# Full Stack Development
## Lab Management 1

# Basic git commands

## 1. Lab objectives

This lab is intended for you to get a solid understanding of git basics

## 2. Basics

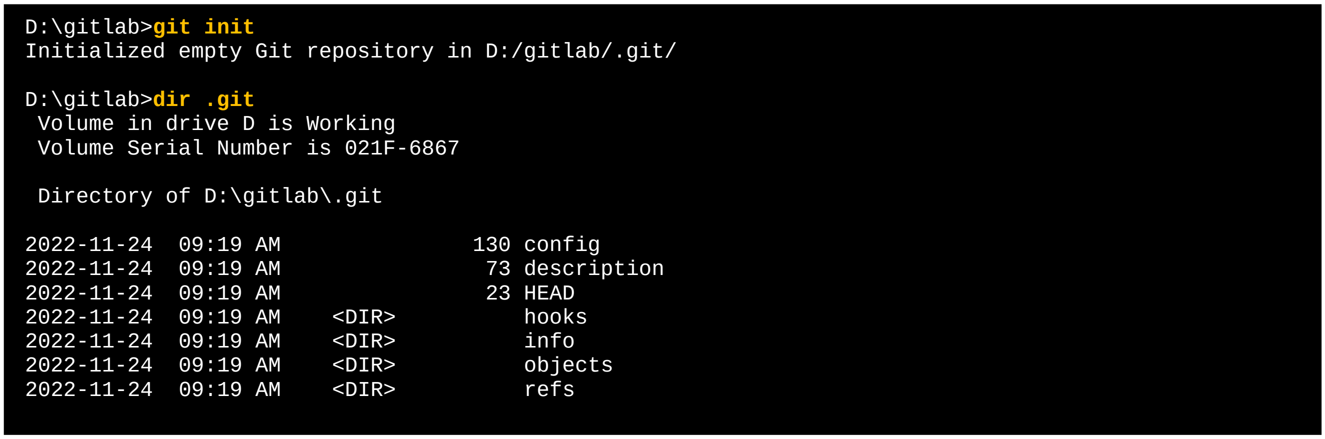
Create a new directory, locate to it and run the git status command. Because the directory is not a repository, any git command will recognize that.

```
D:\>mkdir gitlab

D:\>cd gitlab

D:\gitlab>git status
fatal: not a git repository (or any of the parent directories): .git
```

However, your lab machine does have some global configurations set that you can display.

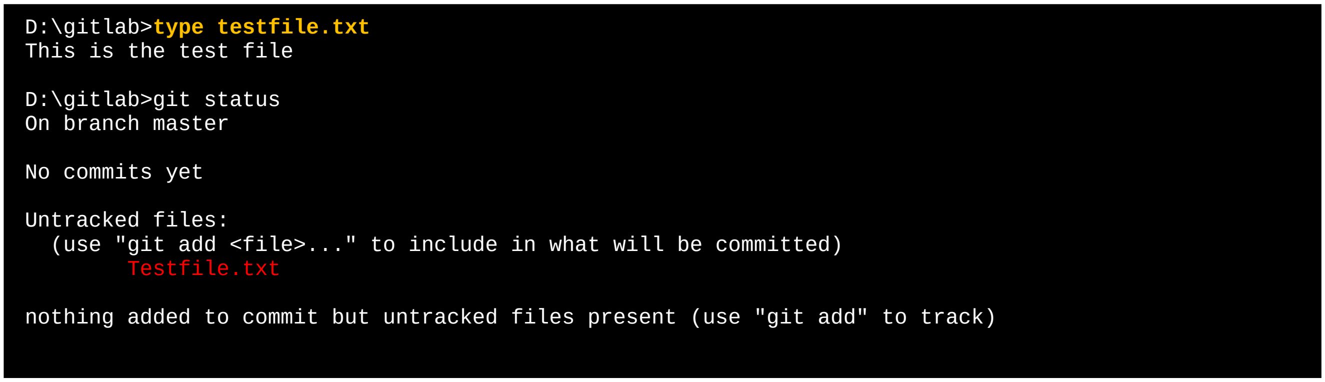
```
C:\Users\Student>git config --global --list
core.editor="C:\Users\Student\AppData\Local\Programs\Microsoft VS Code\bin\code" --wait
user.name=Student
user.email=Student@dummyemail.com
```

