

## WEEK 8

**Naren Karthick A**

**St. Joseph's Institute of Technology**

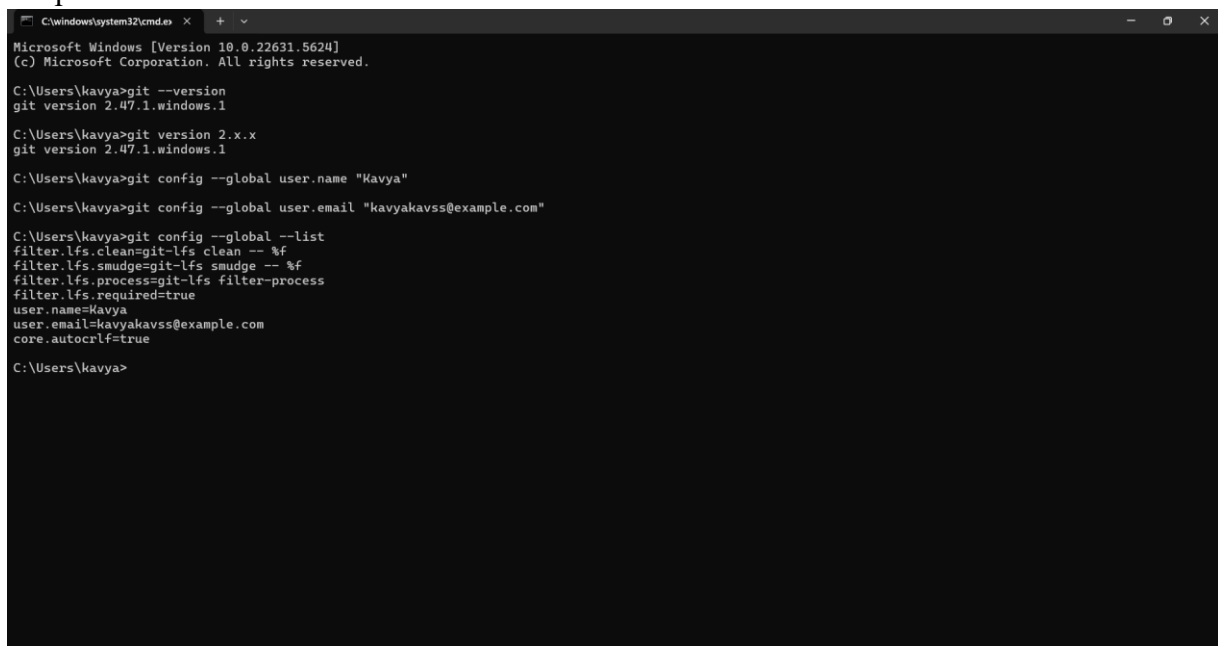
**Superset ID: 6377326**

### Exercise 1:

```
git --version

git version 2.x.x
git config --global user.name "Naren"
git config --global user.email Naren@example.com
git config --global --list
git config --global core.editor "code --wait"
git config --global -e
mkdir GitDemo
cd GitDemo
git init
ls -a
echo "Welcome to Git Demo" > welcome.txt
cat welcome.txt
git status
git add welcome.txt
git commit
Added welcome.txt
git status
git remote add origin https://github.com/Naren/GitDemo.git
```

Output:



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.22631.5624]
(c) Microsoft Corporation. All rights reserved.

