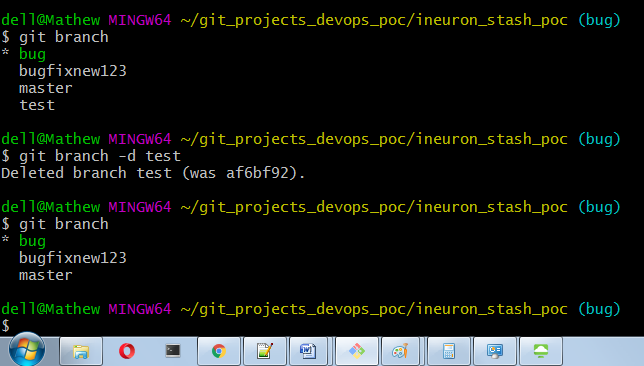
**GIT Assignment 4**

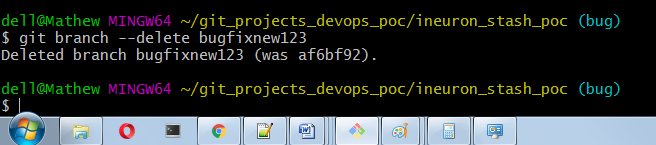
1. How can new features be added to the main branch? What is a git conflict?

* The main idea behind new feature is that we do not want to make changes directly into main branch. Developers create a sub-branch where they can make/develop/test their code which will not impact the main branch.
* once the new code /feature is now ready we can merge that into main branch
* **git checkout -b new\_feature** this will create a branch with name new\_feature\*
* Now developer can update/add/commit/push the source code
* git init
* make some changed\_add/develop new features
* git add .
* git commit -m "type\_comments"
* git push -u <branch> new\_feature

1. How do I remove a branch from GIT?

* Command to remove branch is below two ways
* **git branch -d branch\_name**
* **git branch --delete branch\_name**
* just replace the branch\_name with the branch you want to delete .
* As usual below is reference poc :)





1. What is the purpose of the git checkout command?

* **git checkout** command is used to switch from one branch to other branch
* for eg.
* **git branch** -> to show what all branches available and with \* means you are currently in that branch
* **git branch bugfix** -> this will create a branch with the name bugfix
* **git checkout bugfix** -> this will switch/change branch form master branch or anyother branch in which you are currently in to bugfix
* **git branch** -> now again you can very list all branch and \* means that you will be in that branch in this case it is bugfix
* **git checkout -b bugfixnew** -> this will create a bugfixnew branch and will also switch to bugfixnew branch so in one single command create and switch we can use together.
* **git switch master** -> **after version 2.23** **onwards** we can use switch keyword to switch the branch, else you are used to checkout then it's fine
* **git switch -c TEST** -> this will create a TEST branch and will branch also switch to bugfixnew branch so in one single command, this is same as above

1. What is the purpose of the git commit command?

* **git commit** will capture the snapshot of currently staged changed in to current repository
* there are three stages(working,stage,commit) so first when we create a file and add some content in that is working then do
* **git add <file\_name>** or **git add .** to add all files in this path use **full stop** at end
* **git commit** - this will commit the changes and it will open a new terminal where we can mention what changes are done.. to avoid that we can used
* **git commit -m "Type you message or changes done"**

1. What is the purpose of the command 'git rm'?

* The git rm command is used to remove individual files or multiple files.
* The primary function of git rm is to remove tracked files from the Git index staging area
* after removing files you need to do a commit.
* ther are few parameters/arguments which are available and can be passed based on requirements.
* Also if you have by mistake added file to staging state and you want to remove from staging state nad revert back to working are then you can do that with these command git rm --cachced filename
* You can also pass file or folder/path names.

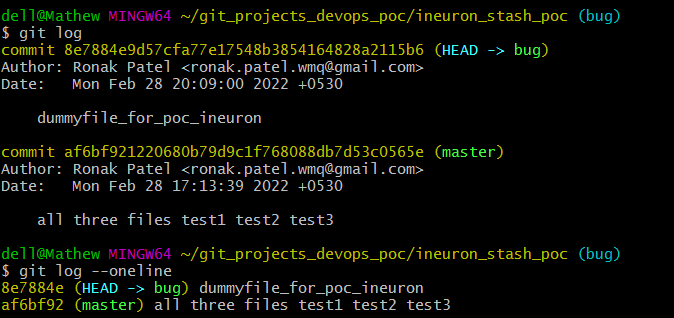
1. What is the purpose of the git log command?

