

PRAJWAL WAGHMODE
121A3060

Experiment No: 1

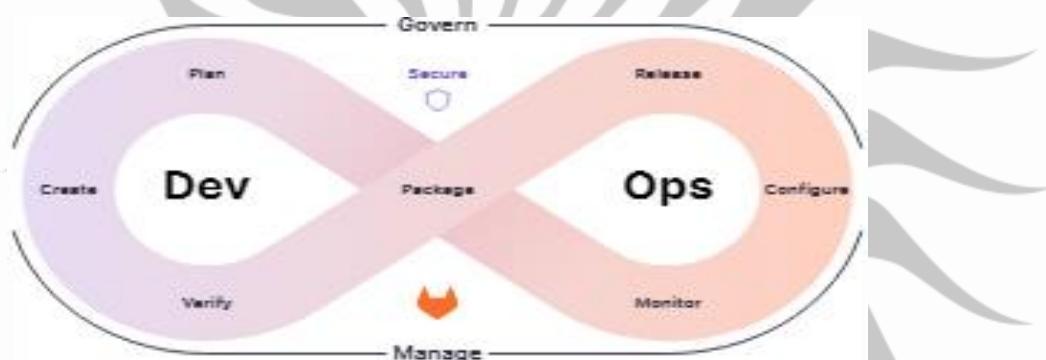
AIM: To understand DevOps: Principles, practices, DevOps, Engineer role and responsibilities.

THEORY: a) Introduction to DevOps

DevOps can be best explained as people working together to conceive, build and deliver secure software at top speed. DevOps practices enable software development (dev) and operations (ops) teams to accelerate delivery through automation, collaboration, fast feedback, and iterative improvement.

- What is DevOps ?

DevOps combines development and operations to increase the efficiency, speed, and security of software development and delivery compared to traditional processes. A more nimble software development lifecycle results in a competitive advantage for businesses and their customers.



- Need of DevOps

- 1) **Shorter development cycles that encourage innovation:** It is generally known that the more innovative companies are, the higher their chances of outrunning the competition. Which is essential to increase significantly competitiveness.
- 2) **More collaboration, better communication:** The DevOps culture is based on achieving the best performance in such a union, instead of worrying about individual objectives. As a result of both departments being fused, the process becomes more fluid since everyone is oriented towards a common goal.
- 3) **Reduced deployment failures and faster time to recover:** Most failures during development occur due to programming defects. Having a DevOps team will allow for more releases in shorter time spans. This way, it is easier and more likely to find possible defects in the code.
- 4) **Efficiency: Improved resource management:** Increased efficiency helps speed up development and reduce coding defects and problems.

- DevOps Principles

- 1) **Automation of the software development lifecycle:** This includes automating testing, builds, releases, the provisioning of development environments, and other manual tasks that can slow down or introduce human error into the software delivery process.
- 2) **Collaboration and communication:** A great DevOps team also has effective collaboration and communication.
- 3) **Continuous improvement and minimization of waste:** high performing DevOps teams are regularly looking for areas that could be improved.
- 4) **Hyperfocus on user needs with short feedback loops:** DevOps teams can take a moment and focus on what real users really want, and how to give it to them.

- DevOps Practices

- 1) **Continuous development.** This practice spans the planning and coding phases of the DevOps lifecycle. Version-control mechanisms might be involved.
Continuous testing. This practice incorporates automated, prescheduled, continued code tests as application code is being written or updated. Such tests can speed the delivery of code to production.
- 2) **Continuous integration (CI).** This practice brings configuration management (CM) tools together with other test and development tools to track how much of the code being developed is ready for production. It involves rapid feedback between testing and development to quickly identify and resolve code issues.
- 3) **Continuous delivery.** This practice automates the delivery of code changes, after testing, to a preproduction or staging environment. A staff member might then decide to promote such code changes into production.
- 4) **Continuous deployment (CD).** Similar to continuous delivery, this practice automates the release of new or changed code into production. A company doing continuous deployment might release code or feature changes several times per day. The use of container technologies, such as Docker and Kubernetes, can enable continuous deployment by helping to maintain consistency of the code across different deployment platforms and environments.
- 5) **Continuous monitoring.** This practice involves ongoing monitoring of both the code in operation and the underlying infrastructure that supports it. A feedback loop that reports on bugs or issues then makes its way back to development.
- 6) **Infrastructure as code.** This practice can be used during various DevOps phases to automate the provisioning of infrastructure required for a software release. Developers add infrastructure “code” from within their existing development tools.

- DevOps Engineer role and responsibilities
 - 1) **DevOps Evangelist** – The principal officer (leader) responsible for implementing DevOps
 - 2) **Release Manager** – The one releasing new features & ensuring post-release product stability
 - 3) **Automation Expert** – The guy responsible for achieving automation & orchestration of tools
 - 4) **Software Developer/ Tester** – The one who develops the code and tests it
 - 5) **Quality Assurance** – The one who ensures the quality of the product confirms to its requirement
 - 6) **Security Engineer** – The one always monitoring the product's security & health

b) Case study on DevOps use cases & Real-life applications

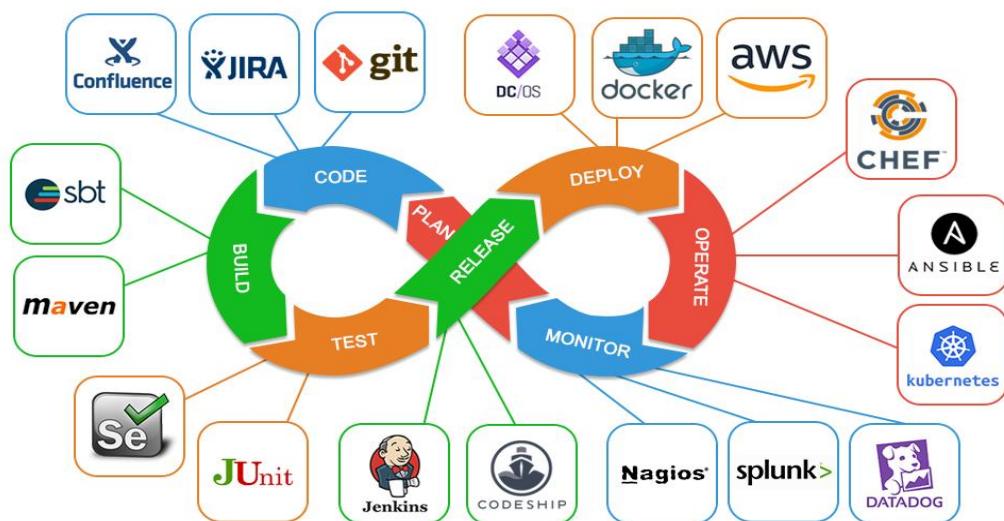
- Name of the Organization implementing DevOps: Amazon
- Challenges faced in implementing DevOps:

After Amazon started using DevOps in the form of Amazon Web Services (AWS), they faced several challenges and problems. While AWS has been highly successful, it doesn't mean that it was without any difficulties. Here are some challenges Amazon faced:

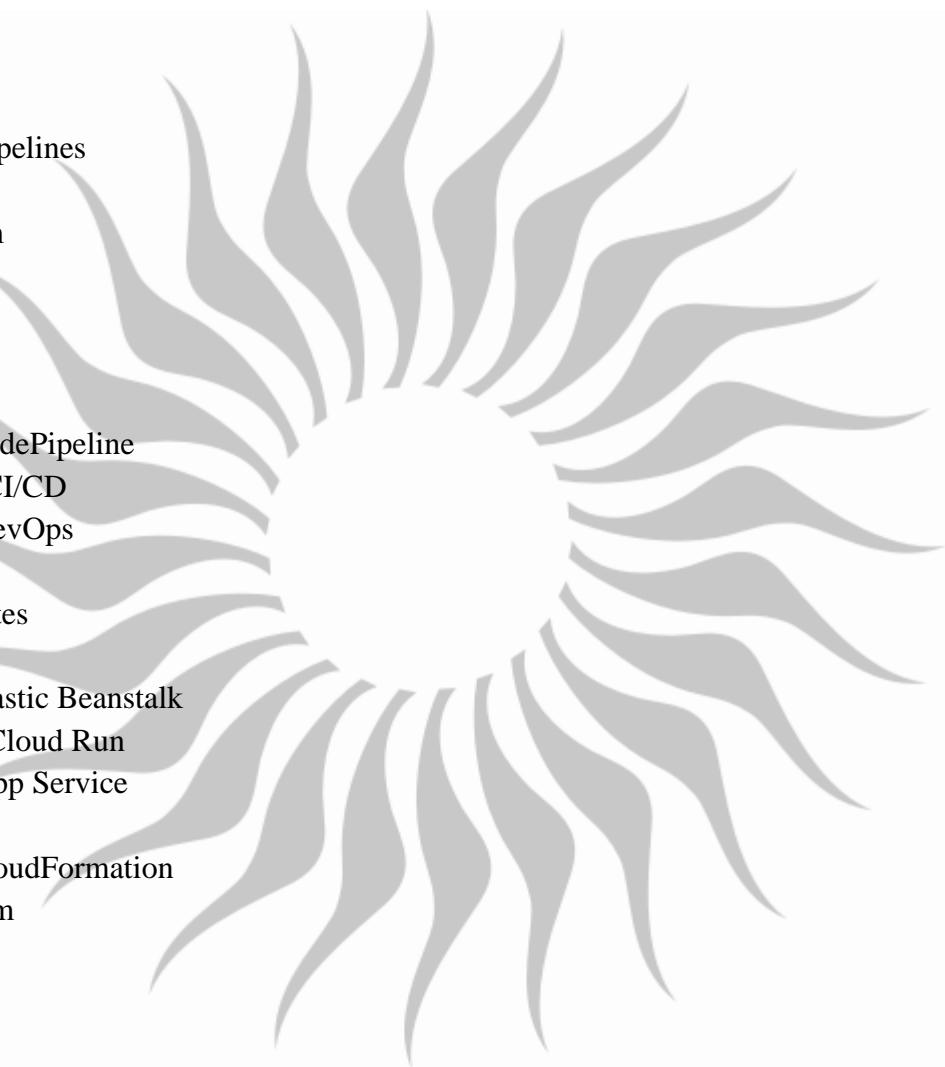
 - a. Scalability: Amazon tackled the challenge of accommodating increasing demand by continuously monitoring, optimizing, and adjusting their systems to ensure scalability.
 - b. Security: Amazon implemented robust security measures to address concerns about data security and privacy, safeguarding customer data and preventing unauthorized access or breaches.
 - c. Complexity: Amazon invested in tools, automation, and monitoring systems to simplify and streamline the management of the expanding AWS platform, which became increasingly complex with a wide range of services and features.
 - d. Service Outages: Amazon worked on improving fault tolerance, redundancy, and disaster recovery capabilities to minimize downtime and mitigate the impact of occasional service outages.
- Solution to overcome the challenges:
 - a. Scalability: Amazon invests in a highly scalable infrastructure and utilizes auto-scaling capabilities in AWS to handle increasing demand.
 - b. Security: Amazon prioritizes security in AWS, implementing features such as identity and access management, encryption, and compliance certifications. They provide comprehensive security documentation and guidelines for customers.
 - c. Complexity: Amazon develops tools like AWS CloudFormation, AWS Elastic Beanstalk, and AWS Systems Manager to simplify the management of AWS resources and streamline processes.
 - d. Service Outages: Amazon focuses on improving the resilience and availability of AWS services by investing in redundancy, fault tolerance, and disaster recovery mechanisms. They learn from past outages to prevent similar issues.

- Benefits of implementing the DevOps:
 1. Faster time to market: DevOps enables faster development cycles and quicker release of software, resulting in reduced time from concept to deployment.
 2. Continuous integration and deployment: DevOps emphasizes automated integration and deployment processes, ensuring frequent and reliable software updates.
 3. Improved collaboration and communication: DevOps fosters better collaboration and communication between teams, promoting shared responsibilities and effective problem-solving.
 4. Increased stability and reliability: DevOps practices, such as automated testing and continuous monitoring, enhance the stability and reliability of software systems, reducing failures and downtime.
 5. Scalability and flexibility: DevOps allows organizations to scale resources dynamically and adapt to changing demands, ensuring optimal performance and resource utilization.
 6. Enhanced quality and feedback loops: DevOps focuses on continuous improvement through practices like automated testing and feedback loops, resulting in higher software quality and customer satisfaction.
 7. Cost optimization: DevOps optimizes costs by reducing manual processes, minimizing downtime, and leveraging cloud resources efficiently, leading to cost savings.
 8. Encourages innovation and experimentation: DevOps provides a platform for rapid development and deployment, enabling teams to innovate and experiment with new ideas more easily.

c) List of various tools used in DevOps Life Cycle.



1. Plan:
 - a) Jira
 - b) Trello
 - c) Asana
2. Code:
 - a) Git
 - b) Bitbucket
 - c) GitHub
3. Build:
 - a) Jenkins
 - b) CircleCI
 - c) Azure Pipelines
4. Test:
 - a) Selenium
 - b) JUnit
 - c) Cypress
 - d) JMeter
5. Release:
 - a) AWS CodePipeline
 - b) GitLab CI/CD
 - c) Azure DevOps
6. Deploy:
 - a) Kubernetes
 - b) Docker
 - c) AWS Elastic Beanstalk
 - d) Google Cloud Run
 - e) Azure App Service
7. Operate:
 - a) AWS CloudFormation
 - b) Terraform
 - c) Puppet
 - d) Chef
8. Monitor:
 - a) Prometheus
 - b) Grafana
 - c) ELK Stack (Elasticsearch, Logstash, Kibana)
 - d) Datadog



Conclusion: Thus we have completed the experiment to understand DevOps: Principles, practices, DevOps, Engineer role and responsibilities and DevOps combines development and operations to enable faster, more efficient, and higher quality software delivery.

**PRAJWAL WAGHMODE
121A3060**

EXPERIMENT-2

Aim:

- A)** To understand Version Control System / Source Code Management, install git and create a GitHub account.
- B)** To perform various GIT operations on local and Remote repositories using GIT Cheat-Sheet.

Theory:

Version Control System/Source code management:

A version control system is a software that tracks changes to a file or set of files over time so that you can recall specific versions later. It also allows you to work together with other programmers. The version control system is a collection of software tools that help a team to manage changes in a source code. It uses a special kind of database to keep track of every modification to the code. Developers can compare earlier versions of the code with an older version to fix the mistakes.

GitHub is a global company that provides hosting for software development version control using Git. It is a Subsidiary of Microsoft, which acquired the company in 2018 for \$7.5 billion. It offers all of the distributed version control and source code management (SCM) functionality of Git as well as adding its own features. It provides access control and several collaboration features such as bug tracking, feature requests, task management, and wikis for every project.

GitHub offers plans for free, professional, and enterprise accounts. Free GitHub accounts are commonly

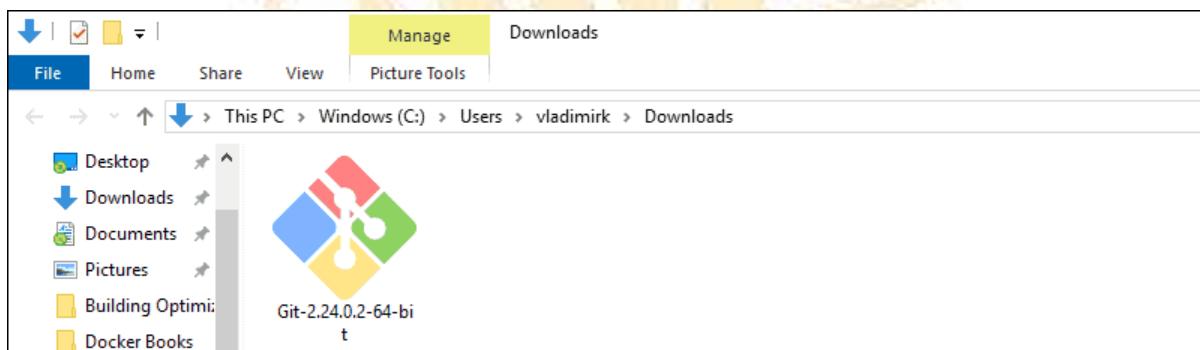
Used to host open source projects. As of January 2019, GitHub offers unlimited private repositories to allPlans, including free accounts. As of May 2019, GitHub reports having over 37 million users and more than 100 million repositories (including at least 28 million public repositories), making it the largest host of source code in the world.

Installation of Git:

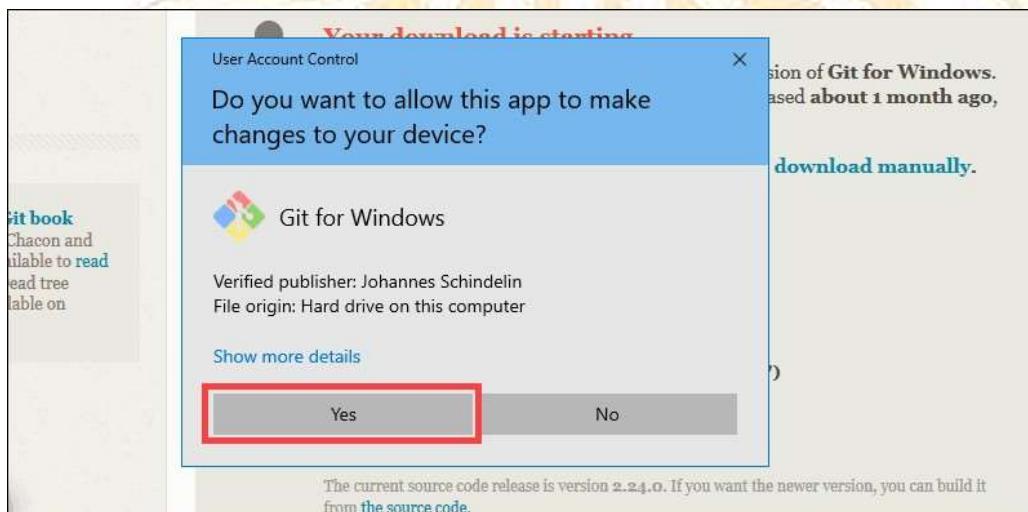
1. Browse to the official Git website: <https://git-scm.com/downloads>
2. Click the download link for Windows and allow the download to complete.



3. Browse to the download location (or use the download shortcut in your browser). Double-click the file to extract and launch the installer.



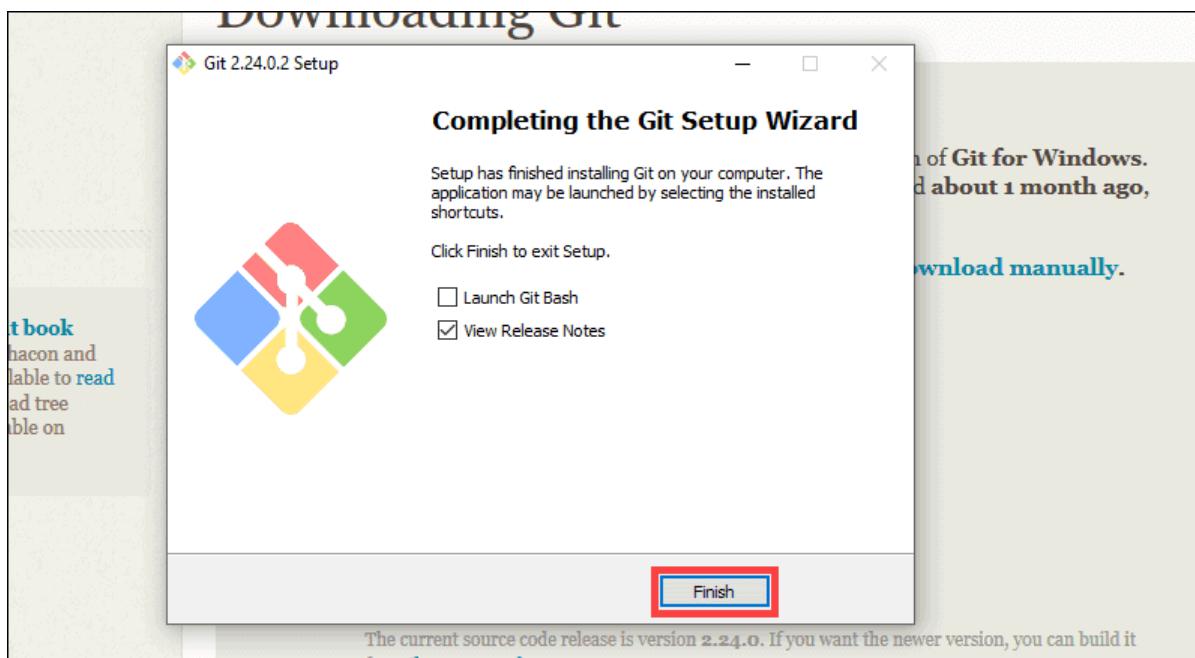
4. Allow the app to make changes to your device by clicking Yes on the User Account Control dialog that opens.



5. Review the [GNU General Public License](#), and when you're ready to install, click **Next**.

6. Continue to install the Git application as per your requirement.

7. Once the installation is complete, tick the boxes to view the Release Notes or Launch Git Bash, then click **Finish**.



