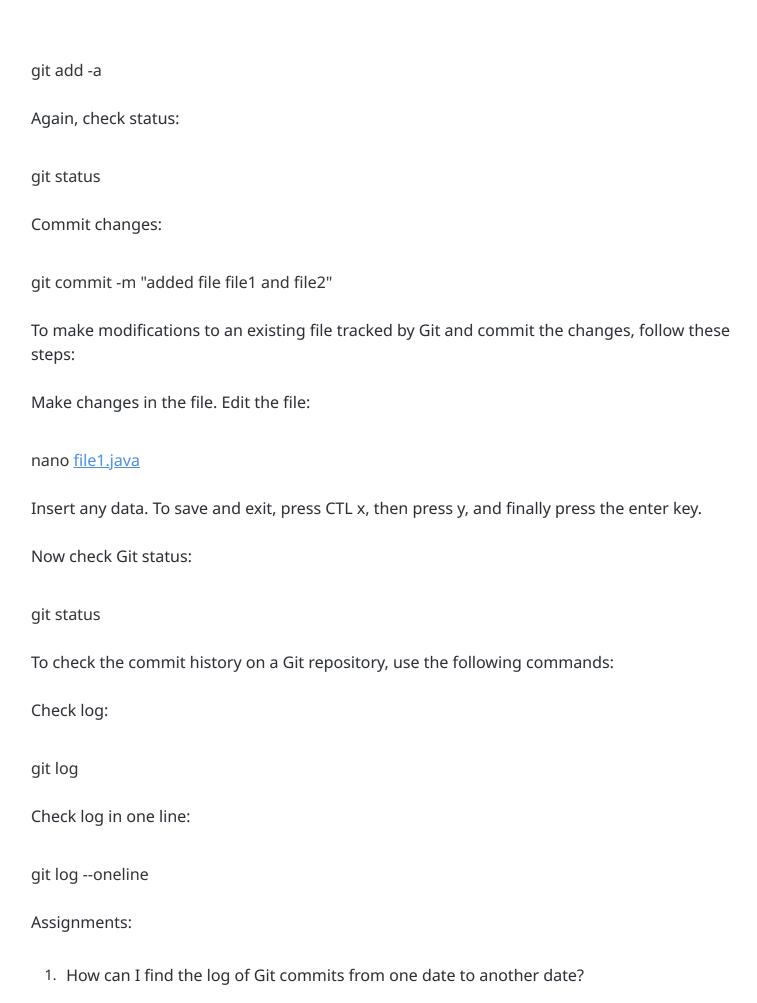
Git Command Notes
To become a root user (applicable for Linux only):
sudo su -
To create a working directory:
mkdir myproject
To navigate into the directory:
cd myproject
To create empty files:
touch <u>file1.java</u> <u>file2.java</u>
To create a local repository:
git init
For Git configurations, use the following commands:
Set user name:
git configglobal <u>user.name</u> malay
Set user email:
git configglobal <u>user.email</u> <u>malay@gmail.com</u>
There are three types of configurations:

Config Variables: <u>user.name</u> and <u>user.email</u>

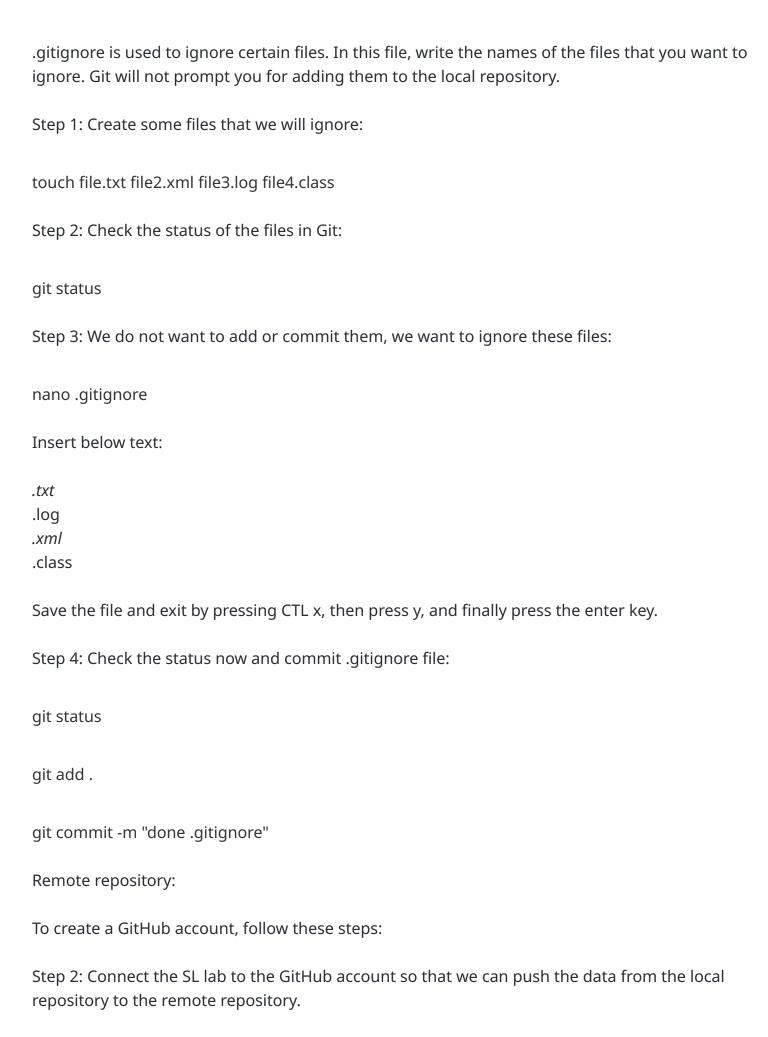
Git Notes

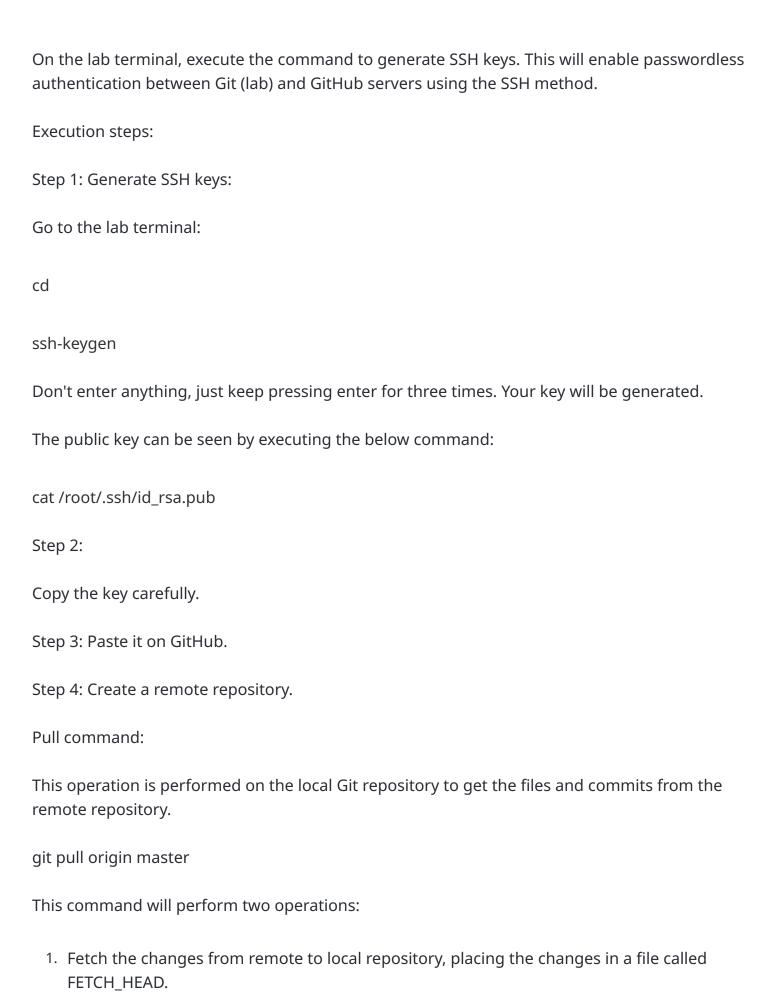
Local: If the configuration is set with a flag as local, the config variable will apply only to that particular repository.
Global: If the configuration is set with a flag as global, the config variable will apply to every repository created by the current user logged in the machine.
System: If the configuration is set with a flag as system, the config variable will apply to every repository created by every user logged in the machine.
To remove the configurations, use the following commands:
Remove user name:
git configglobalunset <u>user.name</u>
Remove user email:
git configglobalunset <u>user.email</u>
To find out the location where the Git configurations are saved, use the following commands:
For local:
For global: cat ~/.gitconfig
For system:
To add a file to the local repository, use the following commands:
Check status:
git status
Add all files:
git add .
Or