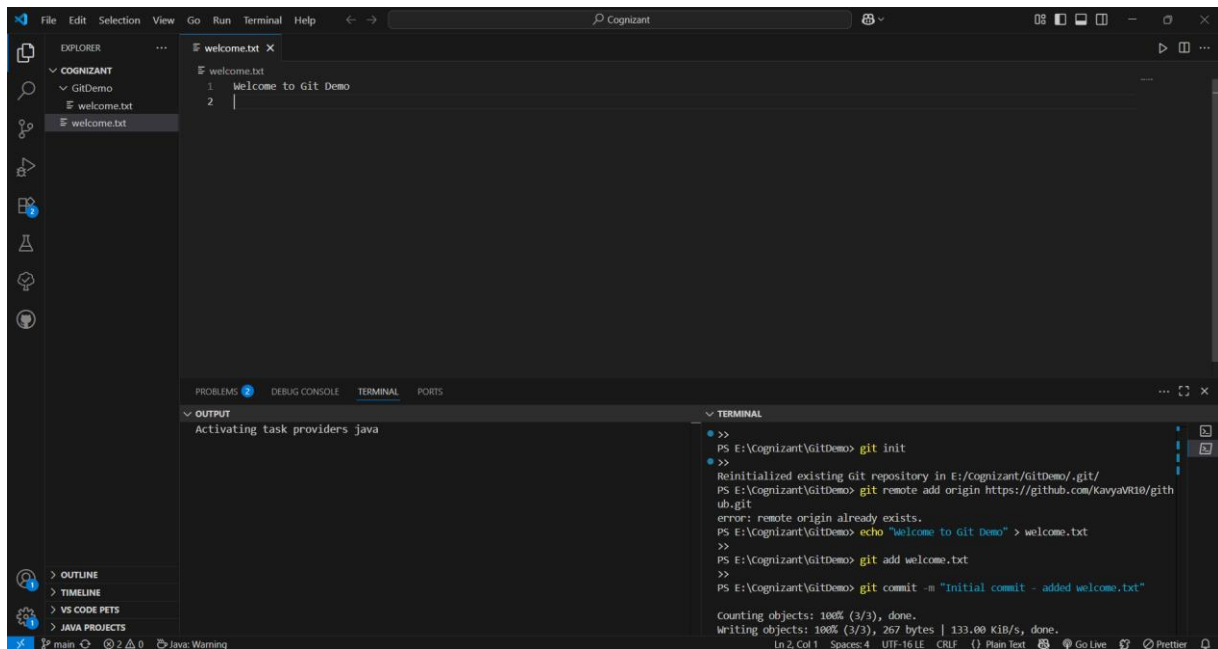
C:\Users\kavya>git --version
git version 2.47.1.windows.1

C:\Users\kavya>git version 2.x.x
git version 2.47.1.windows.1

C:\Users\kavya>git config --global user.name "Kavya"
C:\Users\kavya>git config --global user.email "kavyakavss@example.com"

C:\Users\kavya>git config --global --list
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
user.name=Kavya
user.email=kavyakavss@example.com
core.autocrlf=true