Working on Repository:

Setting up local repository:

```
Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir
$ git --version
git version 2.40.0.windows.1

Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir
$ git init
Initialized empty Git repository in E:/myWorkDir/.git/

Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (master)
$ git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)
```

Adding files to git and committing:

```
Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (master)
$ touch f1 f2 f3

Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (master)
$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    f1
    f2
    f3

nothing added to commit but untracked files present (use "git add" to track)
```

Setting user configuration

```
Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (master)
$ git config --global user.name "SP"

Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (master)
$ git config --global user.email "prajwalw02@gmail.com"

Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (master)
$ git config --global --list
user.name=SP
user.email=prajwalw02@gmail.com
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
filter.lfs.clean=git-lfs clean -- %f
```

```
Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (master)
$ git config --global --list
user.name=Sp
user.email=prajwalw02@gmail.com
```

Checking the status of untracked files

```
Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (master)
$ touch f1 f2 f3

Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (master)
$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    f1
    f2
    f3

nothing added to commit but untracked files present (use "git add" to track)
```

Adding files to staging area

```
Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (master)
$ cat >>f1
this is file 1 line 1
Prajwal Shreesh

Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (master)
$ cat f1
this is file 1 line 1
Prajwal Shreesh

Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (master)
$ git add f1
warning: in the working copy of 'f1', LF will be replaced by CRLF the next time Git touches

Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (master)
$ git status
On branch master

No commits yet

Changes to be committed:
(use "git rm --cached <file>..." to unstage)
  new file:   f1

Untracked files:
(use "git add <file>..." to include in what will be committed)
  f2
  f3

Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (master)
$ cat >>f2
121A3060 121A3060
pw sk

Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (master)
$ cat >>f3
hello this is file 3

Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (master)
$ git add .
warning: in the working copy of 'f2', LF will be replaced by CRLF the next time Git touches
warning: in the working copy of 'f3', LF will be replaced by CRLF the next time Git touches

Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (master)
$ git status
On branch master

No commits yet

Changes to be committed:
(use "git rm --cached <file>..." to unstage)
  new file:   f1
  new file:   f2
```

Committing changes and Detecting changed files:

```
Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (master)
$ git commit -m "first version" f1
warning: in the working copy of 'f1', LF will be replaced by CRLF the next time Git touches it
[master (root-commit) ac87627] first version
 1 file changed, 2 insertions(+)
 create mode 100644 f1

Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   f2
    new file:   f3

Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (master)
$ git log
commit ac876277a9a41f2fc6d65818d212a089a6224d4a (HEAD -> master)
Author: SP <prajwalw02@gmail.com>
Date:   Thu Aug 3 16:15:27 2023 +0530

  first version

Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (master)
$ git log --oneline
ac87627 (HEAD -> master) first version

Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (master)
$ git commit -m "file 2 and 3" f2 f3
warning: in the working copy of 'f2', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'f3', LF will be replaced by CRLF the next time Git touches it
[master 78e533b] file 2 and 3
 2 files changed, 3 insertions(+)
 create mode 100644 f2
 create mode 100644 f3

Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (master)
$ git log --oneline
78e533b (HEAD -> master) file 2 and 3
ac87627 first version

Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (master)
$ git log
commit 78e533be6fd7084b081e381d4d738cf4de4b4080 (HEAD -> master)
Author: SP <prajwalw02@gmail.com>
Date:   Thu Aug 3 16:18:14 2023 +0530

  file 2 and 3

commit ac876277a9a41f2fc6d65818d212a089a6224d4a
Author: SP <prajwalw02@gmail.com>
Date:   Thu Aug 3 16:15:27 2023 +0530

  first version
```

Adding remote repository:

```
Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (master)
$ git branch
* master

Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (master)
$ git branch -M main

Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (main)
$ git remote add origin https://github.com/prajwal0211/Experiment2DevOps.git

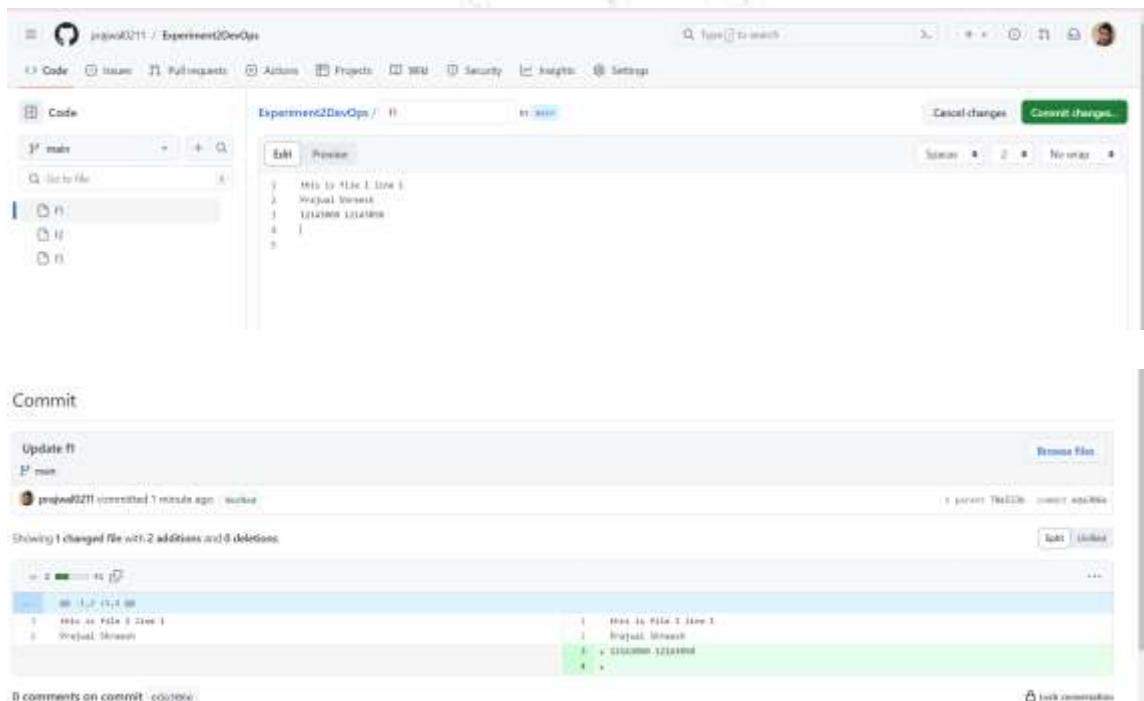
Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (main)
$ git remote show origin
* remote origin
  Fetch URL: https://github.com/prajwal0211/Experiment2DevOps.git
  Push URL: https://github.com/prajwal0211/Experiment2DevOps.git
  HEAD branch: (unknown)
```

Showing a Remote Repository & Pushing to a Remote Repository:

```
Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (main)
$ git push -u origin main
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 8 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (7/7), 540 bytes | 540.00 KiB/s, done.
Total 7 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/prajwal0211/Experiment2DevOps.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.

Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (main)
$ git remote show origin
* remote origin
  Fetch URL: https://github.com/prajwal0211/Experiment2DevOps.git
  Push URL: https://github.com/prajwal0211/Experiment2DevOps.git
  HEAD branch: main
  Remote branch:
    main tracked
  Local branch configured for 'git pull':
    main merges with remote main
  Local ref configured for 'git push':
    main pushes to main (up to date)
```

Updating First File on Remote Repository



Pulling changes from remote repository:

```
Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (main)
$ git pull https://github.com/prajwal0211/Experiment2DevOps.git
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 717 bytes | 6.00 KiB/s, done.
From https://github.com/prajwal0211/Experiment2DevOps
 * branch           HEAD      -> FETCH_HEAD
Updating 78e533b..eda306e
Fast-forward
 f1 | 2 ++
 1 file changed, 2 insertions(+)

Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (main)
$ cat f1
this is file 1 line 1
Prajwal Shreesh
121A3060 121A3050
```

Fetching Git

```
Avinash@DESKTOP-K2P82MI MINGW64 /E/myworkDir (main)
$ git fetch https://github.com/prajwal0211/Experiment2DevOps.git
From https://github.com/prajwal0211/Experiment2DevOps
 * branch           HEAD      -> FETCH_HEAD

Avinash@DESKTOP-K2P82MI MINGW64 /E/myworkDir (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean

Avinash@DESKTOP-K2P82MI MINGW64 /E/myworkDir (main)
$ git log
commit eda306e5e7bfbaab4da249c3652a3569fbcb8db9de (HEAD -> main)
Author: Prajwal Waghmode <121559901-prajwal0211@users.noreply.github.com>
Date:   Thu Aug 3 16:37:01 2023 +0530

    Update f1

commit 78e533be6fd7084b081e381dd738cf4de4b4080 (origin/main)
Author: SP <prajwalw02@gmail.com>
Date:   Thu Aug 3 16:18:14 2023 +0530

    file 2 and 3
```

```
Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (main)
$ cat >>f2
prajwal waghmode
shreesh kulkarni

Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (main)
$ cat f2
121A3060 121A3060
pw_sk
prajwal waghmode
shreesh kulkarni

Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (main)
$ git add .
warning: in the working copy of 'f2', LF will be replaced by CRLF the next time Git touches it

Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   f2

Avinash@DESKTOP-K2P82MI MINGW64 /E/myWorkDir (main)
$ git commit -m "file 2 commit" f2
warning: in the working copy of 'f2', LF will be replaced by CRLF the next time Git touches it
[main 2e02d0a] file 2 commit
 1 file changed, 2 insertions(+)
```

```

Avinash@DESKTOP-K2PBZMI MINGW64 /E/myWorkDir (main)
$ git push -u origin main
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 335 bytes | 335.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/prajwal0211/Experiment2DevOps.git
  eda306e..2e02d0a main -> main
branch 'main' set up to track 'origin/main'.

Avinash@DESKTOP-K2PBZMI MINGW64 /E/myWorkDir (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean

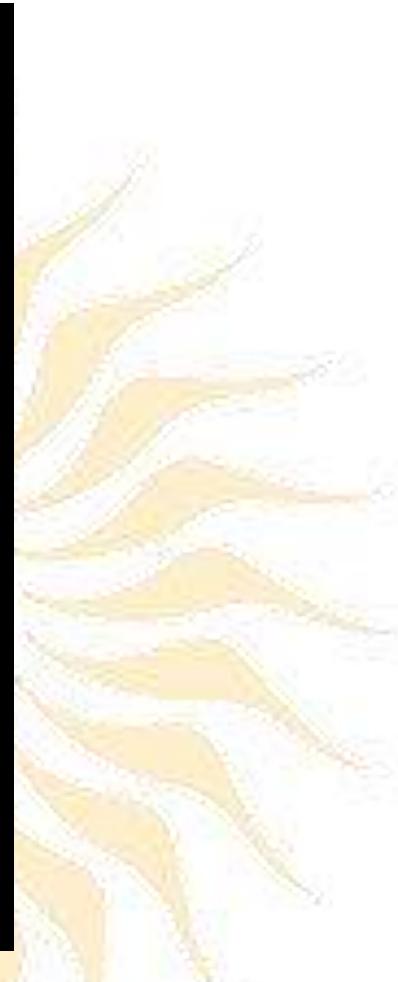
Avinash@DESKTOP-K2PBZMI MINGW64 /E/myWorkDir (main)
$ cat f2
121A3060 121A3060
pw sk
prajwal waghmode
shreesh kulkarni

Avinash@DESKTOP-K2PBZMI MINGW64 /E/myWorkDir (main)
$ git pull https://github.com/prajwal0211/Experiment2DevOps.git
remote: Enumerating objects: 8, done.
remote: Counting objects: 100% (8/8), done.
remote: Compressing objects: 100% (6/6), done.
remote: Total 6 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (6/6), 1.37 KiB | 2.00 KiB/s, done.
From https://github.com/prajwal0211/Experiment2DevOps
 * branch           HEAD      -> FETCH_HEAD
Updating 2e02d0a..e864abb
Fast-forward
 f2 | 2 ++
 1 file changed, 2 insertions(+)

Avinash@DESKTOP-K2PBZMI MINGW64 /E/myWorkDir (main)
$ git log
commit e864abbed0071615a4dfcbb0dce3649adc54233a3 (HEAD -> main)
Author: Prajwal Waghmode <121559901+prajwal0211@users.noreply.github.com>
Date:   Thu Aug 3 16:54:48 2023 +0530

    final f2

```



Experiment2DevOps [Public](#)

[Pin](#) [Unwatch](#) 1

[main](#) • [1 branch](#) [0 tags](#)

[Go to file](#) [Add file](#) [Code](#)

prajwal0211 Update f2 final	4817aF# now	5 commits
f1	Update f1	12 minutes ago
f2	Update f2 final	now
f3	file 2 and 3	31 minutes ago

[Commit](#)

Update f2 final
by [prajwal0211](#) committed 2 minutes ago | [View file](#)

Browsed files
1 parent: 2e02d0a | commit: 4817aF#

Showing 1 changed file with 1 addition and 0 deletions.

File	Line	Content
1	1	prajwal waghmode
2	2	shreesh kulkarni
3	3	+ Hello world

Synchronized repository

Commit

final f2
1 project segments
2 stress lookout
prajwal0211 committed now Verified Commit ID: 0f017af8 Date: 2023-08-03T10:44:00Z

Showing 1 changed file with 1 addition and 0 deletions.

Split Unified

0 comments on commit 0f017af8 Lock conversion

Reading Log of commits:

Commits

main

Commits on Aug 3, 2023

Commit	Author	Date	Message	Status	SHA
final f2	prajwal0211	1 minute ago	final f2	Verified	a165ab6
Update f2 final	prajwal0211	7 minutes ago	Update f2 final	Verified	4f017af8
file 2 commit	SP	8 minutes ago	file 2 commit		2a02d8a
Update f1	prajwal0211	10 minutes ago	Update f1	Verified	61a30fa
file 2 and 3	SP	11 minutes ago	file 2 and 3		70e533b
first version	SP	40 minutes ago	first version		ac87627

Conclusion :

Hence, we have successfully performed the experiment to Installation of Git and Working with local and Remote Repository and thus learned how to create remote origin , commit , push and pull file to in the repository created.

**PRAJWAL WAGHMODE
121A3060**

Experiment No.: 3

AIM:

Branching and Merging in Git.

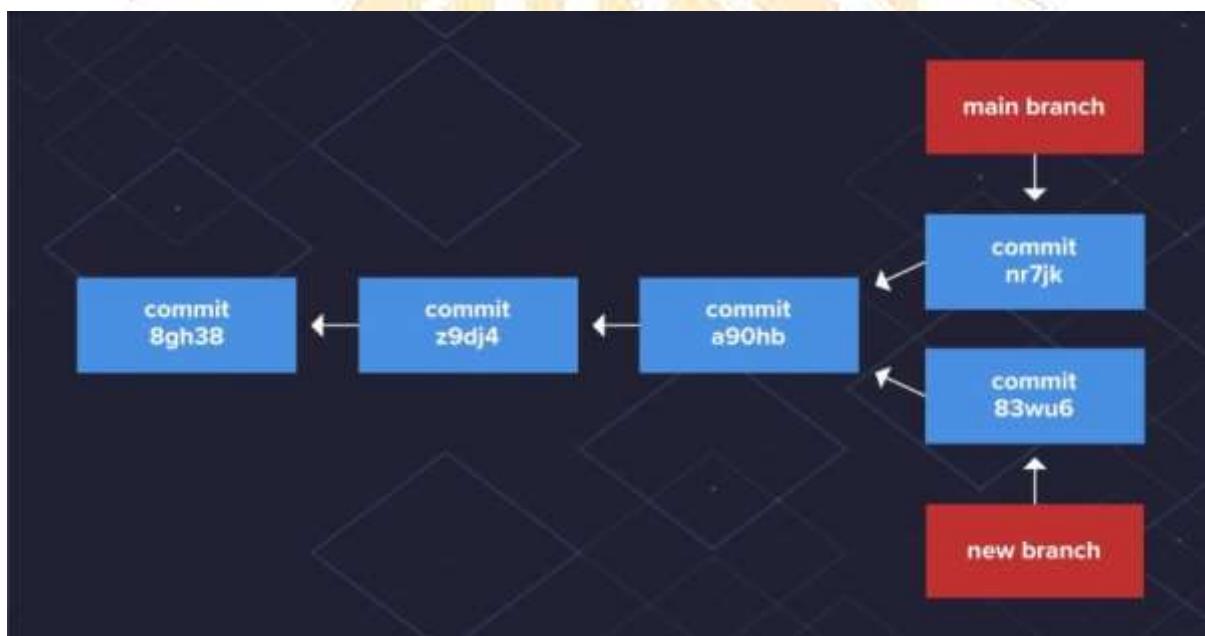
THEORY:

Git branching is a powerful feature of the version control system Git that allows developers to work on multiple independent lines of development within the same repository. Each line of development is represented by a branch, which is essentially a pointer to a specific commit in the commit history of the repository.

Git branching allows developers to diverge from the production version of code to fix a bug or add a feature. Developers create branches to work with a copy of the code without modifying the existing version. You create branches to isolate your code changes, which you test before merging to the main branch.

Once you have completed working on your branch, it is time to merge it into the main branch. Git merging is the process of combining the changes from one branch into another branch. When you have multiple branches in a Git repository, merging allows you to integrate the work done in one branch (source branch) into another branch (target branch). This helps to consolidate changes and ensures that all the new features, bug fixes, or improvements made in one branch become part of the main codebase.

Branching:



Merging:



RESULT:

- Branching:

```
exam@IT121 MINGW64 /c/Users/exam/Desktop/gitWorkDir (main)
$ cat f1
Prajwal AND shreesh
pw sk
SK PW
60 50

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitWorkDir (main)
$ git init
Reinitialized existing Git repository in C:/Users/exam/Desktop/gitWorkDir/.git/

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitWorkDir (main)
$ git branch feature1

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitWorkDir (main)
$ git branch
  feature1
* main

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitWorkDir (main)
$ ls
f1  f2

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitWorkDir (main)
$ git status
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   f2

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitWorkDir (main)
$ git commit -m "First version of f2" f2
warning: in the working copy of 'f2', LF will be replaced by CRLF the next time Git touches it
[main b10e3a5] First version of f2
  1 file changed, 2 insertions(+)
```

```
exam@IT121 MINGW64 /c/Users/exam/Desktop/gitWorkDir (main)
$ git checkout main
Already on 'main'
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitWorkDir (main)
$ git checkout feature1
Switched to branch 'feature1'

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitWorkDir (feature1)
$ ls
f1 f2

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitWorkDir (feature1)
$ cat f2
121A3060 And 121A3050
60 50
prajwal shreesh

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitWorkDir (feature1)
$ cat >> f2
feature 1

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitWorkDir (feature1)
$ git status
On branch feature1
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
    (use "git restore <file>..." to discard changes in working directory)
      modified:   f2

no changes added to commit (use "git add" and/or "git commit -a")

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitWorkDir (feature1)
$ git add f2
warning: in the working copy of 'f2', LF will be replaced by CRLF the next time Git touches it

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitWorkDir (feature1)
$ git commit -m "features 1 commit" f2
[feature1 6f5a63f] features 1 commit
 1 file changed, 1 insertion(+)

 0:00:00
```

```
MINGW64:/c/Users/exam/Desktop/gitWorkDir
exam@IT121 MINGW64 /c/Users/exam/Desktop/gitWorkDir (feature1)
$ git commit -m "features 1 commit" f2
warning: in the working copy of 'f2', LF will be replaced by CRLF the next time Git touches it
[feature1 6f5a63f] features 1 commit
 1 file changed, 1 insertion(+)

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitWorkDir (feature1)
$ cat f2
121A3060 And 121A3050
60 50
prajwal shreesh
feature 1

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitWorkDir (feature1)
$ git checkout main
Switched to branch 'main'
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)
```

- Merging:

```
exam@IT121 MINGW64 /c/Users/exam/Desktop/gitWorkDir (main)
$ git merge feature1
Updating 5c17e3a..6f5a63f
Fast-forward
 f2 | 1 +
 1 file changed, 1 insertion(+)

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitWorkDir (main)
$ cat f2
121A3060 And 121A3050
60 50
prajwal shreesh
feature 1

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitWorkDir (main)
$ git log --oneline --graph
fatal: unrecognized argument: --online

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitWorkDir (main)
$ git log --oneline --graph
* 6f5a63f (HEAD -> main, feature1) features 1 commit
* 5c17e3a Update f2
* b10e3a5 (origin/main) First version of f2
* 32c2eab Update f1
* e851be4 Second version
* 5c0f013 first version

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitWorkDir (main)
$ git checkout feature1
Switched to branch 'feature1'
```

```
exam@IT121 MINGW64 /c/Users/exam/Desktop/gitWorkDir (main)
$ cat f2
121A3060 And 121A3050
60 50
prajwal shreesh
feature 1

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitWorkDir (main)
$ git log --oneline --graph
fatal: unrecognized argument: --online

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitWorkDir (main)
$ git log --oneline --graph
* 6f5a63f (HEAD -> main, feature1) features 1 commit
* 5c17e3a Update f2
* b10e3a5 (origin/main) First version of f2
* 32c2eab Update f1
* e851be4 Second version
* 5c0f013 first version

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitWorkDir (main)
$ git checkout feature1
Switched to branch 'feature1'

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitWorkDir (feature1)
$ cat f1
Prajwal AND shreesh
pw sk
SK PW
60 50

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitWorkDir (feature1)
$ cat >>f1
```



```

MINGW64 /c/Users/exam/Desktop/gitworkDir (feature1)
$ cat >>f1
Line1
good morning

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitworkDir (feature1)
$ git status
On branch feature1
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
    (use "git restore <file>..." to discard changes in working directory)
      modified:   f1

no changes added to commit (use "git add" and/or "git commit -a")

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitworkDir (feature1)
$ git add f1
warning: in the working copy of 'f1', LF will be replaced by CRLF the next time Git touches it

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitworkDir (feature1)
$ git commit -m "Second commit of f1"
f1
warning: in the working copy of 'f1', LF will be replaced by CRLF the next time Git touches it
[feature1 de50b1f] Second commit of f1
 1 file changed, 2 insertions(+)

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitworkDir (feature1)
$ git status
On branch feature1
nothing to commit, working tree clean

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitworkDir (main)
$ git checkout main
Switched to branch 'main'
Your branch is ahead of 'origin/main' by 2 commits.
  (use "git push" to publish your local commits)

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitworkDir (main)
$ git merge --no-ff feature1
Merge made by the 'ort' strategy.
 f1 | 2 ++
 1 file changed, 2 insertions(+)

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitworkDir (main)
$ git log --oneline --graph
* 4e3f47e (HEAD -> main) Merge branch 'feature1'
|\
| * de50b1f (feature1) Second commit of f1
|/
* 6f5a63f features 1 commit
* 5c17e3a Update f2
* b10e3a5 (origin/main) First version of f2
* 32c2eb0 Update f1
* e891be4 Second version
* 5c0f013 first version

exam@IT121 MINGW64 /c/Users/exam/Desktop/gitworkDir (main)
$ ls
f1 f2

