

GIT Introduction

GitBash

- Download if not already downloaded.
- <https://git-scm.com/downloads>
- Follow installation document and install it.

GitLab Login

- <https://gitlab.com/>
- Create new user (if not created earlier) as per installation document.
- If user is already created, you may change user name as per convention. (Refer another document).

Using GitBash (Simple Example)

- Create a new folder to store your project files.
- Open GitBash in that folder and "init" git repository.
 - `git> git init`
- Create file f1.txt.

```
abcd
```

- Stage and commit the file.
 - `git> git status`
 - `git> git add f1.txt`
 - `git> git status`
 - `git> git commit -m "first commit"`
- If you are using git for first time you may need to set your Username and Email. This information will be recorded into all GIT commits. Replace with your name and email id below.
 - `git> git config --global user.name "Nilesh Ghule"`
 - `git> git config --global user.email "nilesh@sunbeaminfo.com"`
 - `git> git config --global --list`
 - After setting this give last "git commit" command again.
- Verify commit details.
 - `git> git log`
- Create file f2.txt.

```
this is f2 file.
```

- "Append" into f1.txt.

```
pqrs
```

- Stage and commit the file.
 - git> git status
 - git> git add f1.txt f2.txt
 - git> git status
 - git> git commit -m "second commit"
- Create new file f3.txt

```
this is f3 file.
```

- Edit into f1.txt. pqrs --> pqr. Test the file difference.
 - git> git diff f1.txt
- Restore contents of f1.txt
 - git> git restore f1.txt
- Stage and commit the file.
 - git> git status
 - git> git add f3.txt
 - git> git status
 - git> git commit -m "third commit"
- Create new file f4.txt

```
this is f4 file.
```

- Create new file f5.txt

```
this is f5 file.
```

- Commit each file in separate commits.
 - git> git status
 - git> git add f4.txt
 - git> git status
 - git> git commit -m "fourth commit"
 - git> git status
 - git> git add f5.txt
 - git> git status
 - git> git commit -m "fifth commit"
- Check commit history.
 - git> git log
 - You can scroll through to see all commits.
 - You may need to press "q" to come out of log command.

- For next demo refer code from -- <https://gitlab.com/nilesh-sip/demo>
- Create file main.c -- with hello world message. (You may compile and execute. Compilation is not needed for GIT testing).
- Stage and commit the file.
 - `git> git status`
 - `git> git add main.c`
 - `git> git status`
 - `git> git commit -m "main() implemented"`
- Add menu-driven part in main.c (do-while and switch-case).
- Stage and commit the file.
 - `git> git status`
 - `git> git add main.c`
 - `git> git status`
 - `git> git commit -m "menu driven added"`
- Create files calc.h, calc.c with sum() and subtract functions. Also call them from main.c. (You may compile and execute. Compilation is not needed for GIT testing).
- Stage and commit the files (in two different commits).
 - `git> git status`
 - `git> git add calc.h calc.c`
 - `git> git status`
 - `git> git commit -m "add & subtract fns"`
 - `git> git status`
 - `git> git add main.c`
 - `git> git status`
 - `git> git commit -m "called add & subtract"`
- Create a .gitignore file to skip files like *.exe, *.obj, etc.
- Stage and commit the file. Note that "." will add all changes in current directory into staging area.
 - `git> git status`
 - `git> git add .`
 - `git> git status`
 - `git> git commit -m "gitignore added"`

Uploading Local Git Repo on GitLab.

- On GitLab, Login with your username and password.
- Menu --> Projects --> Your Projects.
- Click on "New Project".
- On next screen Click on "Create blank project".
- Fill project details like project name, Project Visibility = Private and Uncheck README file and "Create Project".
- On GitBash (Ensure that you are still in last project directory). Replace below GIT URL with your project GIT URL.
 - `git> git remote add origin https://gitlab.com/nilesh-sip/demo.git`
 - `git> git push -u origin master`
- Above push command might ask you username and password (if using GitLab id for first time). Enter "GitLab" username and password.

Guidelines for New Projects (for classwork practice or assign evaluation)

- Create project on GitLab (as mentioned in above steps). You may "check" README file option.
- Copy Clone --> "Clone with HTTPS" link from project GitLab page.
- Decide directory in which you want to create local repository on your machine. Go to that directory and Right click "Git Bash Here".
- `git> git clone https://gitlab.com/nilesh-sip/java-assign.git`
 - Replace above URL with URL of your project (copied in earlier step).
 - Close this terminal.
- Use your favorite IDE (like VSCode or Eclipse) to implement the project. On each logical step do commits. You may need to use following commands on each logical task completion. You need to open GitBash from the local repository directory.
 - `git> git add .`
 - `git> git commit -m "meaningful commit message"`
- It is recommended to push the assignment changes daily (not necessarily after each commit).
 - `git> git push origin main`
- If this project is meant to be evaluated add your lab mentor (their ID will be shared with you) in your GitLab repository as a "Developer".
- You need to submit your assignment for evaluation on Moodle LMS. We will conduct separate lecture for Moodle LMS usage.

Guidelines for Daily Classwork (Uploaded by Lecture Faculty)

- You must be added under Sunbeam groups under GitLab to follow these steps. Please follow instructions from Course Coordinator for the same.
- Each module GitLab repository link will be shared on Zoom channel at the start of module.
 - <https://gitlab.com/pg-dac/cop>
 - <https://gitlab.com/pg-kdac/cop>
 - <https://gitlab.com/pg-dmc/programming-concepts>
 - https://gitlab.com/pg-dbda/programming_concepts
- Visit the link and copy Clone --> "Clone with HTTPS" link.
- Decide directory in which you want to create local repository on your machine. Go to that directory and Right click "Git Bash Here".
- `git> git clone https://gitlab.com/pg-dac/cop`
 - Replace above URL with URL of your project (copied in earlier step).
 - Close this terminal.
- Each day classwork will be uploaded by the Faculty on GitLab repository. For subsequent days you need to pull the changes. You need to open GitBash from the local repository directory.
 - `git> git pull`
- Students are not expected to do any changes in this repository. You can copy it and/or implemented into your project (at some other folder location).