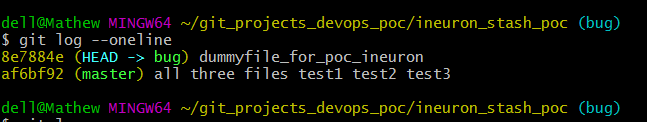
* To see the history of any source code like what changes are done by which user with commit id, author, date time etc.. and also we can use git log for multiple filtering purpose.

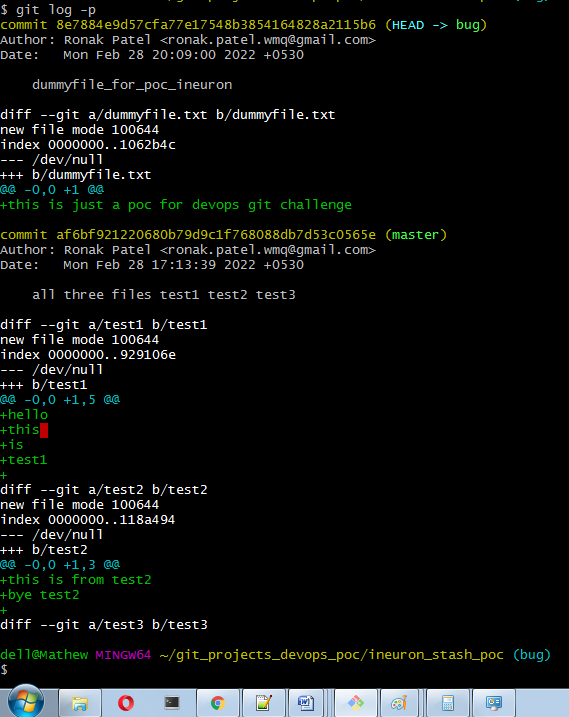
eg. **git log  
git log --oneline**

**git log -p**

**git log --path**







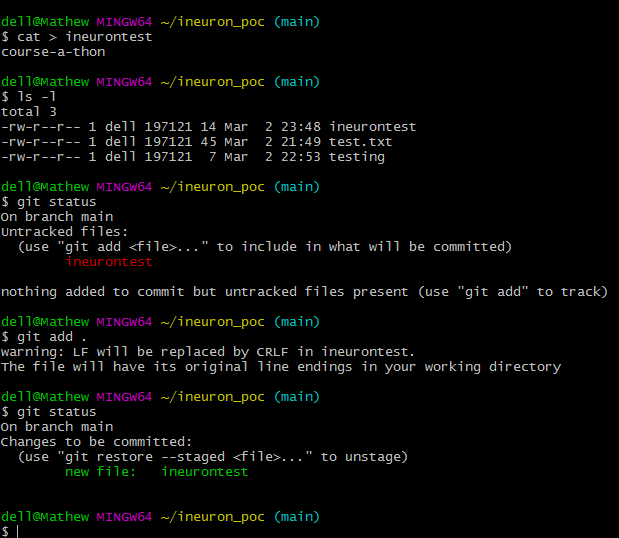
1. What is the purpose of 'git add'?

* **git add** command will add change from working directory/area to staging area

syntax is git add filename or ig you want to add all files in that directory then use

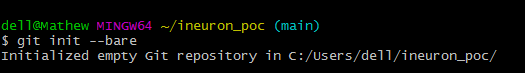
**git add .**

* See the difference before add the new file it is in working directory and after adding it is in staging directory we can see the colour difference.

****

1. In GIT, what is a 'bare repository'?

* In the simplest terms, a bare repository is the contents of your project’s .git directory and nothing else. It has no working directory.
* No commits can be made
* A bare git repository is used as a remote repository where code is shared between different team members
* The bare Git repo is not intended for local development.



1. What's the difference between git remote and git clone?

* remote repository are version of your project that are hosted on internet or network somewhere
* git remote add just creates an entry in your git config that specifies a name for a particular URL
* with git remote we can do three type of operations
* List all remote : **git remote -v**
* adding remote : **git remote add origin\_name https\_link/ssh**
* delete remotes : **git remote remove origin\_name**
* git clone is a git command line utility which is used to target an existing repository and create a clone of the target repository.
* In other words git clone  creates a brand new git repository by copying an existing one located at the URI you specify.
* **git clone https\_url** (connection-string started with https end with .git)
* **git clone ssh\_link** (starting with git@ end with .git)