```

- Conflict while merge:

```

Avinash@DESKTOP-K2P82MI MINGW64 /E/myworkDir (feature1)
$ cat f1
this is file 1 line 1
Prajwal Shreesh
121A3060 121A3050

Avinash@DESKTOP-K2P82MI MINGW64 /E/myworkDir (feature1)
$ cat >>f1
feature1

Avinash@DESKTOP-K2P82MI MINGW64 /E/myworkDir (feature1)
$ cat f1
this is file 1 line 1
Prajwal Shreesh
121A3060 121A3050

feature1

Avinash@DESKTOP-K2P82MI MINGW64 /E/myworkDir (feature1)
$ git add .
warning: in the working copy of 'f1', LF will be replaced by CRLF the next time Git touches it

Avinash@DESKTOP-K2P82MI MINGW64 /E/myworkDir (feature1)
$ git commit -m "commit through feature1"
[feature1 d1ea1d0] commit through feature1
 1 file changed, 1 insertion(+)

```

```
Avinash@DESKTOP-K2P82MI MINGW64 /E/myworkDir (feature1)
$ git checkout main
switched to branch 'main'
Your branch is ahead of 'origin/main' by 2 commits.
  (use "git push" to publish your local commits)

Avinash@DESKTOP-K2P82MI MINGW64 /E/myworkDir (main)
$ cat f1
this is file 1 line 1
Prajwal Shreesh
121A3060 121A3050

Avinash@DESKTOP-K2P82MI MINGW64 /E/myworkDir (main)
$ cat >>f1
main

Avinash@DESKTOP-K2P82MI MINGW64 /E/myworkDir (main)
$ git add .
warning: in the working copy of 'f1', LF will be replaced by CRLF the next time Git touches it

Avinash@DESKTOP-K2P82MI MINGW64 /E/myworkDir (main)
$ git commit -m "commit through main"
[main ad1bf1a] commit through main
 1 file changed, 1 insertion(+)

Avinash@DESKTOP-K2P82MI MINGW64 /E/myworkDir (main)
$ git checkout feature1
switched to branch 'feature1'
```

```
Avinash@DESKTOP-K2P82MI MINGW64 /E/myworkDir (feature1)
$ git merge main
Auto-merging f1
CONFLICT (content): Merge conflict in f1
Automatic merge failed; fix conflicts and then commit the result.

Avinash@DESKTOP-K2P82MI MINGW64 /E/myworkDir (feature1|MERGING)
$ |
```

- Conflicted file



```
f1 - Notepad
File Edit Format View Help
this is file 1 line 1
Prajwal Shreesh
121A3060 121A3050

<<<<< HEAD
feature1
=====
main
>>>>> main
```

Conflict Resolving :

```
Avinash@DESKTOP-K2P82MI MINGW64 /E/myworkDir (feature1|MERGING)
$ cat f1
this is file 1 line 1
Prajwal Shreesh
121A3060 121A3050

feature1
main

Avinash@DESKTOP-K2P82MI MINGW64 /E/myworkDir (feature1|MERGING)
$ git add .

Avinash@DESKTOP-K2P82MI MINGW64 /E/myworkDir (feature1|MERGING)
$ git commit -m "fixed the conflicts"
[feature1 7591e35] fixed the conflicts

Avinash@DESKTOP-K2P82MI MINGW64 /E/myworkDir (feature1)
$ git checkout main
Switched to branch 'main'
Your branch is ahead of 'origin/main' by 3 commits.
  (use "git push" to publish your local commits)

Avinash@DESKTOP-K2P82MI MINGW64 /E/myworkDir (main)
$ git merge feature1
Updating ad1bf1a..7591e35
Fast-forward
  f1 | 2 ++
  1 file changed, 2 insertions(+)

Avinash@DESKTOP-K2P82MI MINGW64 /E/myworkDir (main)
$ cat f1
this is file 1 line 1
Prajwal Shreesh
121A3060 121A3050

feature1
main
```

CONCLUSION:

Hence, we have successfully complete the experiment of Branching and Merging in Git.

- a. We learned how to create branches in the repositories using git branch command.
- b. Learned how to merge different branches into the main branch.
- c. We came across the problem of conflicting while merging. To tackle the problem the main branch, need to resolve the content in the file to solve the conflict.

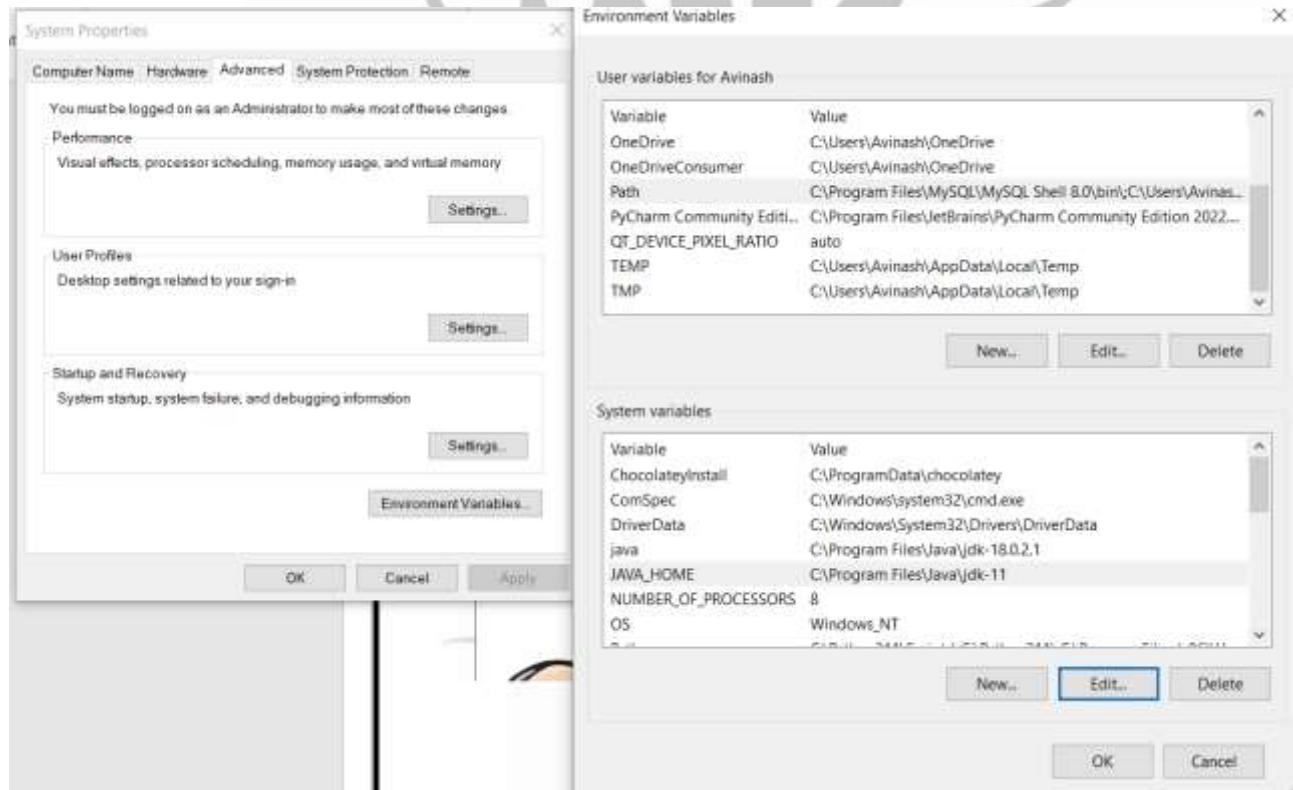
PRAJWAL WAGHMODE
121A3060

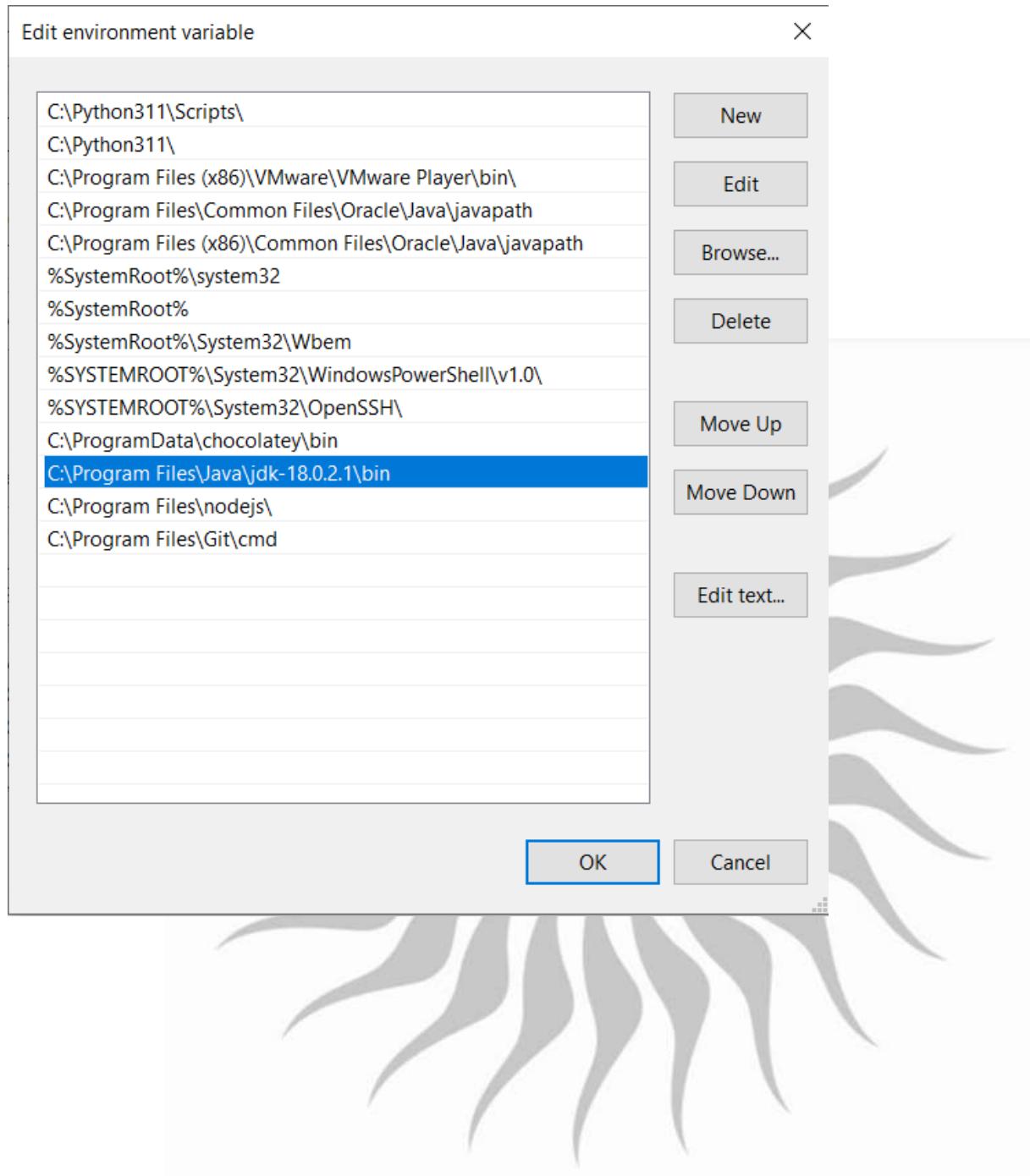
Experiment No: 4

AIM: Installation of Jenkins with Maven/Ant/Gradle, GitHub and Python Plugins to setup a build Job.

THEORY: Jenkins is a free and open source automation server. It helps automate the parts of software development related to building, testing, and deploying, facilitating continuous integration and continuous delivery. It is a server-based system that runs in servlet containers such as Apache Tomcat.

Installation process of Jenkins

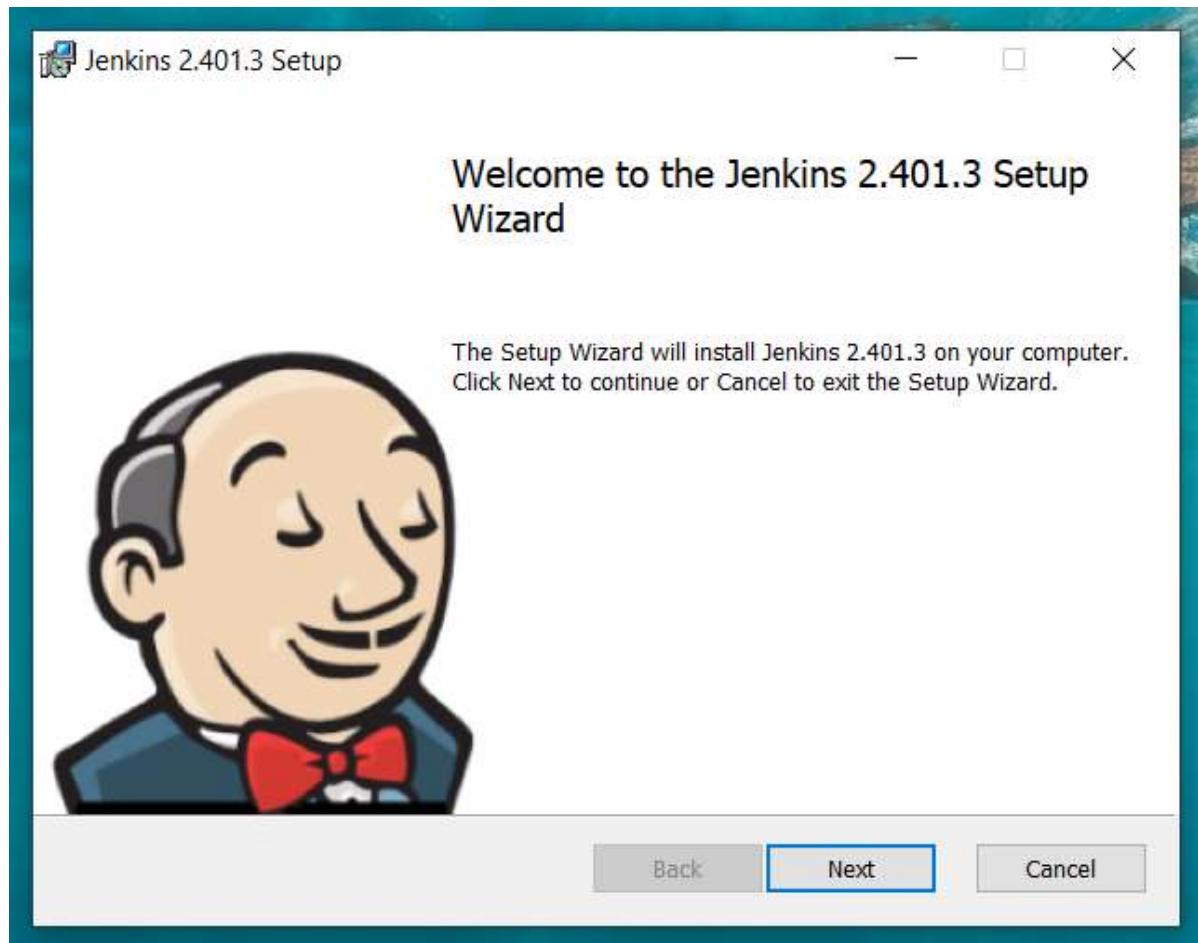




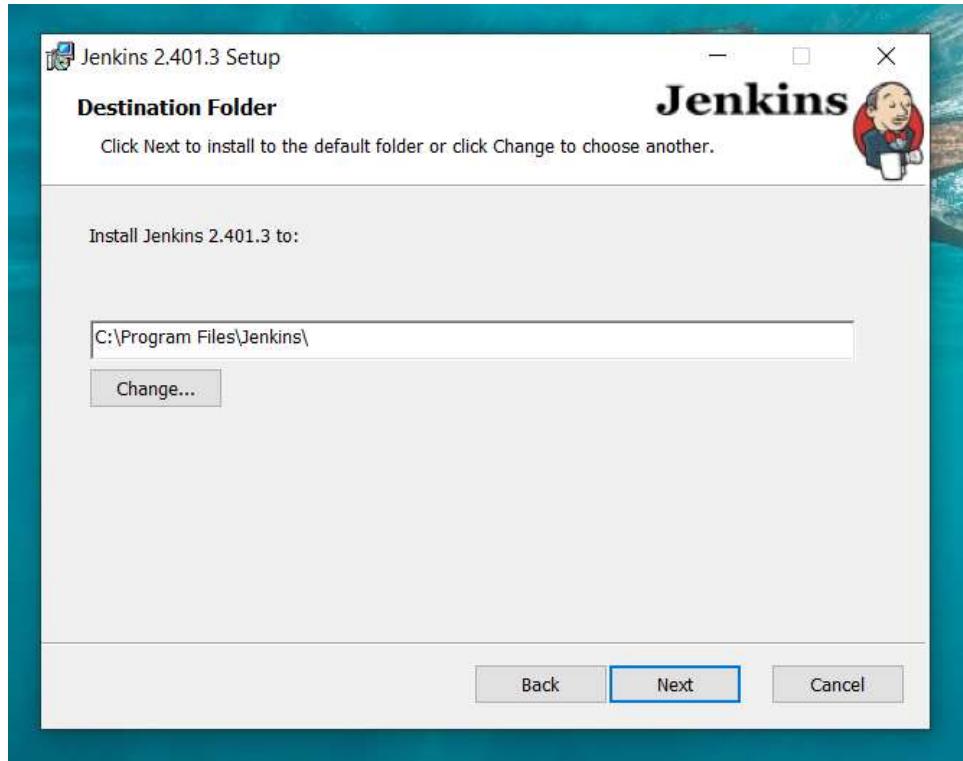
Step 1) Got to <https://www.jenkins.io/download/> and select the platform. In our case Windows

Step 2) Go to download location from local computer and unzip the downloaded package. Double-click on unzipped jenkins.msi. You can also Jenkins using a WAR (Web application ARchive) but that is not recommended.

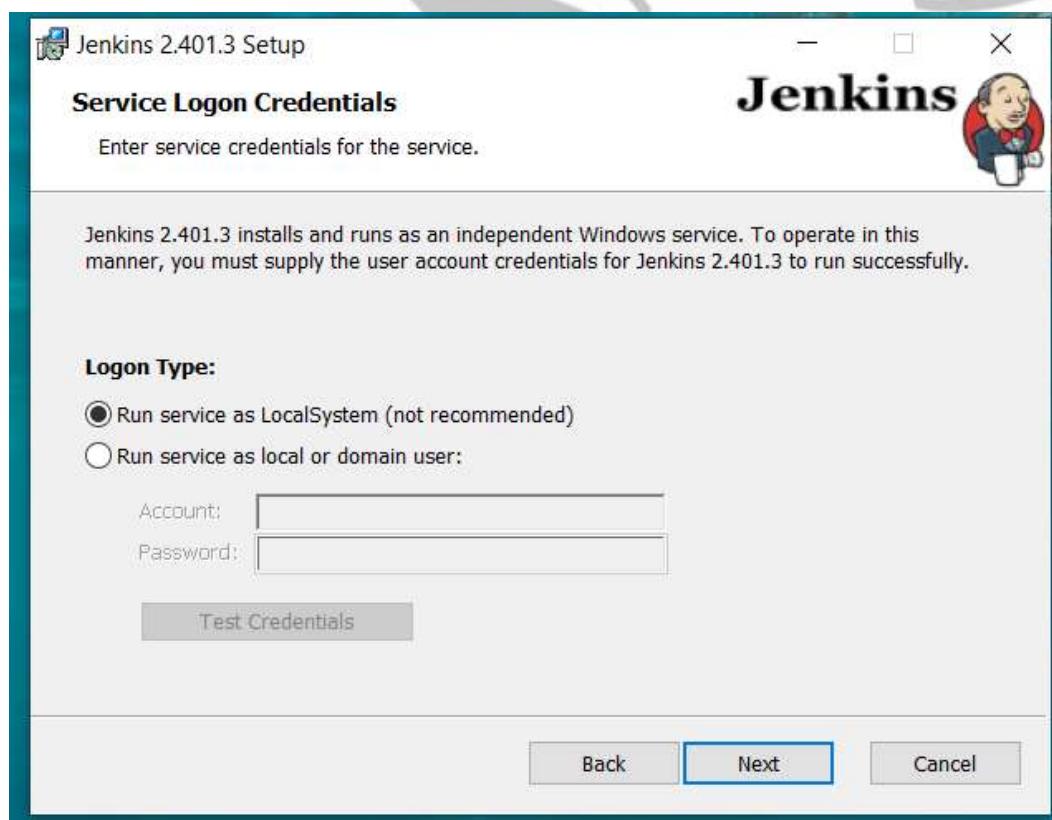
Step 3) In the Jenkins Setup screen, click Next.



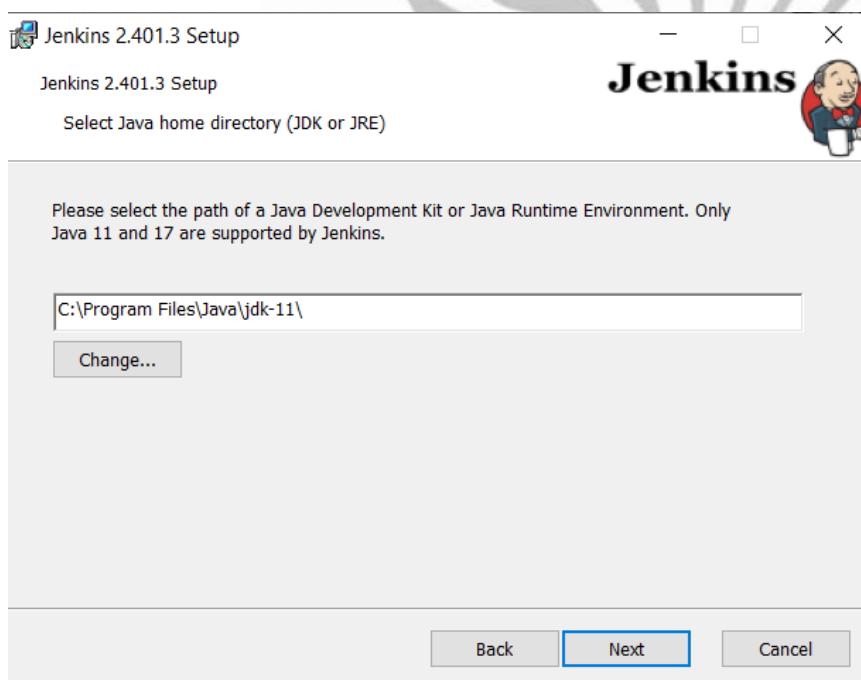
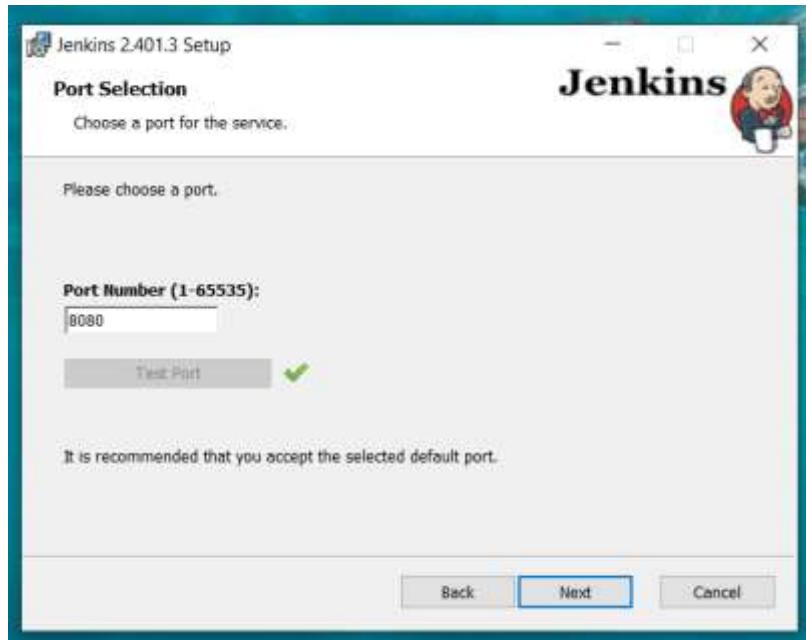
Step 4) Choose the location where you want to have the Jenkins instance installed (default location is C:\Program Files (x86)\Jenkins), then click on Next button.

