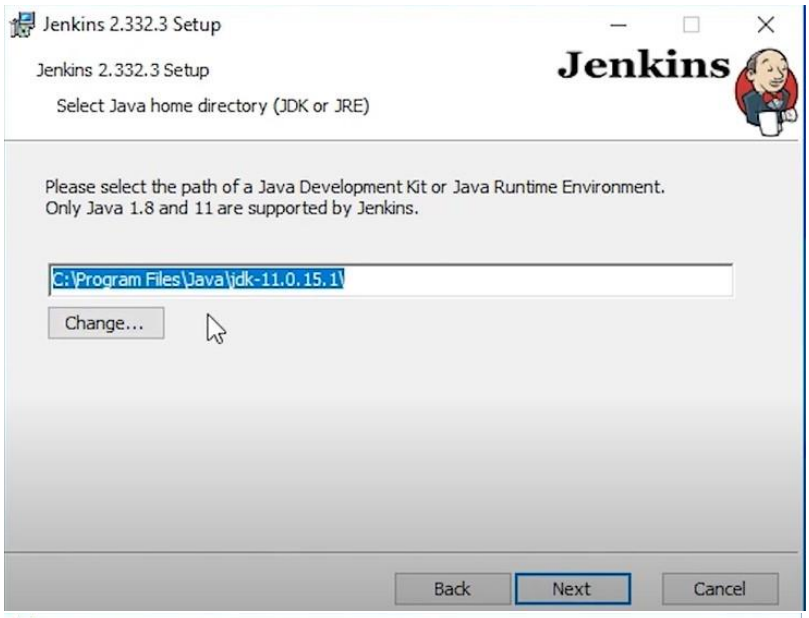


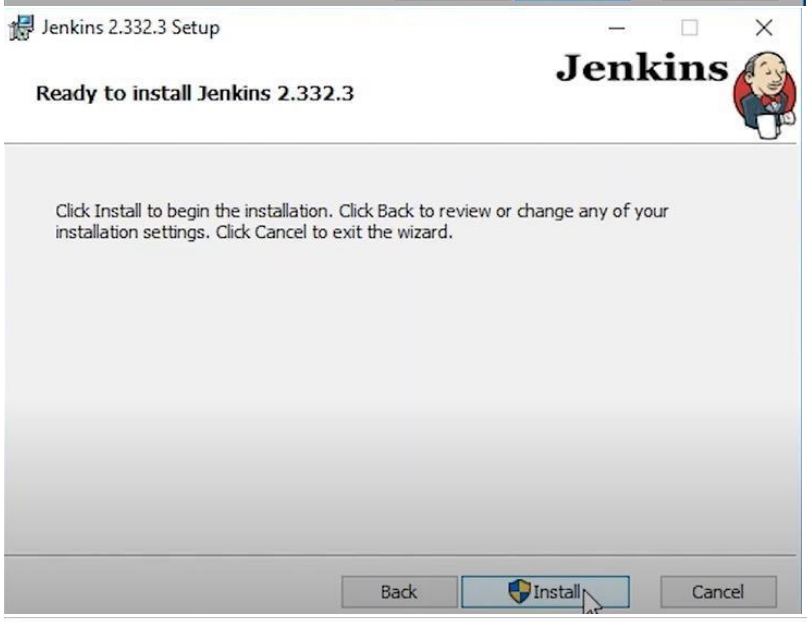
NAME :- Aman Singh
BATCH :- T11
ROLL NO :- 128

