# DN 4.0 JAVA FSE SOLUTIONS – WEEK 8 Skill: GIT

#### Hands-On 2

# **Objective**

This document outlines the steps taken to implement a .gitignore file to ignore unwanted files and folders from being tracked by a Git repository.

# **Step 1: Create Ignored Files and Folders**

First, I created the file and folder that the lab exercise requires to be ignored by Git.

- Create a log file and folder: I used the following commands to create a
  file with a .log extension and a folder named log in my Git working
  directory.
  - Commands:
  - cd GitDemo
  - touch example.log
  - mkdir log
  - echo "This is a log file." > log/app.log

#### **Output:**

```
manas@Manasa MINGW64 ~
$ cd GitDemo

manas@Manasa MINGW64 ~/GitDemo (master)
$ touch example.log
mkdir log

manas@Manasa MINGW64 ~/GitDemo (master)
$ echo "This is a log file." > log/app.log
```

- Check initial status: I executed git status to verify that both the file and folder were untracked by Git.
  - o Command:
  - git status

# **Output:**

# Step 2: Implement the .gitignore File

Next, I created and updated the .gitignore file to tell Git to ignore these items.

- Create and edit .gitignore: I opened and edited a new .gitignore file using Notepad++.
  - o Command:
  - o notepad++ .gitignore

### **Output:**

```
manas@Manasa MINGW64 ~/GitDemo (master)
$ notepad++ .gitignore
```

- Add ignore rules: I added the following patterns to the .gitignore file to ignore all files ending with .log and the entire log directory.
  - Content of .gitignore:
  - # Ignore all files with a .log extension
  - \*.log

0

- # Ignore the entire 'log' directory
- o log/

#### **Output:**

```
# Ignore all files with a .log extension
*.log
# Ignore the entire 'log' directory
log/
```

# **Step 3: Verify and Commit Changes**

After saving the .gitignore file, I verified that Git was ignoring the specified files and then committed the changes to the repository.

- Verify new status: I ran git status again. The example.log and log/ folder were no longer listed, confirming that the .gitignore file was working correctly. Only the .gitignore file itself was untracked.
  - command:
  - git status

## **Output:**

- **Commit the .gitignore file:** I added the .gitignore file to the staging area and committed it with a descriptive message.
  - Commands:
  - o git add .gitignore
  - o git commit -m "Add .gitignore to ignore log files and folder"

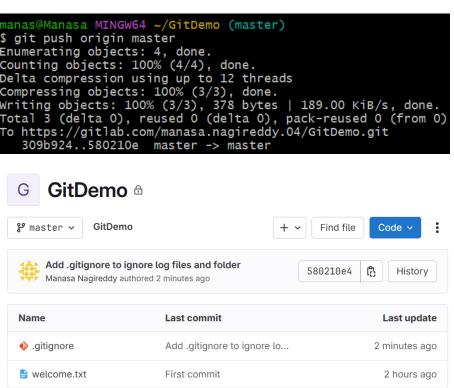
#### **Output:**

```
manas@Manasa MINGW64 ~/GitDemo (master)
$ git add .gitignore

manas@Manasa MINGW64 ~/GitDemo (master)
$ git commit -m "Add .gitignore to ignore log files and folder"
[master 580210e] Add .gitignore to ignore log files and folder
1 file changed, 5 insertions(+)
create mode 100644 .gitignore
```

- Push to remote repository: Finally, I pushed the changes to the remote repository on GitLab.
  - Command:
  - git push origin master

# Output:



The .gitignore file has now been successfully added to the repository, ensuring that all log files and folders will be ignored by Git from this point forward.