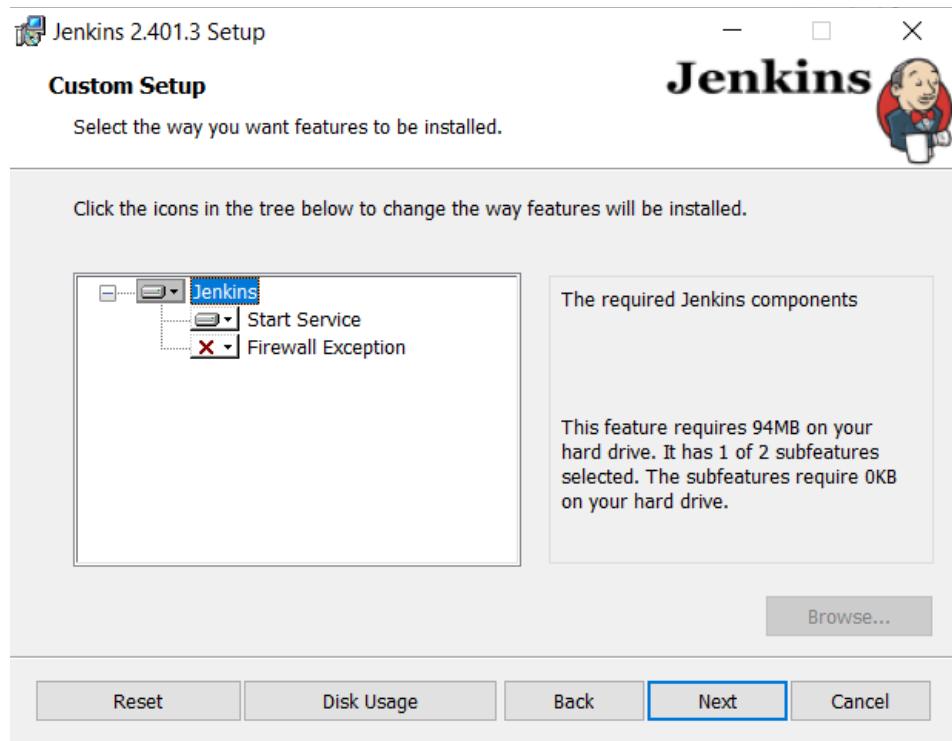


Here, Select Logon type as ' Run service as LocalSystem(Windows equivalent root)

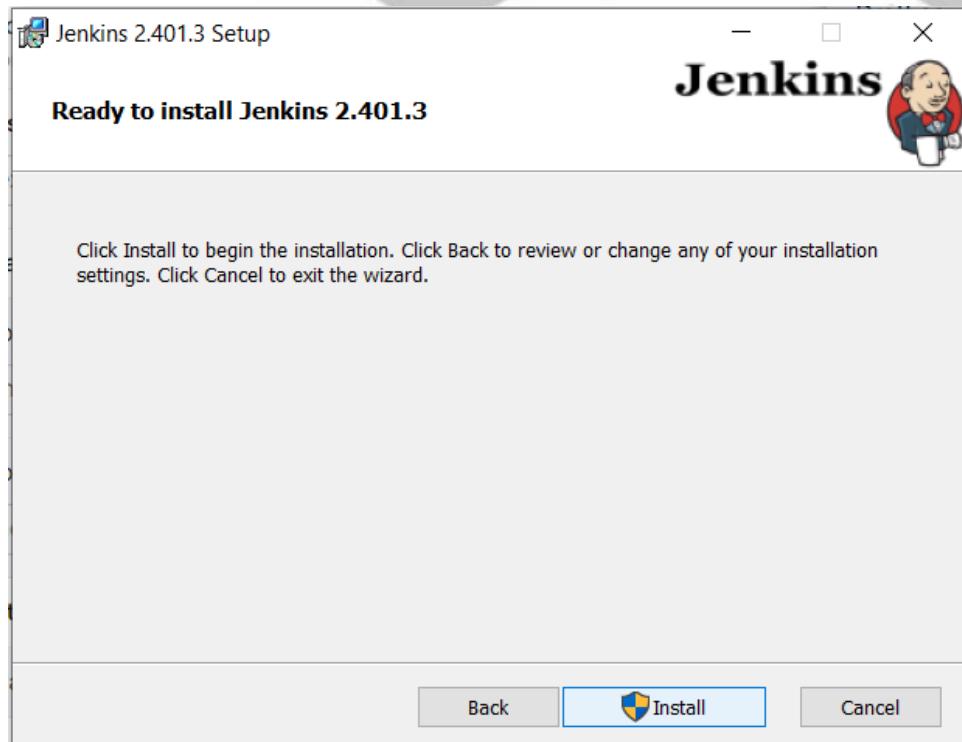


Step 5) Click on the Install button.





Step 6) Once install is complete, click Finish.





Step 7) During the installation process an info panel may pop-up to inform the user that for a complete setup, the system should be rebooted at the end of the current installation. Click on OK button when the Info panel is popping-up:



Please wait while Jenkins is getting ready to work ...

Your browser will reload automatically when Jenkins is ready.

How to Unblock Jenkins?

After completing the Jenkins installation phase, you should proceed further and start its configuration. Next steps will guide you how you can unblock Jenkins application:

Step 1) After completing the Jenkins installation process, a browser tab will pop-up asking for the initial Administrator password. To access Jenkins, you need to go to browse the following path in your web browser. <http://localhost:8080>

Getting Started

Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log ([not sure where to find it?](#)) and this file on the server:

C:\ProgramData\Jenkins\.jenkins\secrets\initialAdminPassword

Please copy the password from either location and paste it below.

Administrator password

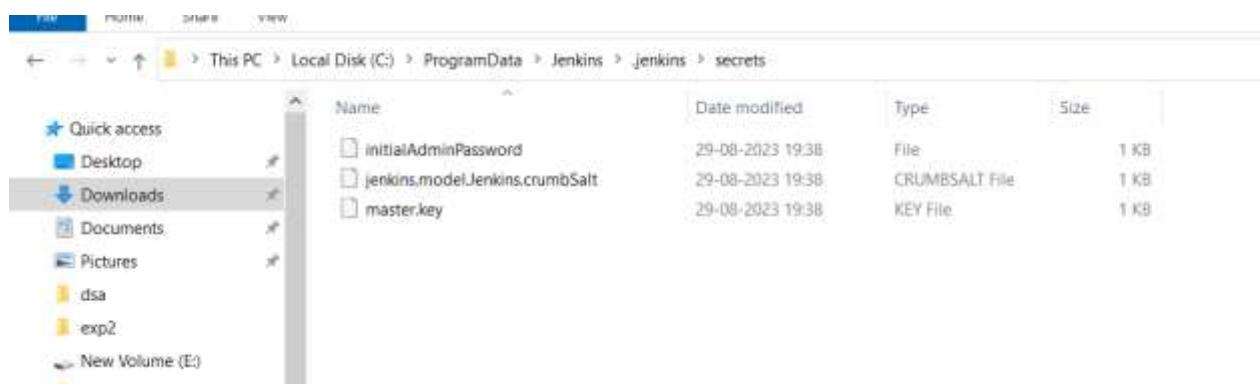
Continue

If you can access the above URL, then it confirms that Jenkins is successfully installed in your system.

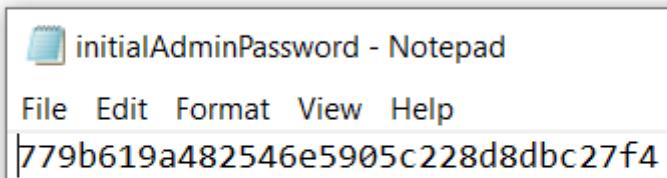
Step 2) The initial Administrator password should be found under the Jenkins installation path (set at Step 4 in Jenkins Installation).

For default installation location to C:\Program Files (x86)\Jenkins, a file called initialAdminPassword can be found under C:\Program Files (x86)\Jenkins\secrets.

However, If a custom path for Jenkins installation was selected, then you should check that location for initialAdminPassword file.



Step 3) Open the highlighted file and copy the content of the initialAdminPassword file.



Step 4) Paste the password it into browser's pop-up tab (<http://localhost:8080/login?form=%2F>) and click on Continue button.

Getting Started

Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log ([not sure where to find it?](#)) and this file on the server:

C:\ProgramData\Jenkins\.jenkins\secrets\initialAdminPassword

Please copy the password from either location and paste it below.

Administrator password

.....

Customize Jenkins

You can also customize your Jenkins environment by below-given steps:

Step 1) Click on the "Install suggested plugins button" so Jenkins will retrieve and install the essential plugins

The screenshot shows the Jenkins 'Getting Started' page. At the top, there's a large heading 'Customize Jenkins' with the subtext 'Plugins extend Jenkins with additional features to support many different needs.' Below this, there are two main options: 'Install suggested plugins' and 'Select plugins to install'. The 'Install suggested plugins' option is highlighted with a blue background and contains the text 'Install plugins the Jenkins community finds most useful.' The 'Select plugins to install' option has a grey background and contains the text 'Select and install plugins most suitable for your needs.' At the bottom of the page, there's a table titled 'Getting Started' listing various Jenkins plugins. The table has four columns: Folders, Pipeline, Git, and LDAP. The Pipeline column includes 'Timestamper', 'GitHub Branch Source', 'Matrix Authorization Strategy', and 'Email Extension'. The Git column includes 'SSH Build Agents'. The LDAP column includes 'OWASP Markup Formatter', 'Workspace Cleanup', 'Ant', 'SSH Build Agents', and 'Email Extension'. The Pipeline column includes 'Build Timeout', 'Ant', 'Gradle', 'Pipeline: GitHub Groovy Libraries', 'Pipeline: Stage View', and 'Mailer'. The Git column includes 'Credentials Binding', 'Gradle', 'Pipeline: Stage View', 'PAM Authentication', and 'Mailer'. A note at the bottom right of the table says '** - required dependency'.

Folders	Pipeline	Git	LDAP
Timestamper	Github Branch Source	SSH Build Agents	OWASP Markup Formatter
Pipeline	Matrix Authorization Strategy	Ant	Workspace Cleanup
Git	SSH Build Agents	Build Timeout	Credentials Binding
LDAP	Email Extension	Gradle	Gradle
		Pipeline: Stage View	Pipeline: Stage View
		Mailer	Mailer

Jenkins will start to download and install all the necessary plugins needed to create new Jenkins Jobs.

Note: You can choose the Option "Select Plugins to Install" and select the plugins you want to install

Step 2) After all suggested plugins were installed, the "Create First Admin User" panel will show up. Fill all the fields with desired account details and hit the "Save and Finish" button.

Getting Started

Create First Admin User

Username

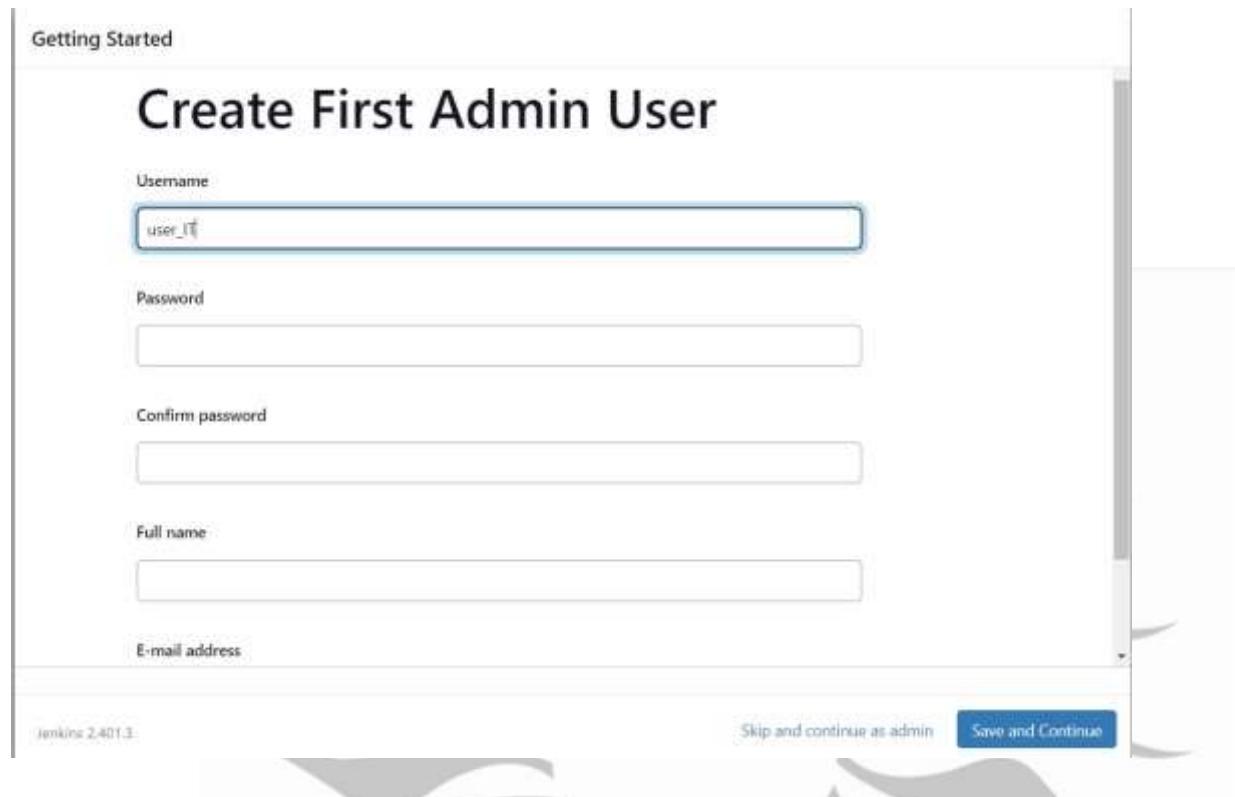
Password

Confirm password

Full name

E-mail address

Jenkins 2.401.3:
Skip and continue as admin



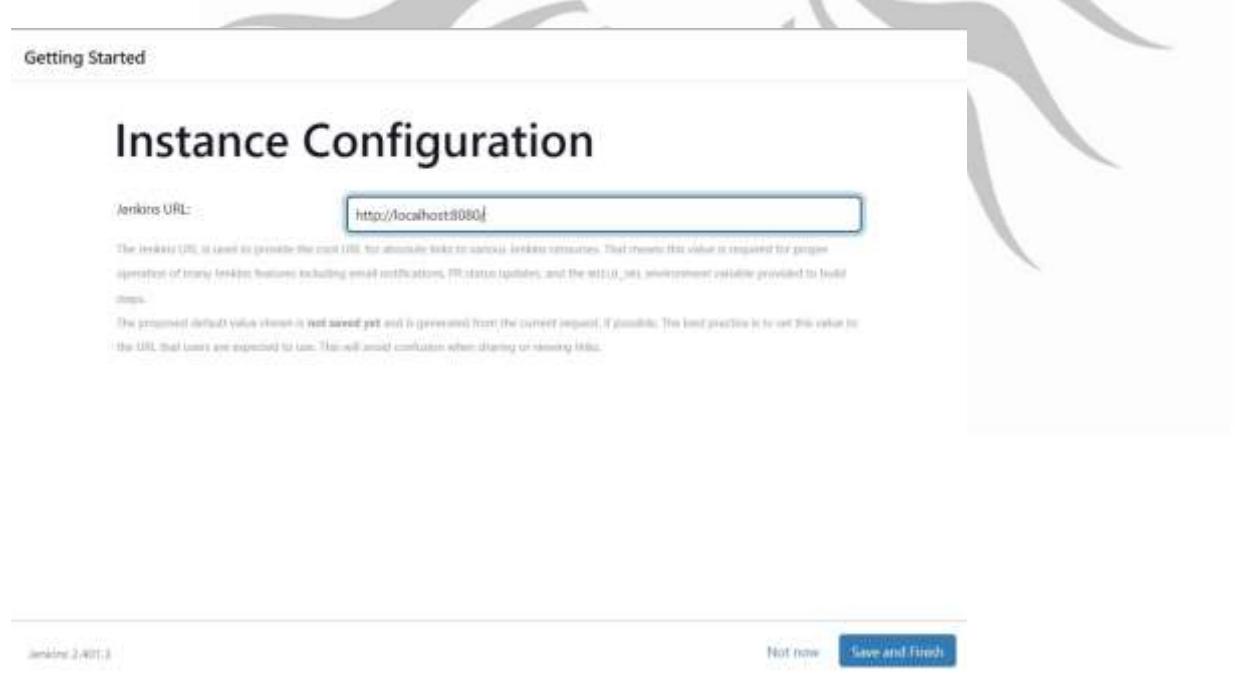
Getting Started

Instance Configuration

Jenkins URL:

The Jenkins URL is used to provide the root URL for absolute links to various Jenkins resources. This means that value is required for proper operation of many Jenkins features including email notifications, PR status updates, and the `RETRY` environment variable provided to build steps.
The proposed default value shown is **not saved yet** and is (presumed) from the current request, if possible. The best practice is to set this value to the URL that users are expected to use. This will avoid confusion when sharing or linking URLs.

Jenkins 2.401.3
Not now



Step 3) Once you have filled the above data, finally it will ask for URL information where you can configure the default instance path for Jenkins. Leave it as it is to avoid any confusion later. However, if another application is already using 8080 port, you can use another port for Jenkins and finally save the settings, and you are done with installation of Jenkins. Hit the "Save and Continue" button:

Getting Started

Jenkins is ready!

Your Jenkins setup is complete.

[Start using Jenkins](#)

We have successfully installed a new Jenkins Server. Hit the "Start using Jenkins" button. Below you can find the Jenkins instance up and run, ready to create first Jenkins jobs:

The screenshot shows the Jenkins dashboard. On the left, there's a sidebar with links: 'Dashboard' (selected), 'New Item', 'People', 'Build History', 'Manage Jenkins', and 'My Views'. Below these are two dropdown menus: 'Build Queue' (No builds in the queue) and 'Build Executor Status' (1 idle, 2 idle). The main content area has a title 'Welcome to Jenkins!'. It says, 'This page is where your Jenkins jobs will be displayed. To get started, you can set up builds or start building a software project.' Below this are three buttons: 'Start building your software project' (Create a job, Set up a distributed build, Set up an agent, Configure a cloud), 'Learn more about distributed builds', and a link to 'Documentation'.

Installation of Maven Integration Plugins:

The screenshot shows the Jenkins plugin search interface. A search bar at the top contains the text "maven". Below it, a table lists the "Maven Integration" plugin by "Build Tools". The table includes columns for "Install", "Name", and "Released". The "Maven Integration" row has a checked checkbox in the "Install" column. The "Name" column shows "Maven Integration 3.23", and the "Released" column shows "24 days ago". A brief description of the plugin follows.

Install	Name	Released
<input checked="" type="checkbox"/>	Maven Integration 3.23 Build Tools This plugin provides a deep integration between Jenkins and Maven. It adds support for automatic triggers between projects depending on SNAPSHOTs as well as the automated configuration of various Jenkins publishers such as Junit.	24 days ago

The screenshot shows the Jenkins "Manage Jenkins > Plugins" page. It displays a list of installed plugins with their status as "Success". The "Maven Integration" plugin is highlighted with a blue background. At the bottom, there are two buttons: "Go back to the top page" and "Restart Jenkins when installation is complete and no jobs are running".

Plugin	Status
Git	Success
GitHub	Success
GitHub Branch Source	Success
Pipeline: GitHub Groovy Libraries	Success
Pipeline Graph Analysis	Success
Pipeline: REST API	Success
Pipeline: Stage View	Success
Git	Success
SSH Build Agents	Success
Matrix Authorization Strategy	Success
PAM Authentication	Success
LDAP	Success
Email Extension	Success
Mailer	Success
Loading plugin extensions...	Success
Javadoc	Success
JSch dependency	Success
Maven Integration	Success
Loading plugin extensions...	Success

The screenshot shows the Jenkins "Manage Jenkins > Plugins" page with the search bar set to "maven". It displays the "Maven Integration plugin 3.23" entry. The "Name" column shows "Maven Integration plugin 3.23" and the "Enabled" column shows a checked checkbox. A brief description of the plugin follows.