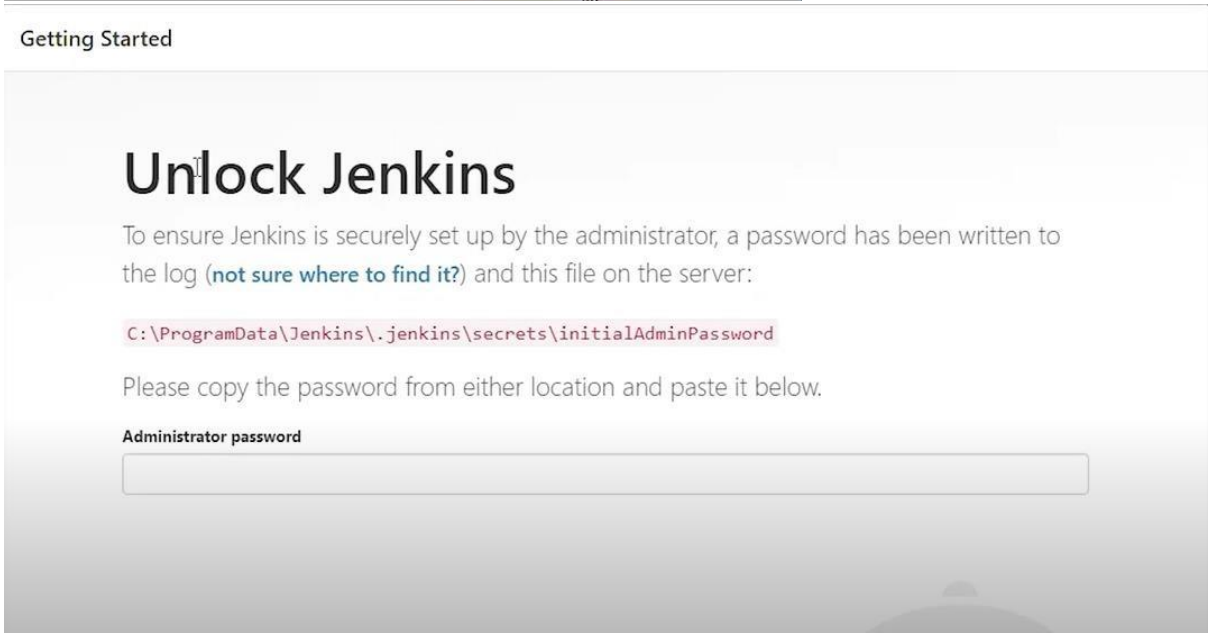




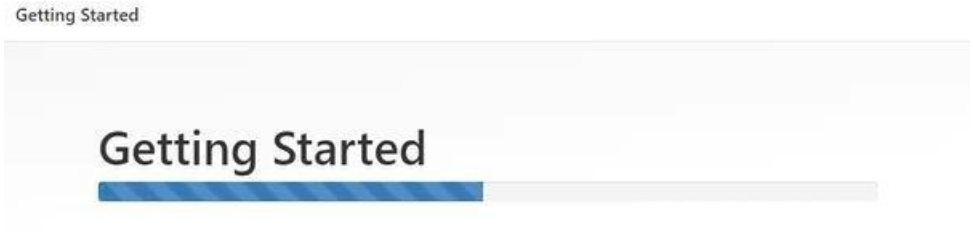
NAME :- Aman Singh
BATCH :- T11
ROLL NO :- 128

