Week 8 - GIT

**Task 1: Setup GitLab and Git on Your Local System**

**Step 1: Setup GitLab Repository**

1. Go to GitLab and **Sign Up**.

**Step 2: Verify Git Installation**

git --version

**Step 3: Configure Git with User Info**

git config --global user.name "Shaarmi"

git config --global user.email "shaarmi@example.com"

**Step 4: Verify Configuration**

git config --list

**Task 2:**

**1. Create a .log file and log/ folder**

echo "This is a test log" > test.log

mkdir log

echo "inside log folder" > log/info.txt

**2. Create or update .gitignore**

echo "\*.log" >> .gitignore

echo "log/" >> .gitignore

**3. Add and commit .gitignore**

git add .gitignore

git commit -m "Ignore .log files and log folder"

**4. Check status**

git status

**Task 3: Branching and Merging in Git Bash**

**Branching**

Create new branch:

git branch GitNewBranch

List branches:

git branch

Switch to new branch:

git checkout GitNewBranch

Add files and commit:

echo "Hello from new branch" > branchfile.txt

git add branchfile.txt

git commit -m "Added file to GitNewBranch"

Check status:

git status

**Merging**

Switch back to master:

git checkout master

Show CLI diff:

git diff GitNewBranch

Use **P4Merge** for visual diff (after setting it as merge tool):

git mergetool

Merge the branch:

git merge GitNewBranch

View log after merge:

git log --oneline --graph --decorate

Delete merged branch:

git branch -d GitNewBranch

**Task 4: Merge Conflict & .gitignore Handling**

Verify clean state:

git status

Create new branch:

git checkout -b GitWork

Add file:

echo "Hello from GitWork" > hello.xml

git add hello.xml

Commit:

git commit -m "Added hello.xml in GitWork"

Switch to master:

git checkout master

Create file in master:

echo "Hello from master" > hello.xml

git add hello.xml

Commit:

git commit -m "Added hello.xml in master"

View logs:

git log --oneline --graph --decorate --all

Check diff:

git diff GitWork

Use P4Merge (after setting up):

git mergetool

Merge:

git merge GitWork

git add hello.xml

git commit -m "Resolved merge conflict in hello.xml"

Add backup file to .gitignore:

echo "\*~" >> .gitignore

git add .gitignore

git commit -m "Added backup files to .gitignore"

List branches:

git branch

Delete merged branch:

git branch -d GitWork

View final log:

git log --oneline --graph --decorate

**Task 5: Sync with Remote GitLab Repository**

Verify clean state:

git status

List branches:

git branch

Pull latest changes from remote:

git pull origin master

Push changes from local branch :

git checkout Git-T03-HOL\_002

git push origin Git-T03-HOL\_002