Name ↓	Enabled
Ant Plugin 497.v94e7d9fffa_b_9 Adds Apache Ant support to Jenkins Report an issue with this plugin	<input checked="" type="checkbox"/>
Apache HttpComponents Client 4.x API Plugin 4.5.14-208.v438351942757 Bundles Apache HttpComponents Client 4.x and allows it to be used by Jenkins plugins. Report an issue with this plugin	<input checked="" type="checkbox"/>
Bootstrap 5 API Plugin 5.3.0-1 Provides Bootstrap 5 for Jenkins Plugins. Bootstrap is (according to their self-perception) the world's most popular front-end component library to build responsive, mobile-first projects on the web. Report an issue with this plugin	<input checked="" type="checkbox"/>
GitHub API Plugin 1.314-431.v78d72a_3fe4c3 This plugin provides GitHub API for other plugins. Report an issue with this plugin	<input checked="" type="checkbox"/>
GitHub Branch Source Plugin 1732.v3f1889a_c475b_... Multibranch projects and organization folders from GitHub. Maintained by CloudBees, Inc. Report an issue with this plugin	<input checked="" type="checkbox"/>
GitHub plugin 1.37.3 This plugin integrates GitHub to Jenkins. Report an issue with this plugin	<input checked="" type="checkbox"/>
Pipeline: GitHub Groovy Libraries 42.v0739460cda_c4 Allows Pipeline Groovy libraries to be loaded on the fly from GitHub. Report an issue with this plugin	<input checked="" type="checkbox"/>

Conclusion:- Hence we have successfully installed Jenkins

**PRAJWAL WAGHMODE
121A3060**

EXPERIMENT 5

AIM: a) To set up and build a Java, Maven /Ant and Python jobs in Jenkins.

b) To build the pipeline of jobs using Maven / Ant in Jenkins, create a pipeline script to Test and deploy an application over the tomcat server.

THEORY: Jenkins is a free and open-source automation server. It helps automate the parts of software development related to building, testing, and deploying, facilitating continuous integration and continuous delivery. It is a server-based system that runs in servlet containers such as Apache Tomcat.

a) To set up and build Java, Maven /Ant and Python jobs in Jenkins.

Build Steps

The screenshot shows a Jenkins build step configuration for a Windows batch command. The step is titled "Execute Windows batch command". The "Command" field contains the command "echo \"Hello world\"". An "Advanced" dropdown menu is visible at the bottom left. A decorative background of grey flame-like patterns is visible behind the configuration panel.

Console Output

```
Started by user Prajwal
Running as SYSTEM
Building in workspace C:\ProgramData\Jenkins\.jenkins\workspace\prajwal
[prajwal] $ cmd /c call C:\WINDOWS\TEMP\jenkins8010312379099617384.bat

C:\ProgramData\Jenkins\.jenkins\workspace\prajwal>echo "Hello World"
"Hello World"

C:\ProgramData\Jenkins\.jenkins\workspace\prajwal>exit 0
Finished: SUCCESS
```

➤ Steps for Creating and Testing Python job in Jenkins:

1. Install python and set environment settings on our machine.
2. Install Python plugins in Jenkins.

i.e., Manage Jenkins-> Manage Plugins -> Available-> Python Plugin ->select the plugin and click on install without restart.

3. Create a new job as freestyle project in Jenkins.

New Item - > Freestyle Project. Add some optional description.

4. Configure -> Build section, add build steps-> commands to run the py script.

Implicit =>select Execute python script => write some implicit python script here -> save and apply. Goto step 5.

Explicit => write the foll. Build steps, assuming your python script is stored in this location D:\python scripts

d:

cd D:\python scripts

python fact.py

5. Build Now - > Console Output.

Console Output:

The screenshot shows the Jenkins console output for a build. On the left, there is a code editor window displaying a Python script named 'fact.py'. The script contains the following code:a = 15
b = 20

sum = a + b

print(f"Sum of {a} and {b} is {sum}")On the right, the Jenkins console output window displays the execution of this script. It shows the script being run from a Windows command prompt (cmd) in a Jenkins workspace. The output includes the user who started the build, the system it's running as, the workspace path, and the command executed. The script runs successfully, printing the sum of 15 and 20, which is 35. The build is marked as successful.

Started by user Prajwal Waghmode
Running as SYSTEM
Building in workspace C:\ProgramData\Jenkins\.jenkins\workspace\prajwal
[prajwal] \$ cmd /c call C:\Windows\TEMP\jenkins2623956348133198668.bat

C:\ProgramData\Jenkins\.jenkins\workspace\prajwal>E:

E:\>cd jenkins

E:\jenkins>python add.py
Sum of 15 and 20 is 35

E:\jenkins>exit 0
Finished: SUCCESS

➤ Steps for Testing Java job in Jenkins:

1. Install Java and set environment settings
2. Create a new job as freestyle project
3. Configure ->Build section, commands to run the Java program.

Add build steps-> select Windows batch commands

Go to Configure ->Build section, write the foll. Build steps, assuming your java program is stored in this location D:\java program

d:

cd D:\\java program

javac fact.java

java fact

-> save and apply

4. Build Now - > Console Output. Verify the output of your java program

Output:

```
class add {  
  
    public static void main(String[] args) {  
  
        int first = 10;  
        int second = 20;  
  
        int sum = first + second;  
        System.out.println(first + " + " + second + " = " + sum);  
    }  
}
```

```
C:\Windows\System32>cd C:\Users\exam\Desktop\prajwal  
C:\Users\exam\Desktop\prajwal>javac add.java  
C:\Users\exam\Desktop\prajwal>java add  
10 + 20 = 30
```

Build Steps

≡ Execute Windows batch command ?

Command

See [the list of available environment variables](#)

```
cd C:\Users\exam\Desktop\prajwal  
javac add.java  
java add
```

Advanced ▾

Add build step ▾

✓ Console Output

```
Started by user Prajwal  
Running as SYSTEM  
Building in workspace C:\ProgramData\Jenkins\.jenkins\workspace\prajwal  
[prajwal] $ cmd /c call C:\WINDOWS\TEMP\jenkins3832986976953623054.bat  
  
C:\ProgramData\Jenkins\.jenkins\workspace\prajwal>cd C:\Users\exam\Desktop\prajwal  
  
C:\Users\exam\Desktop\prajwal>javac add.java  
  
C:\Users\exam\Desktop\prajwal>java add  
10 + 20 = 30  
  
C:\Users\exam\Desktop\prajwal>exit 0  
Finished: SUCCESS
```

```
30 public class add {  
31  
32     public static void main(String[] args) {  
33  
34         int first = 10;  
35         int second = 20;  
36         int a,b,sum;  
37  
38         int add = first + second;  
39         System.out.println(first + " + " + second + " = " + add);  
40  
41         a=Integer.parseInt(args[0]);  
42         b=Integer.parseInt(args[1]);  
43         sum = a + b;  
44         System.out.println("num 1: " +a);  
45         System.out.println("num 2: " +b);  
46         System.out.println("Sum of use input numbers: " + sum);  
47  
48     }  
49 }
```

```
C:\Windows\System32>cd C:\Users\exam\Desktop\prajwal  
C:\Users\exam\Desktop\prajwal>javac add.java  
C:\Users\exam\Desktop\prajwal>java add 40 50  
10 + 20 = 30  
num 1: 40  
num 2: 50  
Sum of use input numbers: 90  
C:\Users\exam\Desktop\prajwal>
```

Project prajwal

This build requires parameters:

This project is parameterized ?

String Parameter ?	Name ?	Default Value ?	Action
String Parameter ?	Name ?	Default Value ?	Action
a	b	7	Build Cancel
40	50		

✓ Console Output

```
Started by user Prajwal
Running as SYSTEM
Building in workspace C:\ProgramData\Jenkins\.jenkins\workspace\prajwal
[prajwal] $ cmd /c call C:\WINDOWS\TEMP\jenkins5530151451607214543.bat

C:\ProgramData\Jenkins\.jenkins\workspace\prajwal>cd C:\Users\exam\Desktop\prajwal

C:\Users\exam\Desktop\prajwal>javac add.java

C:\Users\exam\Desktop\prajwal>java add
10 + 20 = 30
Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: Index 0 out of bounds for length 0
    at add.main(add.java:12)

C:\Users\exam\Desktop\prajwal>java add 40 50
10 + 20 = 30
num 1: 40
num 2: 50
Sum of use input numbers: 90

C:\Users\exam\Desktop\prajwal>exit 0
Finished: SUCCESS
```

➤ Steps for Building Maven job from the GitHub remote repository into Jenkins

- 1 a. Install Maven and set environment settings. Make sure you've previously set java settings properly
 - 1 b. Install **Maven Integration plugin** if not installed earlier and got to Manage Jenkins -> **Global tool configurations** -> Set Name => Maven <version> and Maven Home => path value of MAVEN_HOME
 2. Build maven project in your machine using mvn commands to learn how to create and build any maven project as shown here:
<https://maven.apache.org/guides/getting-started/maven-in-five-minutes.html>
 3. Create a maven job as Maven Project
 4. In Configure section - >go to **Source Code Management** -> Git - > paste this repo url
=><https://github.com/bushsk/SimpleMavenProject.git> (Assuming our maven project is on Github repository)
- Note:** Ensure that the GitHub Plugin is installed at this point. If it has not been installed, then do install it from Manage Jenkins-> Manage Plugins
5. In Build section -> Write the goals and options as => **clean compile package**. Make sure Root POM is set to pom.xml. -> save and apply
 6. Go to build now -> console output and verify whether the **.jar** file with the given artefact has been created or not inside the given path shown in the console output

Output:

Maven installations ^ Edited

Maven installations

List of Maven installations on this system

Add Maven

Maven Name X

MAVEN_HOME

Install automatically ?

Add Maven

Save Apply

```
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time:  2.183 s
[INFO] Finished at: 2023-08-17T23:41:33-07:00
[INFO] -----
Waiting for Jenkins to finish collecting data
[JENKINS] Archiving C:\Users\exam\my-app\pom.xml to com.mycompany.app/my-app/1.0-SNAPSHOT/my-app-1.0-SNAPSHOT.pom
[JENKINS] Archiving C:\Users\exam\my-app\target\my-app-1.0-SNAPSHOT.jar to com.mycompany.app/my-app/1.0-SNAPSHOT/my-app-1.0-SNAPSHOT.jar
[shubham prajwal hariom sahil maven] $ cmd /c call C:\WINDOWS\TEMP\jenkins3661938945052100173.bat
channel stopped

C:\ProgramData\Jenkins\.jenkins\workspace\shubham prajwal hariom sahil maven>C:

C:\ProgramData\Jenkins\.jenkins\workspace\shubham prajwal hariom sahil maven>cd C:\Users\exam\my-app

C:\Users\exam\my-app>java -cp target/my-app-1.0-SNAPSHOT.jar com.mycompany.app.App
Hello World!

C:\Users\exam\my-app>exit 0
Finished: SUCCESS
```

```
[JENKINS] Recording test results
[INFO]
[INFO] --- jar:3.0.2:jar (default-jar) @ my-app ---
[INFO]
[INFO] --- install:2.5.2:install (default-install) @ my-app ---
[INFO] Installing C:\Users\exam\my-app\target\my-app-1.0-SNAPSHOT.jar to
C:\WINDOWS\system32\config\systemprofile\.m2\repository\com\mycompany\app\my-app\1.0-SNAPSHOT\my-app-1.0-SNAPSHOT.jar
[INFO] Installing C:\Users\exam\my-app\pom.xml to C:\WINDOWS\system32\config\systemprofile\.m2\repository\com\mycompany\app\app-1.0-SNAPSHOT.pom
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time:  2.933 s
[INFO] Finished at: 2023-08-17T23:35:07+07:00
[INFO] -----
Waiting for Jenkins to finish collecting data
[JENKINS] Archiving C:\Users\exam\my-app\pom.xml to com.mycompany.app/my-app/1.0-SNAPSHOT/my-app-1.0-SNAPSHOT.pom
[JENKINS] Archiving C:\Users\exam\my-app\target\my-app-1.0-SNAPSHOT.jar to com.mycompany.app/my-app/1.0-SNAPSHOT/my-app-1.0-SNAPSHOT.jar
channel stopped
Finished: SUCCESS
```

Pre Steps

≡ **Invoke top-level Maven targets** ?

Maven Version

Goals

Advanced ▾ Edited

Add pre-build step ▾

Build

Root POM [?](#)

C:\Users\exam\my-app\pom.xml

Goals and options [?](#)

Advanced ^

MAVEN_OPTS [?](#)

- Incremental build - only build changed modules [?](#)
- Disable automatic artifact archiving [?](#)
- Disable automatic site documentation artifact archiving [?](#)
- Disable automatic fingerprinting of consumed and produced artifacts [?](#)
- Enable triggering of downstream projects [?](#)
 - | Block downstream trigger when building [?](#)

Run regardless of build result

Should the post-build steps run only for successful builds, etc.

Execute Windows batch command ?

Command

See [the list of available environment variables](#)

```
C:  
cd C:\Users\exam\my-app  
java -cp target/my-app-1.0-SNAPSHOT.jar com.mycompany.app.App
```

Advanced ▾

Add post-build step ▾

 maven Public [!\[\]\(9a2ed36d16a93eea7b8313ce7acbad86_img.jpg\) Pin](#) [!\[\]\(dd2614cffaf9bcc42ffc93983a8c43a4_img.jpg\) Unwatch 1](#)

 main  1 branch  0 tags [Go to file](#) [Add file ▾](#) [Code ▾](#)

 prajwal0211 maven project done 00247ff in 38 seconds  1 commit

 my-app maven project done now

Help people interested in this repository understand your project by adding a README. [Add a README](#)

```
#6 > Console Output

[INFO]
[INFO] --- install:2.5.2:install (default-install) @ my-app ---
[INFO] Installing C:\Users\exam\my-app\target\my-app-1.0-SNAPSHOT.jar to
C:\WINDOWS\system32\config\systemprofile\.m2\repository\com\mycompany\app\my-app\1.0-SNAPSHOT\my-a
[INFO] Installing C:\Users\exam\my-app\pom.xml to C:\WINDOWS\system32\config\systemprofile\.m2\rep
app-1.0-SNAPSHOT.pom
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 2.167 s
[INFO] Finished at: 2023-08-18T00:12:00-07:00
[INFO] -----
Waiting for Jenkins to finish collecting data
[JENKINS] Archiving C:\Users\exam\my-app\pom.xml to com.mycompany.app/my-app/1.0-SNAPSHOT/my-app-1
[JENKINS] Archiving C:\Users\exam\my-app\target\my-app-1.0-SNAPSHOT.jar to com.mycompany.app/my-ap
[shubham prajwal hariom sahil maven] $ cmd /c call C:\WINDOWS\TEMP\jenkins454460596491429225.bat
channel stopped

C:\ProgramData\Jenkins\.jenkins\workspace\shubham prajwal hariom sahil maven>C:
C:\ProgramData\Jenkins\.jenkins\workspace\shubham prajwal hariom sahil maven>cd C:\Users\exam\my-a
C:\Users\exam\my-app>java -cp target/my-app-1.0-SNAPSHOT.jar com.mycompany.app.App
Hello World!

C:\Users\exam\my-app>exit 0
Finished: SUCCESS
```

Git Build Data

Revision: 00247ffd0523b4f13385ff1062a7819375354f25

Repository: <https://github.com/prajwal0211/maven.git>

- refs/remotes/origin/main

Built Branches

- refs/remotes/origin/main: Build #6 of Revision 00247ffd0523b4f13385ff1062a7819375354f25 (refs/remotes/origin/main)

b) To build the pipeline of jobs using Maven / Ant in Jenkins, create a pipeline script to Test and deploy an application over the tomcat server.

Steps for Building Maven job from the GitHub remote repository into Jenkins using Jenkins pipeline:

Setup and run apache tomcat server:

1 a. Download apache tomcat server from official [Apache's tomcat website](#) and Install it by extracting and set it to run on port 9090 or any other port since Jenkins is already running on 8080 port and by default tomcat also runs on 8080.

1b. To do the above, go to the apache tomcat directory and find server.xml inside config folder. Find this line by **ctrl+f** “<Connector port="8080”” and replace it with “<Connector port="9090” and save it.

1c. Start the server by running the startup.bat file found inside apache-tomcat-9.0.52\bin folder and goto browser and test if the home page of server is up or not on localhost:9090.

2. Go to Jenkins Dashboard and create a new **pipeline** project.

3. Go to Pipeline Section and choose Pipeline Script and write the following script here:

```
pipeline{
    agent any
    tools {
        maven 'Maven 3.8.1'
    }

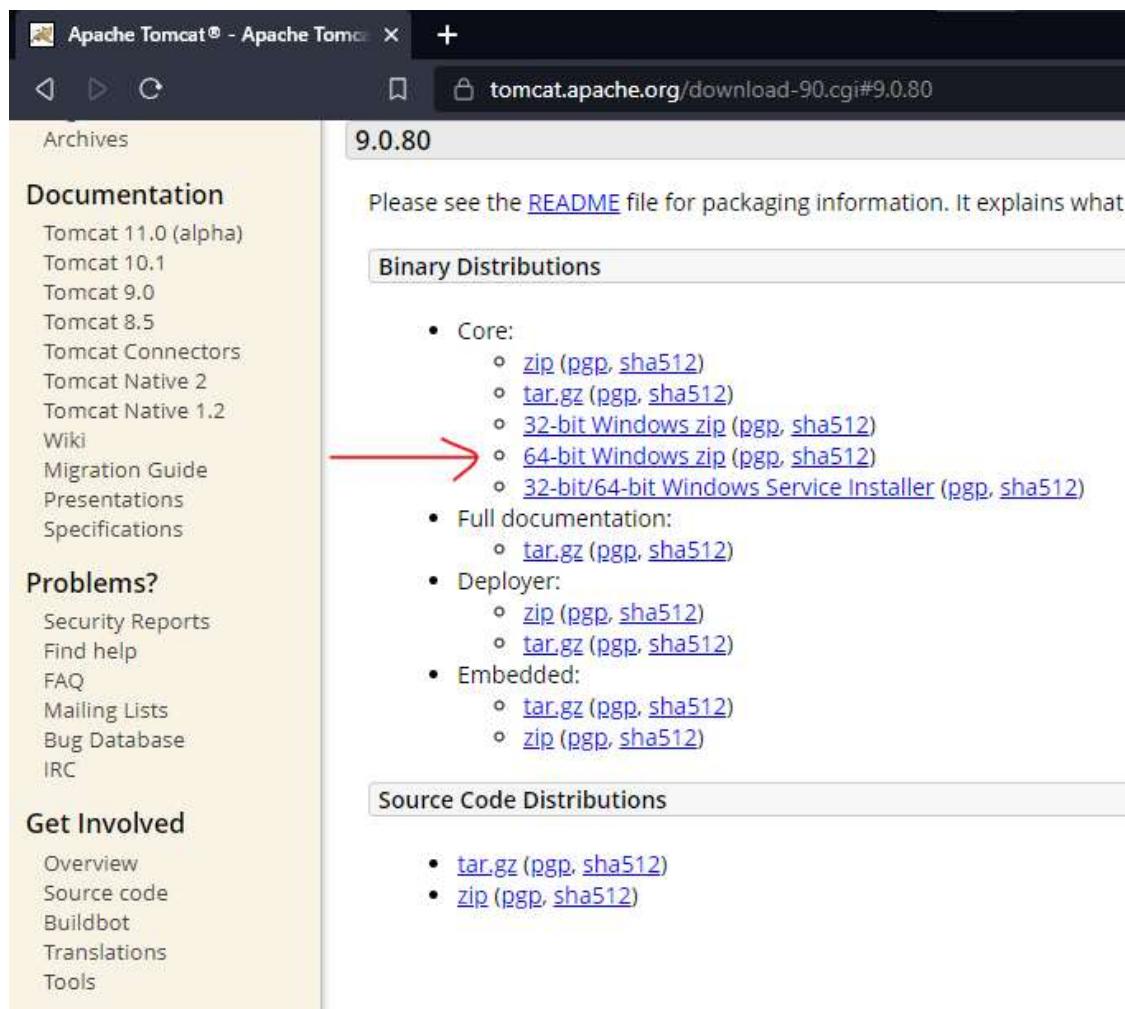
    stages{
        stage("Git Clone"){
            steps
            {
                git 'https://github.com/bushsk/hello-world.git'
            }
        }
        stage("Build"){
            steps
            {
                bat 'mvn clean install'
            }
        }
        stage("Deploy"){
            steps
            {
                bat 'copy C:\\Users\\app\\AppData\\Local\\Jenkins\\.jenkins\\workspace\\
maven_pipeline\\webapp\\target\\webapp.war      C:\\Users\\app\\Downloads\\Tomcat-9\\apache-tomcat-
9.0.52\\webapps'
            }
        }
    }
}
```

{}

4. Got to Build Now and verify if the pipeline build is successful. Go to <http://localhost:9090/webapp/> and verify if the home page of your application is up or not.

Output:

Downloaded and extracted:



The screenshot shows a web browser window with the URL tomcat.apache.org/download-90.cgi#9.0.80. The page displays the Apache Tomcat 9.0.80 download page. On the left, there's a sidebar with links for Documentation, Problems?, and Get Involved. The main content area is titled "9.0.80" and contains a "Binary Distributions" section with a list of download links. A red arrow points from the text above to this section.