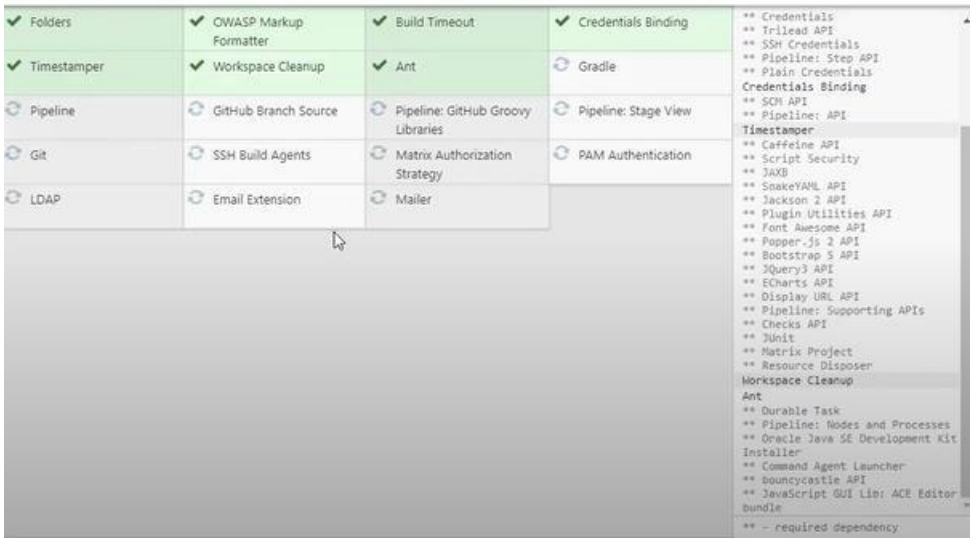




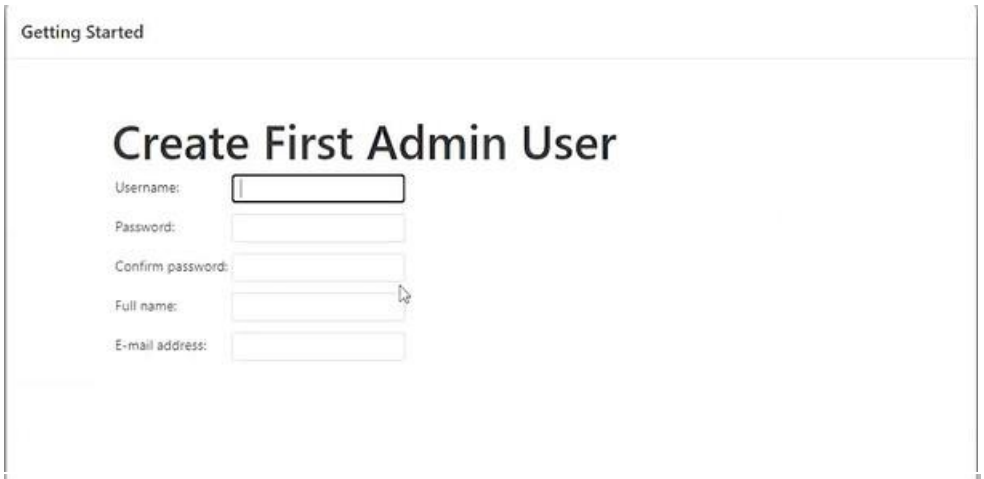


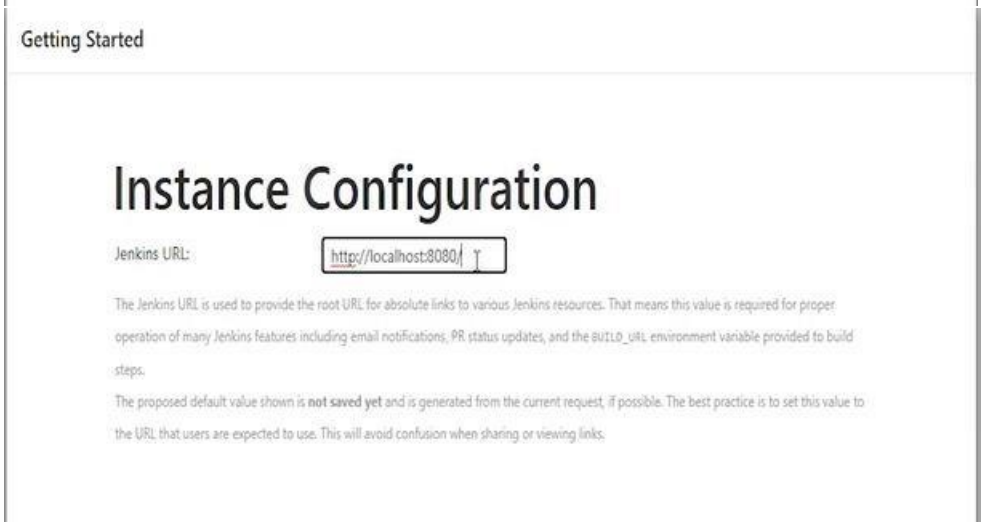
NAME :- Aman Singh
BATCH :- T11
ROLL NO :- 128