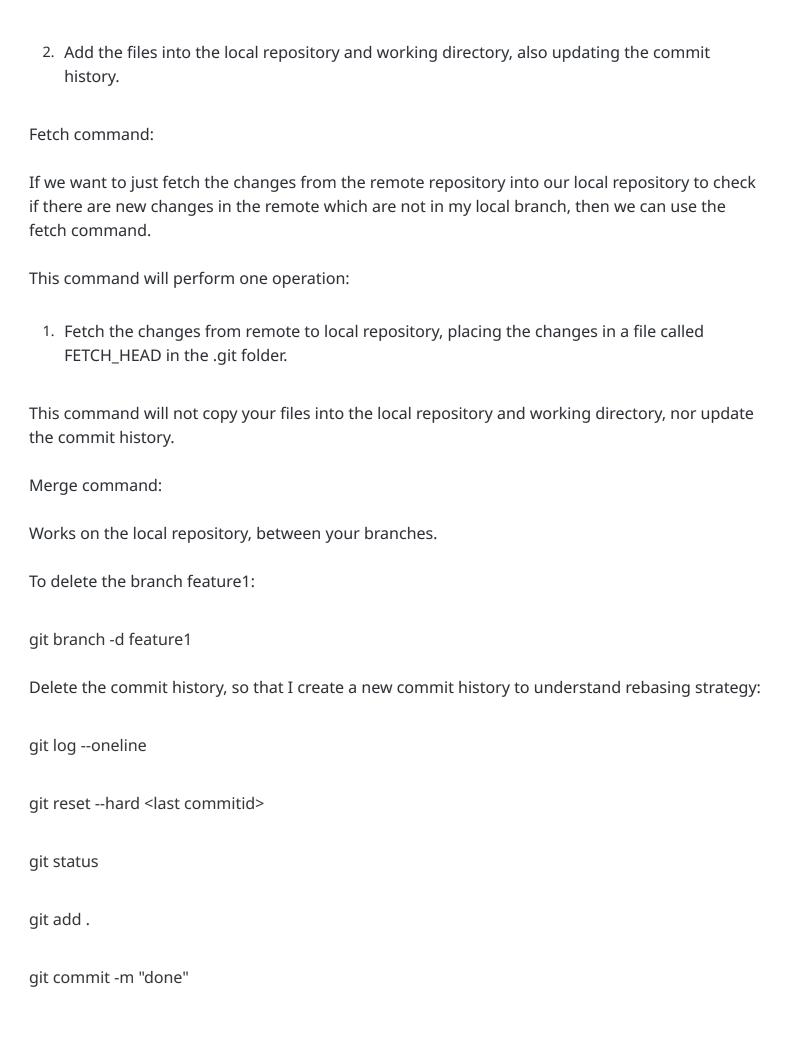


Use the following commands:
since= <date>,after=<date> Show commits more recent than a specific date.</date></date>
until= <date>,before=<date> Show commits older than a specific date.</date></date>
1. Move/copy/push your Git log history into a new text file.
To delete an existing file from the local repository and working directory, follow these steps:
List files:
git ls-files
Choose a filename to be deleted.
Execute the Git delete command to delete a file from the local repository and working directory:
git rm <filename></filename>
For example, to delete <u>file1.java</u> :
git rm <u>file1.java</u>
The file will be removed from the local repository and working directory.
List files:
git ls-files
List directory contents:
Is
Commit the deletion of the file:

git commit -m "deletion of file"
Check log in one line:
git logoneline
To revert changes, use the Git revert command. This command always works on a commit id. It reverts the changes made in the commit back to its original state. When you execute the revert command, Git automatically creates a commit, asking you to enter a message.
Execution steps:
Step 1: Check the commit id that you want to revert.
Take the deletion commit id:
git logoneline
Copy the seven digits, for example, c9bf927.
Step 2: Execute the revert command:
git revert c9bf927
Press the enter key, it will open a nano editor file asking you to insert some message.
Save the file: press CTL x, press y, and finally press the enter key.
Step 3: The file will be back in the local repository and working directory.
List files:
git ls-files
List directory contents:
Is







Check log in one line:
git logoneline
Rebase Command:
Γhis is a merging strategy.
1. Create a new branch feature1 (the base of which is master) and switch to the branch:
git branch
git checkout -b feature1
1. Create new commits on feature 1 branch:
couch file-feature
git add .
git commit -m "done on feature"
1. Switch to master and create new commits:
git checkout master
couch file-master
git add .
git commit -m "done on master"
1. No merging of feature1 on master is done.

2. Switch back to feature1 and do the rebase command (reorganize the base branch commits):

git rebase master
Lesson end project:
Come out of the current directory:
cd
Create a directory with includes the files:
mkdir