Apache Tomcat® - Apache Tomcat X +

tomcat.apache.org/download-90.cgi#9.0.80

Archives 9.0.80

Please see the [README](#) file for packaging information. It explains what

Binary Distributions

- Core:
 - [zip \(pgp, sha512\)](#)
 - [tar.gz \(pgp, sha512\)](#)
 - [32-bit Windows zip \(pgp, sha512\)](#)
 - [64-bit Windows zip \(pgp, sha512\)](#)
 - [32-bit/64-bit Windows Service Installer \(pgp, sha512\)](#)
- Full documentation:
 - [tar.gz \(pgp, sha512\)](#)
- Deployer:
 - [zip \(pgp, sha512\)](#)
 - [tar.gz \(pgp, sha512\)](#)
- Embedded:
 - [tar.gz \(pgp, sha512\)](#)
 - [zip \(pgp, sha512\)](#)

Source Code Distributions

- [tar.gz \(pgp, sha512\)](#)
- [zip \(pgp, sha512\)](#)

Documentation Problems? Get Involved

This PC > Local Disk (C:) > Program Files > apache-tomcat-9.0.80 > conf			
les	Name	Date modified	Type
les (x86)	catalina.policy	23-08-2023 21:59	POLICY
	catalina	23-08-2023 21:59	Proper
	context	23-08-2023 21:59	XML Sc
	jaspic-providers	23-08-2023 21:59	XML Sc
: (D:)	jaspic-providers.xsd	23-08-2023 21:59	XML Sc
: (E:)	logging	23-08-2023 21:59	Proper
:	server	23-08-2023 21:59	XML Sc
	tomcat-users	23-08-2023 21:59	XML Sc
	tomcat-users.xsd	23-08-2023 21:59	XML Sc
	web	23-08-2023 21:59	XML Sc

```
server.xml ●
C: > Program Files > apache-tomcat-9.0.80 > conf > server.xml
57     <Executor name="tomcatThreadPool" namePrefix="catalina-exec-"
58         maxThreads="150" minSpareThreads="4"/>
59     -->
60
61
62     <!-- A "Connector" represents an endpoint by which requests are received
63         and responses are returned. Documentation at :
64             Java HTTP Connector: /docs/config/http.html
65             Java AJP Connector: /docs/config/ajp.html
66             APR (HTTP/AJP) Connector: /docs/apr.html
67             Define a non-SSL/TLS HTTP/1.1 Connector on port 8080
68     -->
69     <Connector port="9090" protocol="HTTP/1.1"
70         connectionTimeout="20000"
71         redirectPort="8443"
72         maxParameterCount="1000"
73         />
74     <!-- A "Connector" using the shared thread pool-->
75     <!--
76     <Connector executor="tomcatThreadPool"
```

```
C:\Program Files\apache-tomcat-9.0.80>cd bin  
C:\Program Files\apache-tomcat-9.0.80\bin>startup.bat  
Using CATALINA_BASE: "C:\Program Files\apache-tomcat-9.0.80"  
Using CATALINA_HOME: "C:\Program Files\apache-tomcat-9.0.80"  
Using CATALINA_TMPDIR: "C:\Program Files\apache-tomcat-9.0.80\temp"  
Using JRE_HOME: "C:\Program Files\Java\jdk-11"  
Using CLASSPATH: "C:\Program Files\apache-tomcat-9.0.80\bin\bootstrap.jar;C:\Program Files\apache-tomcat-9.0.80\lib\tomcat-juli.jar"  
Using CATALINA_OPTS: ""  
C:\Program Files\apache-tomcat-9.0.80\bin>
```

```
Tomcat  
directory [C:\Program Files\apache-tomcat-9.0.80\work\Catalina\localhost\host-manager] for context [/host-manager]  
29-Aug-2023 21:22:54.537 SEVERE [main] org.apache.jasper.EmbeddedServletOptions.<init> The scratchDir you specified: [C:\Program Files\apache-tomcat-9.0.80\work\Catalina\localhost\host-manager] is unusable.  
29-Aug-2023 21:22:54.538 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web application in directory [C:\Program Files\apache-tomcat-9.0.80\webapps\host-manager] has finished in [46] ms  
29-Aug-2023 21:22:54.539 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deploying web application directory [C:\Program Files\apache-tomcat-9.0.80\webapps\manager]  
29-Aug-2023 21:22:54.548 WARNING [main] org.apache.catalina.core.StandardContext.postWorkDirectory Failed to create work directory [C:\Program Files\apache-tomcat-9.0.80\work\Catalina\localhost\manager] for context [/manager]  
29-Aug-2023 21:22:54.577 SEVERE [main] org.apache.jasper.EmbeddedServletOptions.<init> The scratchDir you specified: [C:\Program Files\apache-tomcat-9.0.80\work\Catalina\localhost\manager] is unusable.  
29-Aug-2023 21:22:54.579 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web application in directory [C:\Program Files\apache-tomcat-9.0.80\webapps\manager] has finished in [40] ms  
29-Aug-2023 21:22:54.580 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deploying web application directory [C:\Program Files\apache-tomcat-9.0.80\webapps\ROOT]  
29-Aug-2023 21:22:54.586 WARNING [main] org.apache.catalina.core.StandardContext.postWorkDirectory Failed to create work directory [C:\Program Files\apache-tomcat-9.0.80\work\Catalina\localhost\ROOT] for context []  
29-Aug-2023 21:22:54.612 SEVERE [main] org.apache.jasper.EmbeddedServletOptions.<init> The scratchDir you specified: [C:\Program Files\apache-tomcat-9.0.80\work\Catalina\localhost\ROOT] is unusable.  
29-Aug-2023 21:22:54.615 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web application in directory [C:\Program Files\apache-tomcat-9.0.80\webapps\ROOT] has finished in [35] ms  
29-Aug-2023 21:22:54.622 INFO [main] org.apache.coyote.AbstractProtocol.start Starting ProtocolHandler ["http-nio-9090"]  
29-Aug-2023 21:22:54.969 INFO [main] org.apache.catalina.startup.Catalina.start Server startup in [1816] milliseconds
```

HTTP Status 500 – Internal Server Error

Type Exception Report

Message org.apache.jasper.JasperException: java.lang.ClassNotFoundException: org.apache.jsp.index_jsp

Description The server encountered an unexpected condition that prevented it from fulfilling the request.

Exception

```
org.apache.jasper.JasperException: org.apache.jasper.JasperException: java.lang.ClassNotFoundException: org.apache.jsp.index_jsp
        at org.apache.jasper.servlet.JspServletWrapper.handleJspException(JspServletWrapper.java:589)
        at org.apache.jasper.servlet.JspServletWrapper.service(JspServletWrapper.java:425)
        at org.apache.jasper.servlet.JspServlet.serviceJspFile(JspServlet.java:379)
        at org.apache.jasper.servlet.JspServlet.service(JspServlet.java:327)
        at javax.servlet.http.HttpServlet.service(HttpServlet.java:623)
        at org.apache.tomcat.websocket.server.WsFilter.doFilter(WsFilter.java:51)
```

Root Cause

```
org.apache.jasper.JasperException: java.lang.ClassNotFoundException: org.apache.jsp.index_jsp
        at org.apache.jasper.servlet.JspServletWrapper.getServlet(JspServletWrapper.java:197)
        at org.apache.jasper.servlet.JspServletWrapper.service(JspServletWrapper.java:413)
        at org.apache.jasper.servlet.JspServlet.serviceJspFile(JspServlet.java:379)
        at org.apache.jasper.servlet.JspServlet.service(JspServlet.java:327)
        at javax.servlet.http.HttpServlet.service(HttpServlet.java:623)
        at org.apache.tomcat.websocket.server.WsFilter.doFilter(WsFilter.java:51)
```

Root Cause

```
java.lang.ClassNotFoundException: org.apache.jsp.index_jsp
        at java.base/java.net.URLClassLoader.findClass(URLClassLoader.java:476)
        at org.apache.jasper.servlet.JasperLoader.loadClass(JasperLoader.java:129)
        at org.apache.jasper.servlet.JasperLoader.loadClass(JasperLoader.java:58)
        at org.apache.jasper.servlet.JspServletWrapper.getServlet(JspServletWrapper.java:192)
        at org.apache.jasper.servlet.JspServletWrapper.service(JspServletWrapper.java:413)
        at org.apache.jasper.servlet.JspServlet.serviceJspFile(JspServlet.java:379)
        at org.apache.jasper.servlet.JspServlet.service(JspServlet.java:327)
        at javax.servlet.http.HttpServlet.service(HttpServlet.java:623)
        at org.apache.tomcat.websocket.server.WsFilter.doFilter(WsFilter.java:51)
```

Note The full stack trace of the root cause is available in the server logs.

Definition

Pipeline script

Script ?

```

1 * pipeline{
2+ agent any tools {
3  maven 'Maven 3.8.1'
4 }
5
6+ stages{
7+ stage("Git Clone"){ steps
8 {
9  git 'https://github.com/prajwal0211/DevOps.git'
10 }
11 }
12+ stage("Build"){ steps
13 {
14  bat 'mvn clean install'
15 }
16 }
17+ stage("Deploy"){ steps
18 {
19  bat 'copy C:\\\\Users\\\\app\\\\AppData\\\\Local\\\\Jenkins\\\\workspace\\\\maven_pipeline\\\\webapp\\\\target\\\\webapp.warC:\\\\Program Files\\\\apache-tomcat-9.0.44\\\\webapps'
20 }
21 }
22 }
23 }
24 }
25

```

try sample Pipeline... ▾

Use Groovy Sandbox ?

✖ Console Output

Started by user Prajwal Waghmode

```

org.codehaus.groovy.control.MultipleCompilationErrorsException: startup failed:
WorkflowScript: 4: Tool type "maven" does not have an install of "Maven 3.8.1" configured - did you mean "null"? @ line 4, offset 16
    maven 'Maven 3.8.1'
    ^

```

1 error

```

at org.codehaus.groovy.control.ErrorCollector.failIfErrors(ErrorCollector.java:309)
at org.codehaus.groovy.control.CompilationUnit.applyToPrimaryClassNodes(CompilationUnit.java:1107)
at org.codehaus.groovy.control.CompilationUnit.doPhaseOperation(CompilationUnit.java:624)
at org.codehaus.groovy.control.CompilationUnit.processPhaseOperations(CompilationUnit.java:602)
at org.codehaus.groovy.control.CompilationUnit.compile(CompilationUnit.java:579)
at groovy.lang.GroovyClassLoader.doParseClass(GroovyClassLoader.java:323)
at groovy.lang.GroovyClassLoader.parseClass(GroovyClassLoader.java:293)
at org.jenkinsci.plugins.scriptsecurity.sandbox.groovy.GroovySandbox$Scope.parse(GroovySandbox.java:163)
at org.jenkinsci.plugins.workflow.cps.CpsGroovyShell.doParse(CpsGroovyShell.java:190)
at org.jenkinsci.plugins.workflow.cps.CpsGroovyShell.reparse(CpsGroovyShell.java:175)
at org.jenkinsci.plugins.workflow.cps.CpsFlowExecution.parseScript(CpsFlowExecution.java:580)
at org.jenkinsci.plugins.workflow.cps.CpsFlowExecution.start(CpsFlowExecution.java:526)

```

Conclusion: - Hence we have successfully built python, java and maven jobs using Jenkins. Also, we have created a Jenkins pipeline to deploy tests and deploy an application over the tomcat server.

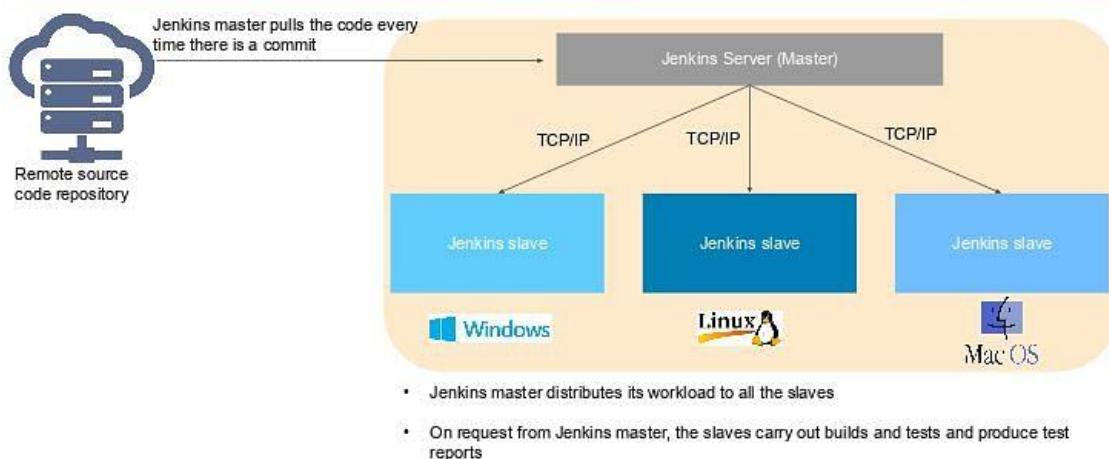
PRAJWAL WAGHMODE
121A3060

EXPERIMENT 6

AIM: To study Jenkins Master-Slave Architecture to scale your Jenkins standalone implementation by implementing slave nodes.

THEORY: Jenkins is a free and open source automation server. It helps automate the parts of software development related to building, testing, and deploying, facilitating continuous integration and continuous delivery. It is a server-based system that runs in servlet containers such as Apache Tomcat.

Jenkins Master-Slave Architecture



simplilearn

©Simplilearn. All rights reserved.

As you can see in the diagram provided above, on the left is the Remote source code repository. The Jenkins server accesses the master environment on the left side and the master environment can push down to multiple other Jenkins Slave environments to distribute the workload.

That lets you run multiple builds, tests, and product environment across the entire architecture. Jenkins Slaves can be running different build versions of the code for different operating systems and the server Master controls how each of the builds operates.

Supported on a master-slave architecture, Jenkins comprises many slaves working for a master. This architecture - the Jenkins Distributed Build - can run identical test cases in different environments. Results are collected and combined on the master node for monitoring.

Steps for Adding Managed Node / Slave:

The screenshot shows the Jenkins dashboard with the following navigation path: Dashboard > Manage Jenkins. The 'Manage Jenkins' option is highlighted with a light gray background. Other options visible include 'New Item', 'People', 'Build History', and 'My Views'. A search bar at the top right says 'Search (CTRL+K)'.

System Configuration

The screenshot shows the 'System Configuration' section of Jenkins. It includes the following items:

- System**: Configure global settings and paths.
- Tools**: Configure tools, their locations and automatic installers.
- Plugins**: Add, remove, disable or enable plugins that can extend the functionality of Jenkins. A red badge indicates 45 available updates.
- Nodes and Clouds**: Add, remove, control and monitor the various nodes that Jenkins runs jobs on.

Nodes

S	Name	Architecture	Clock Difference	Free Disk Space	Free Swap Space	Free Temp Space	Response Time
1	Built-In Node	Windows 10 (amd64)	In sync	16.21 GB	10.11 GB	16.21 GB	0ms
2	itachi		N/A	N/A	N/A	N/A	N/A
3	Replica		N/A	N/A	N/A	N/A	N/A
4	Replica1		N/A	N/A	N/A	N/A	N/A
Data obtained		11 min	11 min	11 min	11 min	11 min	11 min

Step 1: Create a workspace by creating a folder anywhere in your system

Name	Date modified	Type	Size
slaveData	9/6/2023 9:44 AM	File folder	
slaveData1	9/8/2023 11:38 AM	File folder	
jenkins.0.err	9/1/2023 11:25 AM	Text Document	6,345 KB
jenkins.err	9/8/2023 9:03 AM	Text Document	3,581 KB
jenkins	5/31/2023 7:41 AM	Application	606 KB
jenkins.exe	5/31/2023 10:47 AM	Configuration Sou...	1 KB
jenkins.out	9/8/2023 9:03 AM	Text Document	1 KB
Jenkins	5/31/2023 10:32 AM	war	96,058 KB
jenkins.wrapper	Type: Text Document Size: 364 bytes Date modified: 9/8/2023 9:03 AM	Text Document	10 KB
jenkins		XML Source File	3 KB
jenkins.xml	9/1/2023 11:08 AM	backup	3 KB
jenkins.xml.backup_1	9/1/2023 11:17 AM	BACKUP_1 File	3 KB

Step 2: Create a new node with the following configuration:

 Jenkins

Dashboard > Manage Jenkins > Nodes > New node

New node

Node name

Type Permanent Agent

Adds a plain, permanent agent to Jenkins. This is called "permanent" because Jenkins doesn't provide higher level of integration with these agents, such as dynamic provisioning. Select this type if no other agent types apply — for example such as when you are adding a physical computer, virtual machines managed outside Jenkins, etc.

Copy Existing Node

Create

Name

Description
[Plain text] [Preview](#)

Number of executors

Remote root directory

Labels

Usage

Launch method

Launch method [?](#)

Launch agent by connecting it to the controller

Disable WorkDir [?](#)

Custom WorkDir path [?](#)Internal data directory [?](#)

remoting

Fail if workspace is missing [?](#)

Use WebSocket [?](#)

Advanced 

Availability [?](#)

Keep this agent online as much as possible

Node Properties

Disable deferred wipeout on this node [?](#)

Environment variables

Tool Locations

Save

Step 3 : Click on launch to download Jenkins agent.jar and click on agent.jar to download it

Agent Replica

Master Replica

Run from agent command line: (Unix)

```
curl -s0 http://localhost:8080/jnlpJars/agent.jar
java -jar agent.jar -jnlpUrl http://localhost:8080/computer/Replica/jenkins-agent.jnlp -workDir "C:\Program Files\Jenkins\slaveData1"
```

Run from agent command line: (Windows)

```
curl.exe -s0 http://localhost:8080/jnlpJars/agent.jar
java -jar agent.jar -jnlpUrl http://localhost:8080/computer/Replica/jenkins-agent.jnlp -workDir "C:\Program Files\Jenkins\slaveData1"
```

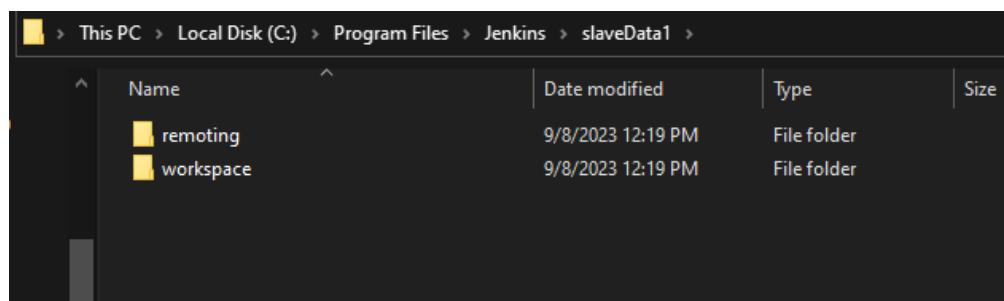
Note: PowerShell users must use curl.exe instead of curl because curl is a default PowerShell cmdlet alias for Invoke-WebRequest.

Labels

Distributed

Projects tied to Replica

None



Step 4: Run the following commands being on the same file location where you download above files and observe the status as connected.

```
Z:\>curl.exe -s0 http://localhost:8080/jnlpJars/agent.jar & java -jar agent.jar -jnlpUrl http://localhost:8080/computer/Replica1/jenkins-agent.jnlp -workDir "C:\Program Files\Jenkins\slaveData1"
Microsoft Windows [Version 10.0.19045.3324]
(c) Microsoft Corporation. All rights reserved.

Z:\>java -jar agent.jar -jnlpUrl http://localhost:8080/computer/Replica1/jenkins-agent.jnlp -workDir "C:\Program Files\Jenkins\slaveData1"
Sep 08, 2023 11:44:24 AM org.jenkinsci.remoting.engine.WorkDirManager initializeWorkDir
INFO: Using C:\Program Files\Jenkins\slaveData1\remoting as a remoting work directory
Sep 08, 2023 11:44:24 AM org.jenkinsci.remoting.engine.WorkDirManager setupLogging
INFO: Both error and output logs will be printed to C:\Program Files\Jenkins\slaveData1\remoting
Sep 08, 2023 11:44:24 AM hudson.remoting.jnlp.Main createEngine
INFO: Setting up agent: Replica1
Sep 08, 2023 11:44:24 AM hudson.remoting.Engine startEngine
INFO: Using Remoting version: 3107.v665000b_51092
Sep 08, 2023 11:44:24 AM org.jenkinsci.remoting.engine.WorkDirManager initializeWorkDir
INFO: Using C:\Program Files\Jenkins\slaveData1\remoting as a remoting work directory
Sep 08, 2023 11:44:24 AM hudson.remoting.jnlp.Main$Cuilistener status
INFO: WebSocket connection open
Sep 08, 2023 11:44:24 AM hudson.remoting.jnlp.Main$Cuilistener status
INFO: Connected
```

Step 5 : Test your slave node

Agent Replica1

master replica

Agent is connected.

Labels

[node1](#)

Projects tied to Replica1

None

Step 6: Create a new job / item and select the label expression of your slave node under 'Restrict where this project can be run'

General

Description

[Plain text] [Preview](#)

- Discard old builds [?](#)
- GitHub project
- This project is parameterized [?](#)
- Throttle builds [?](#)
- Execute concurrent builds if necessary [?](#)
- Restrict where this project can be run [?](#)

Label Expression [?](#)

Distributed

[Label Distributed](#) matches 2 nodes. Permissions or other restrictions provided by plugins ma

[Advanced](#) ▾

Step 7 : Build the job and verify if it has executed on new node .

jt



Console Output

```
Started by user Prajwal
Running as SYSTEM
Building remotely on Replica1 (node1) in workspace C:\Program Files\Jenkins\slaveData\workspace\slavejob
Finished: SUCCESS
```

Conclusion: Thus we have successfully completed the execution of the experiment to study Jenkins Master-Slave Architecture to scale your Jenkins standalone implementation by implementing slave nodes.

Prajwal Waghmode**121A3060****Experiment No: 7**

AIM: a) To setup an Eclipse IDE and Run Selenium Tests Using Maven.

b) To Setup and Run Selenium Tests in Jenkins Using Maven.

THEORY: Jenkins is a free and open source automation server. It helps automate the parts of software development related to building, testing, and deploying, facilitating continuous integration and continuous delivery. It is a server-based system that runs in servlet containers such as Apache Tomcat. Selenium and TestNG are popular tools in the field of automated software testing:

Selenium:

1. Selenium is an open-source automation testing framework primarily used for web application testing.
2. It supports various programming languages, making it versatile for a wide range of developers.
3. Selenium allows testers to write scripts that interact with web elements, enabling functional and regression testing.
4. Its core component, Selenium WebDriver, communicates with web browsers like Chrome, Firefox, and Edge.
5. Selenium is known for its cross-browser compatibility and extensive community support.