C:\Users\kavya>
```



```
File Edit Selection View Go Run Terminal Help
Cognizant

EXPLORER
COGNIZANT
  GitDemo
    welcome.txt

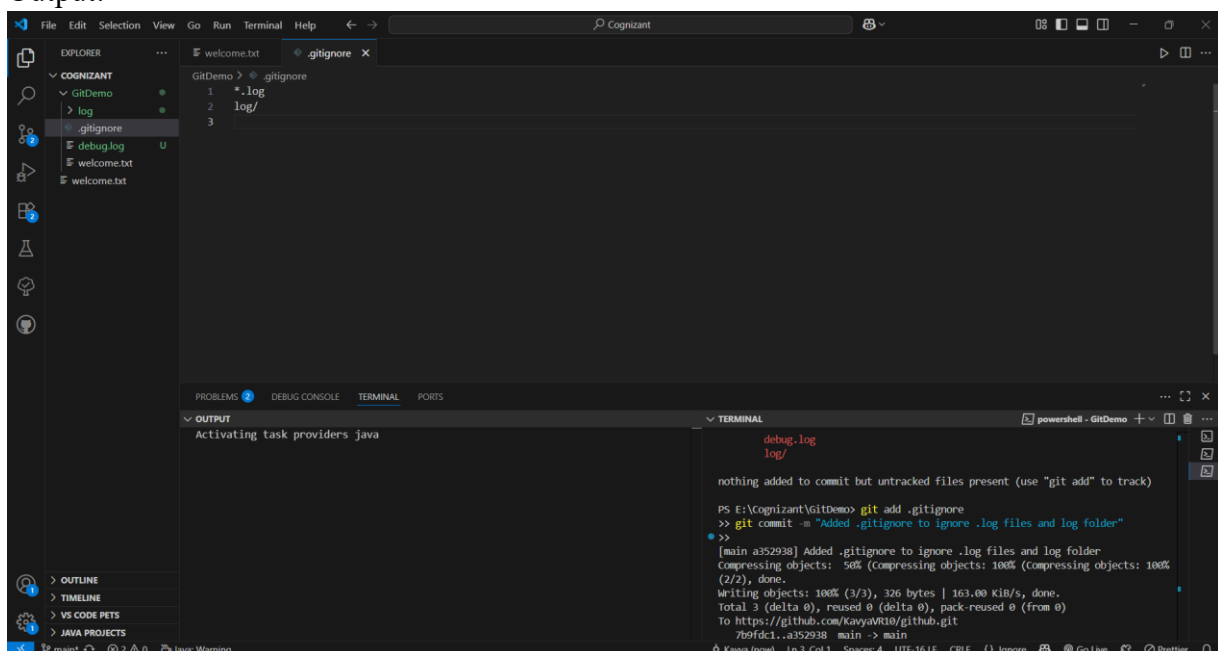
welcome.txt
1 welcome.txt
2

PROBLEMS DEBUG CONSOLE TERMINAL PORTS
OUTPUT
Activating task providers java
TERMINAL
PS E:\Cognizant\GitDemo> git init
PS E:\Cognizant\GitDemo> git remote add origin https://github.com/KavyaVR10/github.git
error: remote origin already exists.
PS E:\Cognizant\GitDemo> echo "Welcome to Git Demo" > welcome.txt
PS E:\Cognizant\GitDemo> git add welcome.txt
PS E:\Cognizant\GitDemo> git commit -m "Initial commit - added welcome.txt"
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 267 bytes | 133.00 KiB/s, done.
ln 2, Col 1 Spaces: 4 UTF-16 LE CRLF {} Plain Text Go Live Prettier
```

## Exercise 2:

```
echo "This is a log file" > debug.log
mkdir log
echo "Log data" > log/output.txt
echo "*.log" > .gitignore
echo "log/" >> .gitignore
git status
git add .gitignore
git commit -m "Added .gitignore to ignore .log files and log folder"
git push
```

Output:



```
File Edit Selection View Go Run Terminal Help
Cognizant

EXPLORER
COGNIZANT
  GitDemo
    log
    .gitignore
    debug.log
    welcome.txt
    welcome.txt

.gitignore
1 *.log
2 log/
3

PROBLEMS DEBUG CONSOLE TERMINAL PORTS
OUTPUT
Activating task providers java
TERMINAL
debug.log
log/

nothing added to commit but untracked files present (use "git add" to track)
PS E:\Cognizant\GitDemo> git add .gitignore
>> git commit -m "Added .gitignore to ignore .log files and log folder"
[main a352938] Added .gitignore to ignore .log files and log folder
Compressing objects: 50% (compressing objects: 100% (compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 326 bytes | 163.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/KavyaVR10/github.git
7b9fdc1..a352938 main -> main
```

### Exercise 3: Branching

git branch GitNewBranch

git branch

git branch -r

git checkout GitNewBranch

echo "This is a new file" > newfile.txt

git add newfile.txt

git commit -m "Added newfile.txt in GitNewBranch"

git status

Merging:

git checkout master

git diff master GitNewBranch

git difftool master GitNewBranch

git merge GitNewBranch

git log --oneline --graph --decorate

git branch -d GitNewBranch

git status

Output:

```
PS E:\Cognizant> cd GitDemo
PS E:\Cognizant\GitDemo> git branch GitNewBranch
PS E:\Cognizant\GitDemo> git branch
  GitNewBranch
* main
PS E:\Cognizant\GitDemo> git branch -r
  origin/main
PS E:\Cognizant\GitDemo> git checkout GitNewBranch
Switched to branch 'GitNewBranch'
PS E:\Cognizant\GitDemo> echo "This is a new file" > newfile.txt
● PS E:\Cognizant\GitDemo> git add newfile.txt
.txt
● PS E:\Cognizant\GitDemo> git status
On branch GitNewBranch
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    debug.log
    log/

nothing added to commit but untracked files present (use "git add" to track)
PS E:\Cognizant\GitDemo>
```

```
newfile.txt
new file mode 100644
index 0000000..ba13b4c
Binary files /dev/null and b/newfile.txt differ
PS E:\Cognizant\GitDemo> git difftool main GitNewBranch

This message is displayed because 'diff.tool' is not configured.
See 'git difftool --tool-help' or 'git help config' for more details.
'git difftool' will now attempt to use one of the following tools:
kompare emerge vimdiff nvimdiff

Viewing (1/1): 'newfile.txt'
Launch 'vimdiff' [Y/n]? n
PS E:\Cognizant\GitDemo> git merge GitNewBranch
Updating a352938..3674635
Fast-forward
 newfile.txt | Bin 0 -> 42 bytes
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 newfile.txt
PS E:\Cognizant\GitDemo> git branch -d GitNewBranch
Deleted branch GitNewBranch (was 3674635).
PS E:\Cognizant\GitDemo>
```

#### Exercise 4:

```
git checkout main
git status
git branch GitWork
git checkout GitWork
echo "<message>Hello from GitWork</message>" > hello.xml
git status
git add hello.xml
git commit -m "Added hello.xml with updated content in GitWork"
git checkout main
echo "<message>Hello from Main branch</message>" > hello.xml
git add hello.xml
git commit -m "Added hello.xml with different content in main"
git log --oneline --graph --decorate --all
git diff main GitWork
git config --global diff.tool p4merge
git config --global merge.tool p4merge
git config --global mergetool.p4merge.path "C:/Program Files/Perforce/p4merge.exe"
git difftool main GitWork
git checkout main
git merge GitWork
git mergetool
git add hello.xml
git commit -m "Merged GitWork into main with conflict resolved"
echo "*.orig" >> .gitignore
```