Initialize the repository and display the .git directory



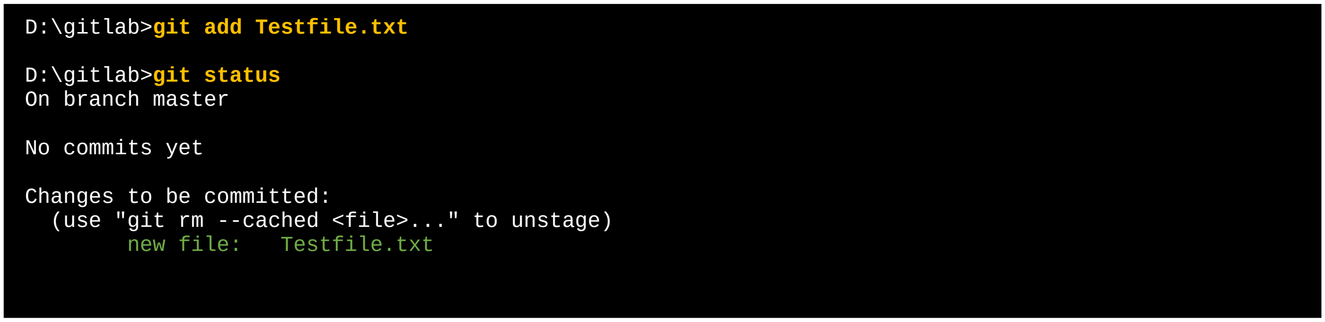
```
D:\gitlab>git init
Initialized empty Git repository in D:/gitlab/.git/

D:\gitlab>dir .git
 Volume in drive D is Working
 Volume Serial Number is 021F-6867

 Directory of D:\gitlab\.git

2022-11-24  09:19 AM               130 config
2022-11-24  09:19 AM                73 description
2022-11-24  09:19 AM                23 HEAD
2022-11-24  09:19 AM    <DIR>          hooks
2022-11-24  09:19 AM    <DIR>          info
2022-11-24  09:19 AM    <DIR>          objects
2022-11-24  09:19 AM    <DIR>          refs
```

Create a file and run git status. This example uses “testfile.txt.”



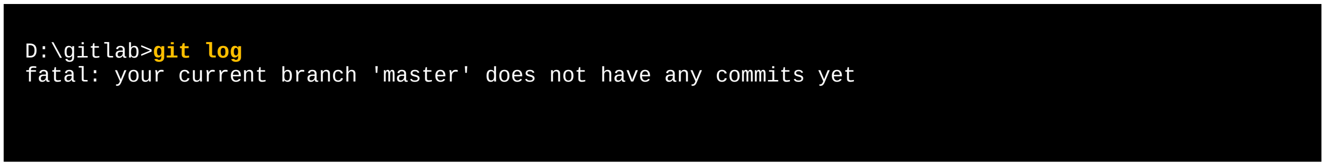
```
D:\gitlab>type testfile.txt
This is the test file

D:\gitlab>git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        Testfile.txt

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

Git is telling you that it is not tracking changes on this file. To stage the file, use git add then run git status again. This does not put the file into the repository but is a preparation for a commit.



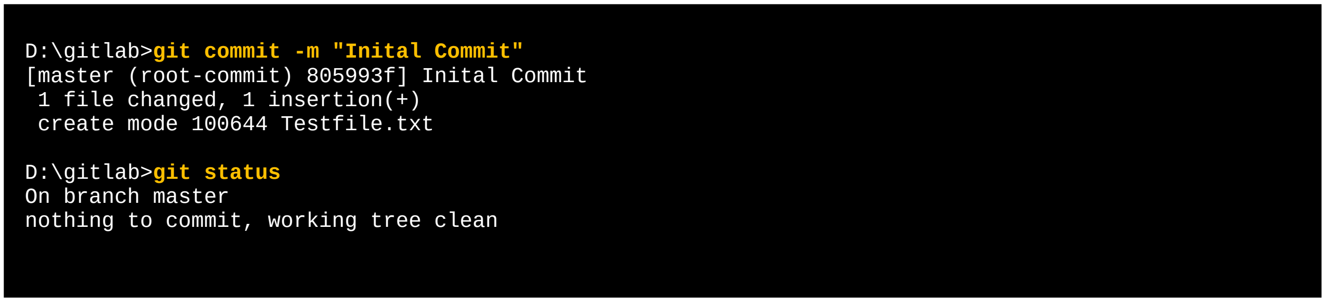
```
D:\gitlab>git add Testfile.txt