TestNG:

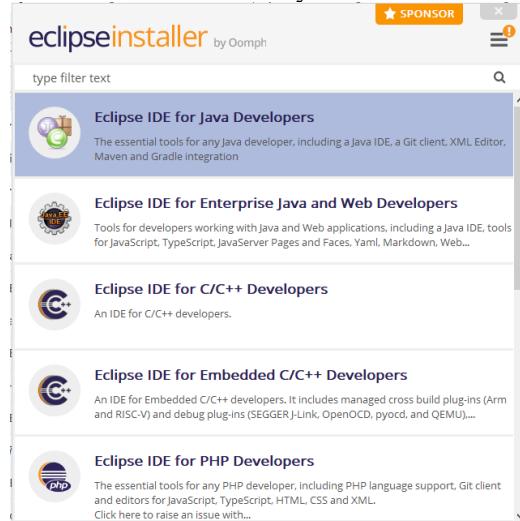
1. TestNG is a testing framework, primarily for Java, that covers unit, functional, and integration testing needs.
2. It offers advanced test configuration, parallel execution, and reporting features.
3. TestNG supports annotations for test setup and execution control, making tests highly configurable.
4. It allows grouping of tests, parameterization, and test dependencies for better test management.
5. TestNG generates detailed test reports, aiding in identifying issues and failures in test suites.

When used together, Selenium and TestNG provide a robust and flexible framework for automating and managing web application testing. Selenium handles browser interactions, while TestNG manages test organization, parallel execution, and reporting, ensuring the quality and reliability of software applications.

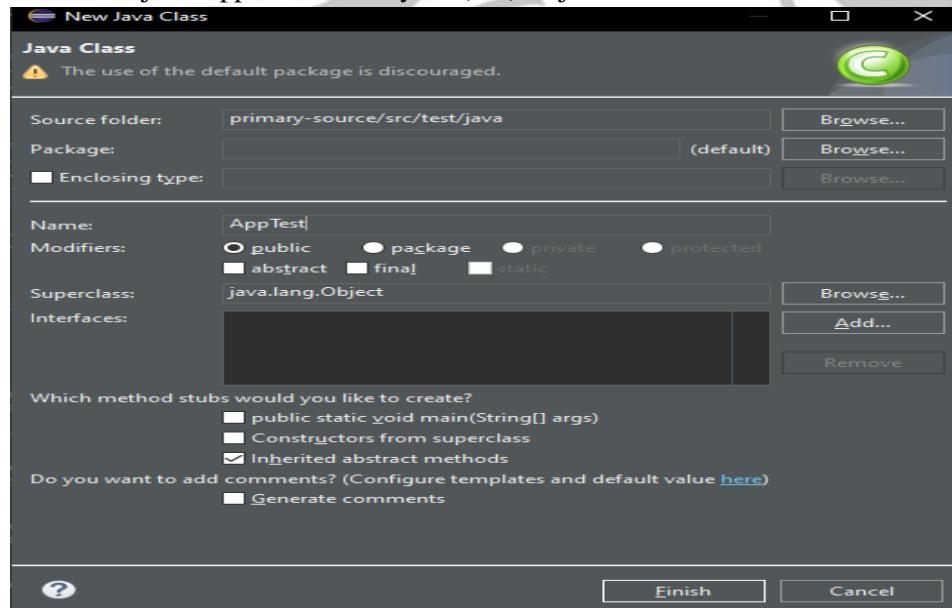
a) To setup an Eclipse IDE and Run Selenium Tests Using Maven.

Steps for Writing Selenium Test Cases in Eclipse using Maven:

- Download and install any latest version of Eclipse IDE from <https://www.eclipse.org/downloads/>



- Create your first maven project.
- Add a new java AppTest class in your \src\test\java. Make sure to that filename should have a keyword „Test“.



- Add the TestNG dependency under dependencies tag in pom.xml <dependencies> visit <https://mvnrepository.com/artifact/org.testng/testng/7.4.0> to copy the foll. dependency tag.

```
<dependency>
    <groupId>org.testng</groupId>
    <artifactId>testng</artifactId>
    <version>7.4.0</version>
    <scope>test</scope>
</dependency>
```

- Write a sample test case using TestNG to print “Hello world” by annotating @Test and left click on red cross when shown as error to import package import org.testng.annotations.Test and similarly write some valid / suitable test cases.

```
public class TestHelloworld {
```

```

    @Test
    public void testhelloworld()
    {
        System.out.println("Hello world test 1 ");
    }
}

```

6. Save and Run as Maven test.
7. Verify if the build is successful or not in the console output.
8. Now add Selenium dependency under dependencies tag in your pom.xml. Visit <https://mvnrepository.com/artifact/org.seleniumhq.selenium/selenium-chrome-driver> to copy the foll dependency tag.

```

<dependency>
    <groupId>org.seleniumhq.selenium</groupId>
    <artifactId>selenium-chrome-driver</artifactId>
    <version>4.0.0-alpha-6</version>
</dependency>

```

9. Download a web driver for chrome similar to the version of your chrome browser and extract chrome driver on some suitable location in your drive. <https://sites.google.com/a/chromium.org/chromedriver/downloads>. *Ensure that it should be of same version as that of your Chrome browser. To Check chrome version => Go in Help -> About Google chrome*
10. Add foll. test case using selenium chrome web drivers : Under this test, we will automate the following scenarios:
 - a. Invoke Google Chrome browser.
 - b. Open URL: <http://www.javatpoint.com>
 - c. Click on the Search text box.
 - d. Type the value "javatpoint tutorials"
 - e. Click on the Search button.

```

@Test
public void testhelloworld2()
{
    System.setProperty("webdriver.chrome.driver","C:\\\\Users\\\\app\\\\Downloads\\\\chromedriver\\\\chromedriver.exe" );
    // Instantiate a ChromeDriver class.
    WebDriver driver=new ChromeDriver();

    // Launch Website
    driver.navigate().to("http://www.javatpoint.com/");

    //Maximize the browser
    driver.manage().window().maximize();

    //Scroll down the webpage by 5000 pixels
    JavascriptExecutor js = (JavascriptExecutor)driver;
    js.executeScript("scrollBy(0, 5000)");

    // Click on the search text box and send value
    driver.findElement(By.id("gsc-i-id1")).sendKeys("core Java");

    // Click on the search button
    driver.findElement(By.className("gsc-search-button")).click();
}

```

11. To Find the id of search input box ,visit the given url <http://www.javatpoint.com> and right click on search box

and select inspect element to find a html below like this:

```
<td id="gs_tti50" class="gsib_a">
  <input autocomplete="off" type="text" size="10" class="gsc-i-nput" name="search" title="search" id="gsc-i-id1" dir="ltr" spellcheck="false" style="width: 100%; padding: 0px; border: none; margin: 0px; height: auto; outline: none;"> == $0
```

12. Similarly Find the class / id of search button

```
▼ <td class="gsc-search-button">
  ▶ <button class="gsc-search-button gsc-search-button-v2">...</button>
  == $0
```

13. Run as Maven test and observe the output.

```
<terminated> C:\Program Files\Java\jdk-20\bin\javaw.exe (Sep 15, 2023, 2:02:47 PM) [pid: 10776]
[INFO] Results:
[INFO]
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 46.363 s
[INFO] Finished at: 2023-09-15T14:03:34+05:30
[INFO] -----
```

The screenshot shows a browser window with the URL javatpoint.com. The search query "core Java" has been entered into the search bar. The results page displays several links related to learning Core Java, including:

- How to learn core Java - Java for beginners** (<https://landing.codemy.cc/>)
- Core Java Course - Basic Java Course Online** (https://www.koenig-solutions.com/core_java/training)
- Core Java Made Easy | Udemy** (<https://www.udemy.com/>)
- Java Developer Software - Now** (<https://www.findresultsnow.com/localized/herc>)
- Java Tutorial | Learn Java Programming - javatpoint** (<http://www.javatpoint.com/java-tutorial>)
- 300 Core Java Interview Questions (2023) - javatpoint** (<http://www.javatpoint.com/corejava-interview-questions>)

b) To Setup and Run Selenium Tests in Jenkins Using Maven

1. Create a maven job as Maven Project
2. In Configure section - >go to **Source Code Management** -> Git - > paste this repo url

The screenshot shows two main sections: Jenkins configuration and a GitHub repository page.

Jenkins Configuration (Top):

- Git** tab selected.
- Repositories** section:
 - Repository URL:** https://github.com/prajwal0211/testing.git
 - Credentials:** - none -
 - Add:** Add Repository
 - Advanced** dropdown.
- Branches to build** section:
 - Branch Specifier (blank for 'any'):** */main

GitHub Repository Page (Bottom):

- Repository Name:** testing (Public)
- Branches:** main (selected), 1 branch, 0 tags
- Actions:** Go to file, Add file, Code
- Commits:**
 - prajwal0211 Add files via upload (3b80421, 18 minutes ago, 1 commit)
 - mavenproject Add files via upload (18 minutes ago)
- README:** Help people interested in this repository understand your project by adding a README. Add a README button.

Note: Ensure that the GitHub Plugin is installed at this point. If it has not been installed, then do install it from Manage Jenkins-> Manage Plugins

3. If your project is on local machine then go to Build section -> Root POM -> C:\Users\app\git\Eclipse_TestNG_Project\firstmaven\pom.xml Write the goals and options as => **test or clean test**. Make sure Root POM is set to pom.xml. -> save and apply

Build

Root POM ?
C:\Users\Avinash\eclipse-workspace\mavenproject\pom.xml

Goals and options ?
clean test

Advanced ▾

4. Go to build now -> console output and verify whether build was successful.

Console Output

```
Started by user Prajwal Waghmode
Running as SYSTEM
Building in workspace C:\ProgramData\Jenkins\.jenkins\workspace\sampletest
The recommended git tool is: NONE
No credentials specified
> git.exe rev-parse --resolve-git-dir C:\ProgramData\Jenkins\.jenkins\workspace\sampletest\.git # ti
Fetching changes from the remote Git repository
> git.exe config remote.origin.url https://github.com/prajwal0211/testing.git # timeout=10
Fetching upstream changes from https://github.com/prajwal0211/testing.git
> git.exe --version # timeout=10
```

Git Build Data

Revision: 3b80421cb22443ef591ee5903cced8c519d2dec1

Repository: <https://github.com/prajwal0211/testing.git>

- refs/remotes/origin/main

Built Branches

- refs/remotes/origin/main: Build #8 of Revision 3b80421cb22443ef591ee5903cced8c519d2dec1 (refs/remotes/origin/main)

Conclusion: - Hence we have successfully created and ran Selenium Tests in Jenkins Using Maven.

PRAJWAL WAGHMODE**121A3060****Experiment No: 8**

AIM: To understand Docker Architecture and Container Life Cycle, install Docker and execute docker commands to manage images and interact with containers.

THEORY: Docker Engine is an open source containerization technology for building and containerizing your applications. Docker Engine acts as a client-server application with:

- A server with a long-running daemon process `dockerd`.
- APIs which specify interfaces that programs can use to talk to and instruct the Docker daemon.
- A command line interface (CLI) client `docker`.

The CLI uses Docker APIs to control or interact with the Docker daemon through scripting or direct CLI commands. Many other Docker applications use the underlying API and CLI. The daemon creates and manages Docker objects, such as images, containers, networks, and volumes..

//Add the description of [Docker Architecture](#) & [Docker Container Lifecycle Management](#) here.

Installation of Docker:

To get started with Docker Engine on Ubuntu, make sure you meet the prerequisites, and then install Docker.

Prerequisites: OS requirements

To install Docker Engine, you need the 64-bit version of one of these Ubuntu versions:

- Ubuntu Hirsute 21.04
- Ubuntu Focal 20.04 (LTS)
- Ubuntu Bionic 18.04 (LTS)

Installation methods: You can install Docker Engine in different ways, depending on your needs:

1. Most users set up Docker's repositories and install from them
2. Some users download the DEB package and install it manually and manage upgrades completely manually.
3. In testing and development environments, some users choose to use automated convenience scripts to install Docker

Install using the convenience script: Docker provides a convenience script at get.docker.com to install Docker into development environments quickly and non-interactively. This example downloads the script from get.docker.com and runs it to install the latest stable release of Docker on Linux:

```
$ curl -fsSL https://get.docker.com -o get-docker.sh  
$ sudo sh get-docker.sh
```

To get OS detail and version

Department of IT, SIES GST

```
it77@it77-OptiPlex-3050 :~$ lsb_release -a
```

Uninstall old versions

```
it77@it77-OptiPlex-3050 :~$ sudo su
```

```
t77@it77-OptiPlex-305i0 :~$ sudo apt-get remove docker docker-engine docker.io containerd runc
```

```
it77@it77-OptiPlex-3050 :~$ sudo apt install curl
```

```
root@it77-OptiPlex-3050:/home/it77# curl -fsSL https://get.docker.com -o get-docker.sh
```

Examine scripts downloaded from the internet

```
root@it77-OptiPlex-3050:/home/it77# ls
```

```
root@it77-OptiPlex-3050:/home/it77# sudo sh get-docker.sh
```

Basic Docker Commands:

Check the version of Docker installed

```
root@it77-OptiPlex-3050:/home/it77# docker --version
```

Running existing Docker images: Go to Docker public repository at <https://hub.docker.com> to get the official images available for testing purpose

Run docker image

```
root@it77-OptiPlex-3050:/home/it77# docker run docker/whalesay cowsay hello_you
```

```
root@it77-OptiPlex-3050:/home/it77# docker run docker/whalesay cowsay hello_me
```

Check all pulled images

```
root@it77-OptiPlex-3050:/home/it77# docker images
```

Pull the sample images

```
root@it77-OptiPlex-3050:/home/it77# sudo docker pull postgres
```

```
root@it77-OptiPlex-3050:/home/it77# docker images
```

Check all running container

```
root@it77-OptiPlex-3050:/home/it77# docker ps // note the container id
```

```
root@it77-OptiPlex-3050:/home/it77# docker ps -a //previously ran containers
```

Pull the docker image of Ubuntu

```
root@it77-OptiPlex-3050:/home/it77# docker pull ubuntu:latest
```

```
root@it77-OptiPlex-3050:/home/it77# docker images
```

Run the command in a container: Getting a bash in Ubuntu

```
root@it77-OptiPlex-3050:/home/it77# docker run -it ubuntu:latest bash
```

```
root@it77-OptiPlex-3050:/home/it77# docker ps
```

```
root@67e9bd16d77b:/var/www/html# cat index.html
```

<html>

<title> First page</title>

```
<body bgcolor="pink">
```

Our home Page

</body>

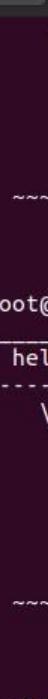
</html>

```
root@67e9bd16d77b:/var/www/html# service apache2 restart
```

```
root@67e9bd16d77b:/var/www/html# service apache2 status
```

* apache2 is running

Activities Terminal Oct 6 12:03



root@admin: /home/vboxuser/Desktop

```
\ \
      ##      .
      ## ## ##      ==
      ## ## ## ##      ===
      /*****      ___/ ===
~~~ {~~ ~~~~ ~~~ ~~~~ ~~ ~ /  ===== ~~~
      \____ o      \_/
      \_\_\_\_\_/_\_
root@admin:/home/vboxuser/Desktop# docker run docker/whalesay cowsay hello_you
< hello_you >
-----
\ \
      ##      .
      ## ## ##      ==
      ## ## ## ##      ===
      /*****      ___/ ===
~~~ {~~ ~~~~ ~~~ ~~~~ ~~ ~ /  ===== ~~~
      \____ o      \_/
      \_\_\_\_\_/_\_
root@admin:/home/vboxuser/Desktop# docker images
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
docker/whalesay      latest      6b362a9f73eb      8 years ago      247MB
root@admin:/home/vboxuser/Desktop# ss
```

```
Activities Terminal Oct 6 12:05
root@admin:/home/vboxuser/Desktop

vboxuser@admin:~/Desktop$ sudo su
[sudo] password for vboxuser:
vboxuser is not in the sudoers file. This incident will be reported.
vboxuser@admin:~/Desktop$ sudo apt-install docker
[sudo] password for vboxuser:
vboxuser is not in the sudoers file. This incident will be reported.
vboxuser@admin:~/Desktop$ sudo su
[sudo] password for vboxuser:
vboxuser is not in the sudoers file. This incident will be reported.
vboxuser@admin:~/Desktop$ %sudo su-
bash: fg: %sudo: no such job
vboxuser@admin:~/Desktop$ % sudo su--
bash: fg: %: no such job
vboxuser@admin:~/Desktop$ su
Password:
root@admin:/home/vboxuser/Desktop# apt install docker.io
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  bridge-utils containerd git git-man liberror-perl pigz runc ubuntu-fan
Suggested packages:
  ifupdown aufs-tools btrfs-progs cgroupfs-mount | cgroup-lite debootstrap
  docker-doc rinse zfs-fuse | zfsutils git-daemon-run | git-daemon-sysvinit
  git-doc git-el git-email git-gui gitk gitweb git-cvs git-mediawiki git-svn
The following NEW packages will be installed:
  bridge-utils containerd docker.io git git-man liberror-perl pigz runc
  ubuntu-fan
0 upgraded, 9 newly installed, 0 to remove and 469 not upgraded.
```

```
Activities Terminal Oct 6 12:05
root@admin:/home/vboxuser/Desktop

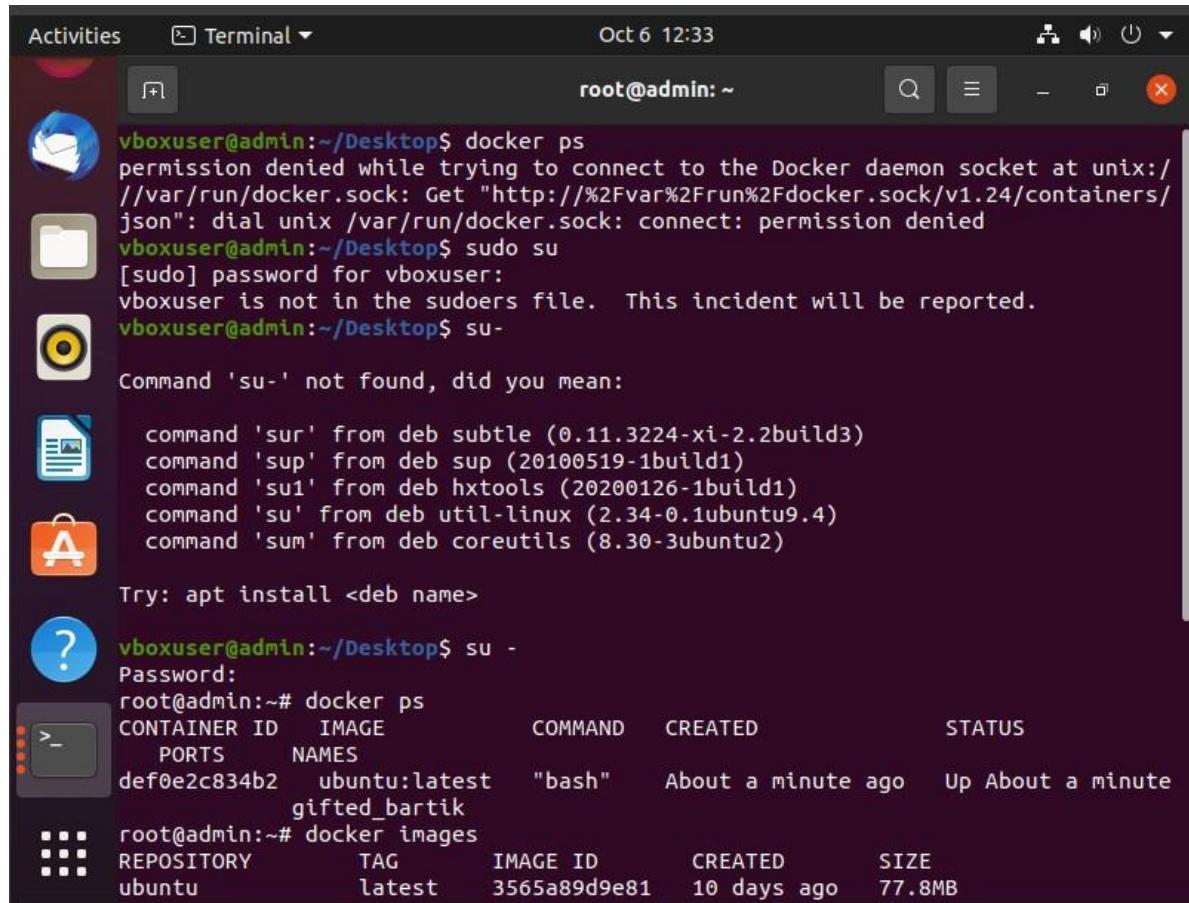
bash: fg: %: no such job
vboxuser@admin:~/Desktop$ su
Password:
root@admin:/home/vboxuser/Desktop# apt install docker.io
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  bridge-utils containerd git git-man liberror-perl pigz runc ubuntu-fan
Suggested packages:
  ifupdown aufs-tools btrfs-progs cgroupfs-mount | cgroup-lite debootstrap
  docker-doc rinse zfs-fuse | zfsutils git-daemon-run | git-daemon-sysvinit
  git-doc git-el git-email git-gui gitk gitweb git-cvs git-mediawiki git-svn
The following NEW packages will be installed:
  bridge-utils containerd docker.io git git-man liberror-perl pigz runc
  ubuntu-fan
0 upgraded, 9 newly installed, 0 to remove and 469 not upgraded.
Need to get 68.4 MB of archives.
After this operation, 304 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://gb.archive.ubuntu.com/ubuntu focal/universe amd64 pigz amd64 2.4-1
[57.4 kB]
Get:2 http://gb.archive.ubuntu.com/ubuntu focal/main amd64 bridge-utils amd64 1
.6-2ubuntu1 [30.5 kB]
Get:3 http://gb.archive.ubuntu.com/ubuntu focal-updates/main amd64 runc amd64 1
.1.7-0ubuntu1~20.04.1 [3,819 kB]
Get:4 http://gb.archive.ubuntu.com/ubuntu focal-updates/main amd64 containerd a
md64 1.7.2-0ubuntu1~20.04.1 [32.5 MB]
Get:5 http://gb.archive.ubuntu.com/ubuntu focal-updates/universe amd64 docker.i
```