```
git status
git add .gitignore
git commit -m "Added .gitignore for backup files"
git branch
git branch -d GitWork
git log --oneline --graph --decorate
```





Output:

```
PS E:\Cognizant> cd GitDemo
PS E:\Cognizant\GitDemo> git checkout main
>> git status
● >>
Already on 'main'
Your branch is ahead of 'origin/main' by 1 commit.
  (use "git push" to publish your local commits)
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
● PS E:\Cognizant\GitDemo> git branch GitWork
>> git checkout GitWork
>> echo "<message>Hello from GitWork</message>" > hello.xml
>> git status
>>
Switched to branch 'GitWork'
On branch GitWork
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    hello.xml

nothing added to commit but untracked files present (use "git add" to track)
PS E:\Cognizant\GitDemo> echo "<update>Updated content in GitWork branch</update>" >> hello.xml
● >> git status
>>
On branch GitWork
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    hello.xml

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

Ln 1, Col 1 Spaces: 4 UTF-16 LE CRLF {} XML    Prettier 

```
Your branch is ahead of 'origin/main' by 1 commit.
(use "git push" to publish your local commits)
PS E:\Cognizant\GitDemo> echo "<message>Hello from Main branch</message>" > hello.xml
>>
PS E:\Cognizant\GitDemo> git add hello.xml
>> git commit -m "Added hello.xml with different content in main"
>>
[main bc782d7] Added hello.xml with different content in main
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 hello.xml
PS E:\Cognizant\GitDemo> git log --oneline --graph --decorate --all
>>
* bc782d7 (HEAD -> main) Added hello.xml with different content in main
| * 7c2b0ec (GitWork) Added hello.xml with updated content in GitWork
|/
* 3674635 Added newfile.txt in GitNewBranch
* a352938 (origin/main) Added .gitignore to ignore .log files and log folder

* 7b9fdc1 Initial commit - added welcome.txt
PS E:\Cognizant\GitDemo> git diff main GitWork
>>
diff --git a/hello.xml b/hello.xml
index 8abd384..74d5180 100644
Binary files a/hello.xml and b/hello.xml differ
PS E:\Cognizant\GitDemo> git config --global diff.tool p4merge
>> git config --global merge.tool p4merge
>> git config --global mergetool.p4merge.path "C:/Program Files/Perforce/p4merge.exe"
>> git difftool main GitWork
Already on 'main'
Your branch is ahead of 'origin/main' by 2 commits.
(use "git push" to publish your local commits)
warning: Cannot merge binary files: hello.xml (HEAD vs. GitWork)
Auto-merging hello.xml
```

```

● >> E:\Cognizant\GitDemo>
[main 661524e] Merged GitWork into main with conflict resolved
PS E:\Cognizant\GitDemo> echo "*.orig" >> .gitignore
● >> git status
>>
On branch main
t branch
>>
    GitWork
* main
PS E:\Cognizant\GitDemo> git branch -d GitWork
>>
Deleted branch GitWork (was 7c2b0ec).
PS E:\Cognizant\GitDemo> git log --oneline --graph --decorate
>>
* 0a896f5 (HEAD -> main) Added .gitignore for backup files
*    661524e Merged GitWork into main with conflict resolved
| \
| * 7c2b0ec Added hello.xml with updated content in GitWork
* | bc782d7 Added hello.xml with different content in main
| /
* 3674635 Added newfile.txt in GitNewBranch
* a352938 (origin/main) Added .gitignore to ignore .log files and log folder
* 7b9fdc1 Initial commit - added welcome.txt
PS E:\Cognizant\GitDemo>

```

Exercise 5:

git branch -a

git fetch origin

git checkout -b Git-T03-HOL\_002 origin/Git-T03-HOL\_002

git push origin main

git checkout -b Git-T03-HOL\_002

git push origin Git-T03-HOL\_002

Output:

```
>> git checkout Git-T03-HOL_002
● >>
>> # Check if there are commits not pushed yet
>> git status
>> git log origin/Git-T03-HOL_002..HEAD --oneline
>>
>> # Push to remote
>> git push origin Git-T03-HOL_002
>>
Already on 'Git-T03-HOL_002'
On branch Git-T03-HOL_002
nothing to commit, working tree clean
Everything up-to-date
○ PS E:\Cognizant\GitDemo> 
```