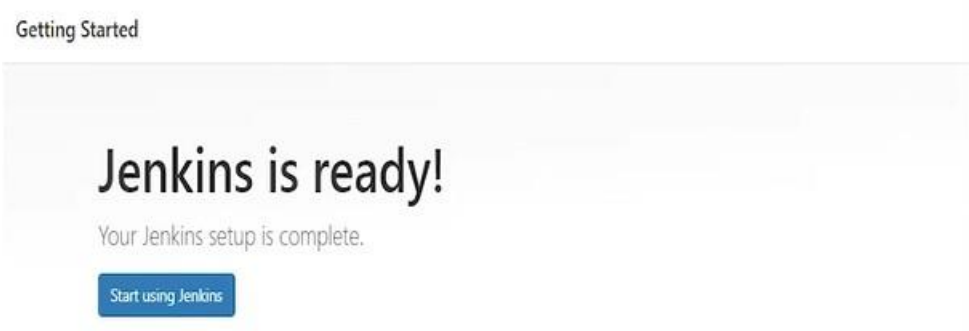


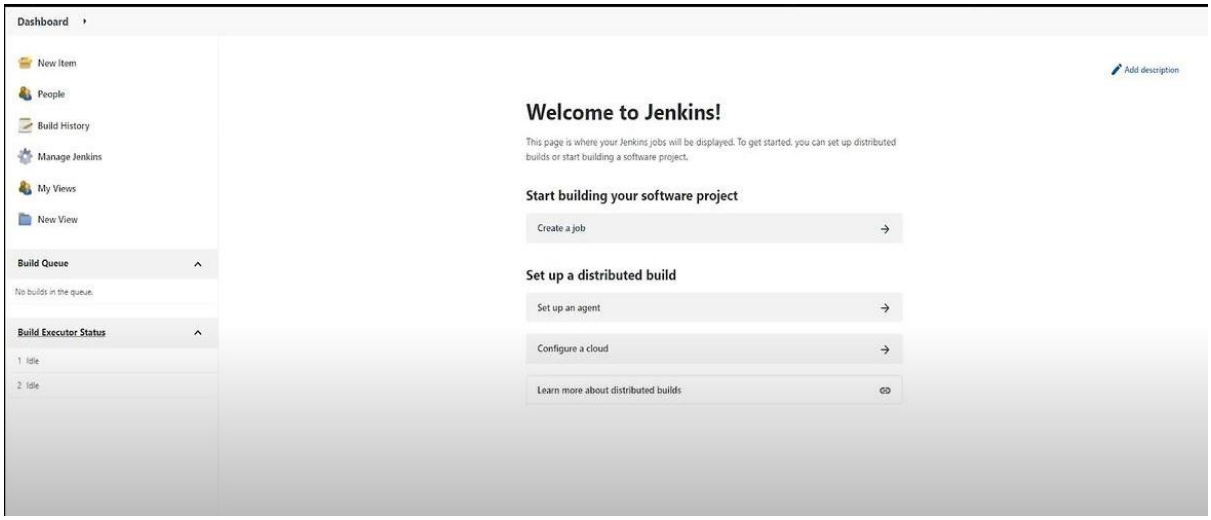
NAME :- Aman Singh
BATCH :- T11
ROLL NO :- 128





NAME :- Aman Singh
BATCH :- T11
ROLL NO :- 128





CONCLUSION:

Here, we understood the use of Jenkins and successfully installed it.