D:\gitlab>git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   Testfile.txt
```

Examine the state of the current repository using the git log command to see that there is nothing in the repository. Adding the file did not put it into the repository



```
D:\gitlab>git log
fatal: your current branch 'master' does not have any commits yet
```

You can add the file using the git commit command.  Note that a commit message is mandatory.

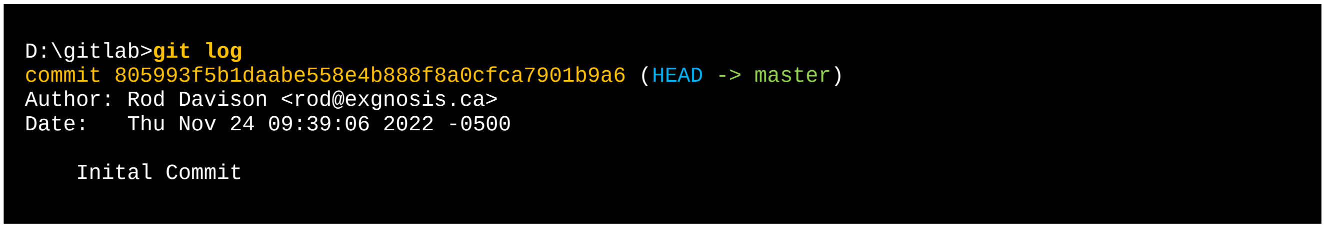


```
D:\gitlab>git commit -m "Inital Commit"
[master (root-commit) 805993f] Inital Commit
 1 file changed, 1 insertion(+)
 create mode 100644 Testfile.txt

D:\gitlab>git status
On branch master
nothing to commit, working tree clean
```

Git is now telling you that any changes that were in the staging area are now in the repository.

```
D:\gitlab>git commit -m "Inital Commit"
[master (root-commit) 805993f] Inital Commit
 1 file changed, 1 insertion(+)
 create mode 100644 Testfile.txt

D:\gitlab>git status
On branch master
nothing to commit, working tree clean
```

Confirm the status of the repository with git log again



```
D:\gitlab>git log
commit 805993f5b1daabe558e4b888f8a0cfca7901b9a6 (HEAD -> master)
Author: Rod Davison <rod@exgnosis.ca>
Date:   Thu Nov 24 09:39:06 2022 -0500

    Inital Commit
```

Note that the hash for the commit has been produced, and the commit message, complete with spelling mistake is displayed

## 3. Aliases

Typing out the same commands can be tedious. Git aliases allow you to create short cuts for various commands. For example, to create and use an alias for “git status” do the following. Note the --global flag sets the alias for all your repositories.

```
D:\gitlab>git config --global alias.s status

D:\gitlab>git s
On branch master
nothing to commit, working tree clean
```

A useful alias for display the git log is shown below. This produces a nice compact visual display.

```
D:\gitlab>git config --global alias.ll "log --oneline --graph --decorate --all"

D:\gitlab>git ll
* 805993f (HEAD -> master) Inital Commit
```

To remove an alias, just “unset” the alias

```
D:\gitlab>git s
On branch master
nothing to commit, working tree clean

D:\gitlab>git config --global --unset alias.s

D:\gitlab>git s
git: 's' is not a git command. See 'git --help'.
```

# End Lab