vboxuser@admin:~/Desktop\$ docker ps
permission denied while trying to connect to the Docker daemon socket at unix:/
//var/run/docker.sock: Get "http://%2Fvar%2Frun%2Fdocker.sock/v1.24/containers/
json": dial unix /var/run/docker.sock: connect: permission denied
vboxuser@admin:~/Desktop\$ sudo su
[sudo] password for vboxuser:
vboxuser is not in the sudoers file. This incident will be reported.
vboxuser@admin:~/Desktop\$ su-
Command 'su-' not found, did you mean:
 command 'sur' from deb subtle (0.11.3224-xi-2.2build3)
 command 'sup' from deb sup (20100519-1build1)
 command 'su1' from deb hxtools (20200126-1build1)
 command 'su' from deb util-linux (2.34-0.1ubuntu9.4)
 command 'sum' from deb coreutils (8.30-3ubuntu2)
Try: apt install <deb name>
vboxuser@admin:~/Desktop\$ su -
Password:
root@admin:~# docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS
PORTS NAMES
def0e2c834b2 ubuntu:latest "bash" About a minute ago Up About a minute
root@admin:~# run docker

```
Activities Terminal Oct 6 12:19
root@def0e2c834b2: /var/www/html
invoke-rc.d: could not determine current runlevel
invoke-rc.d: policy-rc.d denied execution of start.
Processing triggers for libc-bin (2.35-0ubuntu3.3) ...
Processing triggers for ca-certificates (20230311ubuntu0.22.04.1) ...
Updating certificates in /etc/ssl/certs...
0 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d...
done.
root@def0e2c834b2:/# cd /var/www/html# mv index.html index.backup
bash: cd: too many arguments
root@def0e2c834b2:/# cd /var/www/html
root@def0e2c834b2:/var/www/html# /var/www/html# mv index.html index.backup
bash: /var/www/html#: No such file or directory
root@def0e2c834b2:/var/www/html# mv index.html index.backup
root@def0e2c834b2:/var/www/html# ls
index.backup
root@def0e2c834b2:/var/www/html# apt install nano
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Suggested packages:
  hunspell
The following NEW packages will be installed:
  nano
0 upgraded, 1 newly installed, 0 to remove and 2 not upgraded.
Need to get 280 kB of archives.
After this operation, 881 kB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu jammy/main amd64 nano amd64 6.2-1 [280 kB]
Fetched 280 kB in 2s (150 kB/s)
```

```
Activities Terminal Oct 6 12:20
root@def0e2c834b2: /var/www/html
hunspell
The following NEW packages will be installed:
  nano
0 upgraded, 1 newly installed, 0 to remove and 2 not upgraded.
Need to get 280 kB of archives.
After this operation, 881 kB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu jammy/main amd64 nano amd64 6.2-1 [280 kB]
Fetched 280 kB in 2s (158 kB/s)
debconf: delaying package configuration, since apt-utils is not installed
Selecting previously unselected package nano.
(Reading database ... 7860 files and directories currently installed.)
Preparing to unpack .../archives/nano_6.2-1_amd64.deb ...
Unpacking nano (6.2-1) ...
Setting up nano (6.2-1) ...
update-alternatives: using /bin/nano to provide /usr/bin/editor (editor) in auto mode
update-alternatives: warning: skip creation of /usr/share/man/man1/editor.1.gz because associated file /usr/share/man/man1/nano.1.gz (of link group editor) doesn't exist
update-alternatives: using /bin/nano to provide /usr/bin/pico (pico) in auto mode
update-alternatives: warning: skip creation of /usr/share/man/man1/pico.1.gz because associated file /usr/share/man/man1/nano.1.gz (of link group pico) doesn't exist
root@def0e2c834b2:/var/www/html# nano index.html
root@def0e2c834b2:/var/www/html# service apache2 restart
 * Restarting Apache httpd web server apache2
AH00558: apache2: Could not reliably determine the server's fully qualified dom
```

```
root@def0e2c834b2:/var/www/html# service apache2 status
 * apache2 is running
root@def0e2c834b2:/var/www/html#
```



A screenshot of an Ubuntu desktop environment. A terminal window is open in the top dock, showing the command `service apache2 status` and its output. Below the terminal, the desktop environment is visible with various icons for applications like Dash, Home, File Manager, and others.

```
vboxuser@admin:~/Desktop$ docker ps
permission denied while trying to connect to the Docker daemon socket at unix:/
//var/run/docker.sock: Get "http://%2Fvar%2Frun%2Fdocker.sock/v1.24/containers/
json": dial unix /var/run/docker.sock: connect: permission denied
vboxuser@admin:~/Desktop$ sudo su
[sudo] password for vboxuser:
vboxuser is not in the sudoers file. This incident will be reported.
vboxuser@admin:~/Desktop$ su-
Command 'su-' not found, did you mean:

Try: apt install <deb name>
vboxuser@admin:~/Desktop$ su -
Password:
root@admin:~# docker ps
CONTAINER ID   IMAGE      COMMAND   CREATED          STATUS
           PORTS     NAMES
def0e2c834b2   ubuntu:latest "bash"   About a minute ago   Up About a minute
               gifted_bartik
root@admin:~# docker images
REPOSITORY    TAG      IMAGE ID      CREATED          SIZE
ubuntu        latest   3565a89d9e81  10 days ago    77.8MB
```

Activities Terminal Oct 6 12:09

```
root@admin:/home/vboxuser/Desktop# docker images
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
docker/whalesay  latest   6b362a9f73eb  8 years ago  247MB
root@admin:/home/vboxuser/Desktop# docker pull ubuntu:latest
latest: Pulling from library/ubuntu
37aaaf24cf781: Pull complete
Digest: sha256:9b8dec3bf938bc80fbe758d856e96fdfab5f56c39d44b0cff351e847bb1b01ea
Status: Downloaded newer image for ubuntu:latest
docker.io/library/ubuntu:latest
root@admin:/home/vboxuser/Desktop# docker images\
> ^C
root@admin:/home/vboxuser/Desktop# docker images
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
ubuntu          latest   3565a89d9e81  10 days ago  77.8MB
docker/whalesay  latest   6b362a9f73eb  8 years ago  247MB
root@admin:/home/vboxuser/Desktop# docker run -it ubuntu:latest bash
root@def0e2c834b2:# docker ps
bash: docker: command not found
root@def0e2c834b2:# docker run -it
bash: docker: command not found
root@def0e2c834b2:#
```

Activities Terminal Oct 6 12:33

```
vboxuser@admin:~/Desktop$ su -
Password:
root@admin:~# docker ps
CONTAINER ID   IMAGE           COMMAND   CREATED      STATUS
              PORTS NAMES
def0e2c834b2   ubuntu:latest   "bash"    About a minute ago   Up About a minute
                  gifted_bartik
root@admin:~# docker images
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
ubuntu          latest   3565a89d9e81  10 days ago  77.8MB
docker/whalesay  latest   6b362a9f73eb  8 years ago  247MB
root@admin:~# docker ps
CONTAINER ID   IMAGE           COMMAND   CREATED      STATUS      PORTS
              NAMES
def0e2c834b2   ubuntu:latest   "bash"    23 minutes ago   Up 23 minutes
                  gifted_bartik
root@admin:~# docker commit 3565a89d9e81 shreeshk9/somu:s1
Error response from daemon: No such container: 3565a89d9e81
root@admin:~# docker commit def0e2c834b2 shreeshk9/somu:s1
sha256:b7897b758f18fde8557b7f9b2e880d2f4eeff19967c47455b5a12a0f125959e
root@admin:~# docker images
REPOSITORY      TAG      IMAGE ID      CREATED      SIZE
shreeshk9/somu  s1      b7897b758f18  33 seconds ago  231MB
ubuntu          latest   3565a89d9e81  10 days ago  77.8MB
docker/whalesay  latest   6b362a9f73eb  8 years ago  247MB
root@admin:~#
```

```
Activities Terminal Oct 6 12:41
root@admin: ~
vboxuser@admin:~/Desktop$ docker ps
permission denied while trying to connect to the Docker daemon socket at unix:/
//var/run/docker.sock: Get "http://%2Fvar%2Frun%2Fdocker.sock/v1.24/containers/
json": dial unix /var/run/docker.sock: connect: permission denied
vboxuser@admin:~/Desktop$ su -
Password:
root@admin:~# docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS
 NAMES
def0e2c834b2 ubuntu:latest "bash" 30 minutes ago Up 30 minutes
 gifted_bartik
root@admin:~# docker run -itd -p 8888:80 --name s1 shreeshk9/somu
Unable to find image 'shreeshk9/somu:latest' locally
docker: Error response from daemon: pull access denied for shreeshk9/somu, repo-
sitory does not exist or may require 'docker login': denied: requested access t-
o the resource is denied.
See 'docker run --help'.
root@admin:~# docker run -itd -p 8888:80 --name s1 shreeshk9/somu
Unable to find image 'shreeshk9/somu:latest' locally
docker: Error response from daemon: pull access denied for shreeshk9/somu, repo-
sitory does not exist or may require 'docker login': denied: requested access t-
o the resource is denied.
See 'docker run --help'.
root@admin:~# docker run -itd -p 8888:80 --name s1 shreeshk9/somu:s1
e686d9a165f79617c011037771e4032c3faa1ffdf46d9db04ce09e57fca2c456
root@admin:~# ufw allow 8888
Rules updated
Rules updated (v6)
root@admin:~#
```

```
Activities Terminal Oct 6 12:50
root@admin: ~
8081c41040f22d3af185569 is not running
root@admin:~# docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS
 NAMES
e686d9a165f7 shreeshk9/somu:s1 "bash" 5 minutes ago Up 5 minutes 0.0
.0.0:8888->80/tcp, :::8888->80/tcp s1
root@admin:~# docker exec -it e686d9a165f7 service apache2 restart
 * Restarting Apache httpd web server apache2
AH00558: apache2: Could not reliably determine the server's fully qualified dom-
ain name, using 172.17.0.2. Set the 'ServerName' directive globally to suppress
this message
[ OK ]
root@admin:~# ifconfig
Command 'ifconfig' not found, but can be installed with:
apt install net-tools
root@admin:~# apt install net-tools
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
 net-tools
0 upgraded, 1 newly installed, 0 to remove and 468 not upgraded.
Need to get 196 kB of archives.
After this operation, 864 kB of additional disk space will be used.
Get:1 http://gb.archive.ubuntu.com/ubuntu focal/main amd64 net-tools amd64 1.60
+git20180626.aebd88e-1 [196 kB]
```

Activities Terminal Oct 6 12:50

```
root@admin:~# docker exec -it e686d9a165f7 service apache2 restart
 * Restarting Apache httpd web server apache2
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 172.17.0.2. Set the 'ServerName' directive globally to suppress this message
[ OK ]
root@admin:~# ifconfig
Command 'ifconfig' not found, but can be installed with:
apt install net-tools
root@admin:~# apt install net-tools
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  net-tools
0 upgraded, 1 newly installed, 0 to remove and 468 not upgraded.
Need to get 196 kB of archives.
After this operation, 864 kB of additional disk space will be used.
Get:1 http://gb.archive.ubuntu.com/ubuntu focal/main amd64 net-tools amd64 1.60+git20180626.aebd88e-1ubuntu1 [196 kB]
Fetched 196 kB in 32s (6,183 B/s)
Selecting previously unselected package net-tools.
(Reading database ... 187930 files and directories currently installed.)
Preparing to unpack .../net-tools_1.60+git20180626.aebd88e-1ubuntu1_amd64.deb ...
Unpacking net-tools (1.60+git20180626.aebd88e-1ubuntu1) ...
```



The push refers to repository [docker.io/bushsk/ubuntu_1] 35279adae3eb:

Pushed e0b3afb09dc3: Mounted from library/ubuntu6c01b5a53aac:

Mounted from library/ubuntu2c6ac8e5063e:

Mounted from library/ubuntucc967c529ced:

Mounted from library/Ubuntu

Conclusion :-

Docker is an open platform for developing, shipping, and running applications. Docker enables you to separate your applications from your infrastructure so you can deliver software quickly. With Docker, you can manage your infrastructure in the same ways you manage your applications.

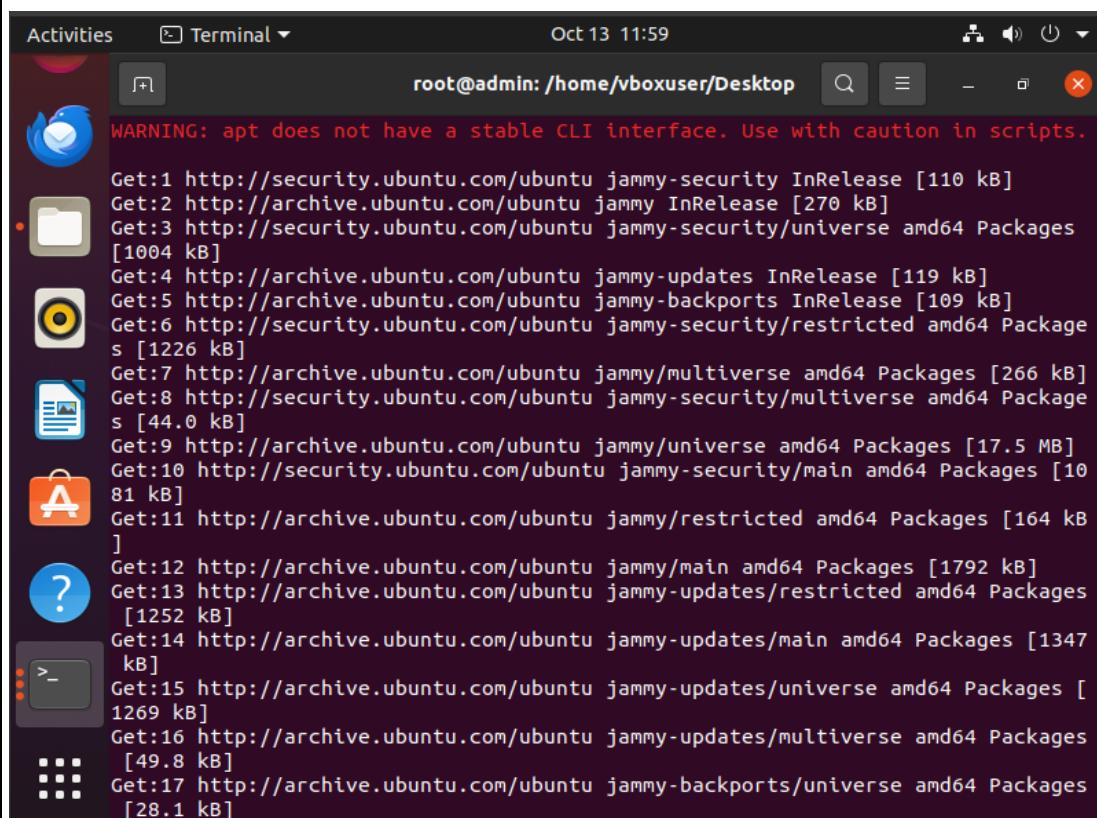
Experiment No: 9

AIM: To learn Dockerfile instructions, build an image for a sample web application using Dockerfile.

THEORY:

Docker also gives you the capability to create your own Docker images, and it can be done with the help of Docker Files. A Docker File is a simple text file with instructions on how to build your images.

```
vboxuser@anjali:~/Desktop$ sudo su
root@anjali:/home/vboxuser/Desktop# nano Dockerfile
root@anjali:/home/vboxuser/Desktop# docker build -t siesnginxservers .
[+] Building 1.2s (8/8) FINISHED
  => [internal] load build definition from Dockerfile
  => => transferring dockerfile: 207B
  => [internal] load .dockerignore
  => => transferring context: 2B
  => [internal] load metadata for docker.io/library/ubuntu:latest
  => [1/4] FROM docker.io/library/ubuntu:latest
  => [internal] load build context
  => => transferring context: 2B
  => CANCELED [2/4] RUN apt update -y
  => CACHED [3/4] RUN apt install nginx -y
  => ERROR [4/4] COPY index.html /var/www/html/index.html
  ----
  > [4/4] COPY index.html /var/www/html/index.html:
  ----
Dockerfile:6
-----
  4 |     RUN apt install nginx -y
  5 |     EXPOSE 80
  6 | >>> COPY index.html /var/www/html/index.html
  7 |     CMD ["nginx", "-g", "daemon off;"]
  8 |
-----
```



```
Activities Terminal ▾ Oct 13 11:59
root@admin: /home/vboxuser/Desktop
WARNING: apt does not have a stable CLI interface. Use with caution in scripts.

Get:1 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:2 http://archive.ubuntu.com/ubuntu jammy InRelease [270 kB]
Get:3 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [1004 kB]
Get:4 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:5 http://archive.ubuntu.com/ubuntu jammy-backports InRelease [109 kB]
Get:6 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [1226 kB]
Get:7 http://archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [266 kB]
Get:8 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 Packages [44.0 kB]
Get:9 http://archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [17.5 MB]
Get:10 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [1081 kB]
Get:11 http://archive.ubuntu.com/ubuntu jammy/restricted amd64 Packages [164 kB]
Get:12 http://archive.ubuntu.com/ubuntu jammy/main amd64 Packages [1792 kB]
Get:13 http://archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [1252 kB]
Get:14 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1347 kB]
Get:15 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1269 kB]
Get:16 http://archive.ubuntu.com/ubuntu jammy-updates/multiverse amd64 Packages [49.8 kB]
Get:17 http://archive.ubuntu.com/ubuntu jammy-backports/universe amd64 Packages [28.1 kB]
```

```
Activities Terminal ▾ Oct 13 11:59
root@admin:/home/vboxuser/Desktop

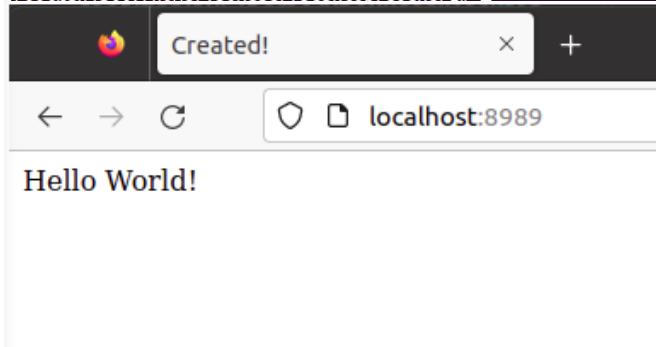
Setting up libgd3:amd64 (2.3.0-2ubuntu2) ...
Setting up libxslt1.1:amd64 (1.1.34-4ubuntu0.22.04.1) ...
Setting up libnginx-mod-http-image-filter (1.18.0-6ubuntu14.4) ...
Setting up libnginx-mod-http-xslt-filter (1.18.0-6ubuntu14.4) ...
Setting up nginx-core (1.18.0-6ubuntu14.4) ...
invoke-rc.d: could not determine current runlevel
invoke-rc.d: policy-rc.d denied execution of start.
Setting up nginx (1.18.0-6ubuntu14.4) ...
Processing triggers for libc-bin (2.35-0ubuntu3.3) ...
Removing intermediate container e51daf24f3f4
--> c55422db778f
Step 5/7 : EXPOSE 80
--> Running in 674a6dc6ab17
Removing intermediate container 674a6dc6ab17
--> 4f05bcdcfda6
Step 6/7 : COPY index.html /var/www/html/index.html
--> 351bbda48dbd
Step 7/7 : CMD ["nginx", "-g", "daemon off;"]
--> Running in 8defa1b9724f
Removing intermediate container 8defa1b9724f
--> baba0b5e15de
Successfully built baba0b5e15de
Successfully tagged siesnginxservers:latest
root@admin:/home/vboxuser/Desktop# docker run -itd -p 8989:80 siesnginxservers
ac3271e3744f0be2674b547fee8b1fbb6ed56161a77ad867f3c802e9c3e68e4f
root@admin:/home/vboxuser/Desktop# ufw allow 8989
Rules updated
Rules updated (v6)
root@admin:/home/vboxuser/Desktop#
```

```
root@anjali:/home/vboxuser/Desktop# mkdir docker1
root@anjali:/home/vboxuser/Desktop# cd docker1
root@anjali:/home/vboxuser/Desktop/docker1# nano index.html
root@anjali:/home/vboxuser/Desktop/docker1# ls
index.html
```

```
GNU nano 4.8
<html>
<title> Created! </title>
<body>
Hello World!
</body>
</html>
```

```
GNU nano 4.8
FROM ubuntu:latest
MAINTAINER "PRAJWAL"
RUN apt update -y
RUN apt install nginx -y
EXPOSE 80
COPY index.html /var/www/html/index.html
CMD ["nginx", "-g", "daemon off;"]
```

```
root@anjali:/home/vboxuser/Desktop/docker1# nano Dockerfile
root@anjali:/home/vboxuser/Desktop/docker1# ls
Dockerfile index.html
root@anjali:/home/vboxuser/Desktop/docker1# docker build -t siesnginxservers .
[+] Building 106.7s (9/9) FINISHED                                            docker:default
=> [internal] load .dockerignore                                              0.0s
=> => transferring context: 2B                                               0.0s
=> [internal] load build definition from Dockerfile                         0.1s
=> => transferring dockerfile: 207B                                           0.0s
=> [internal] load metadata for docker.io/library/ubuntu:latest              0.0s
=> CACHED [1/4] FROM docker.io/library/ubuntu:latest                           0.0s
=> [internal] load build context                                              0.1s
=> => transferring context: 106B                                             0.0s
=> [2/4] RUN apt update -y                                                 62.2s
=> [3/4] RUN apt install nginx -y                                         43.5s
=> [4/4] COPY index.html /var/www/html/index.html                            0.1s
=> exporting to image                                                       0.7s
=> => exporting layers                                                       0.6s
=> => writing image sha256:d75d79907394606bc0208b2bfb4e0f6a4900e654d4311a861d27dcf41 0.0s
=> => naming to docker.io/library/siesnginxservers                          0.0s
root@anjali:/home/vboxuser/Desktop/docker1# docker run -itd -p 8989:80 siesnginxservers
14e1ec7d0edce5f7aef4c7da1957e406fdf791a7806012421181c3f71c45d20c
root@anjali:/home/vboxuser/Desktop/docker1# ufw allow 8989
Rules updated
Rules updated (v6)
root@anjali:/home/vboxuser/Desktop/docker1#
```



Conclusion:

Using **Dockerfile** is a simpler and faster way of building Docker image. It automates the process by going through the script with all the commands for assembling an image.

When building a Docker image, you also want to make sure to keep Docker image size light. Avoiding large images speeds-up building and deploying containers. Therefore, it is crucial to reduce the image size to a minimum.