

19/7/24

classmate

Date \_\_\_\_\_  
Page \_\_\_\_\_

## \* Unstaging Changes

- amend (undo a commit)

→ when a change is committed, & we realise we want to further modify the committed file, use this command

```
git commit -m "first commit"
git add m - file
git commit --amend
```

replaces the first commit with this effectively, the first commit never happened & won't show up in history

- \* for undoing staged files changes, run git status & it will give us the further steps.

(It directs use to use git checkout, git restore as per our position to unstage)

- git remote add <name> <url>

remotes

```
git push <remote> <branch>
```

- The remotes can be renamed & removed.

- for info of remote:

```
git remote -v
```

```
git remote show <remote name>
```

## \* GIT TAGS

- git tag -a <tagname> -m "message"

→ create annotated (recommended) tags (push/pull)

→ we can perform & keep track of tags easily by using git tag <tagname>



19/7/24

## \* GIT ALIASES:

- When using git regularly, aliases are something we should know about.
- Custom shorthand (by you) for any git command.  
*using git config*

For eg:

git config --global alias.co checkout  
git config --global alias.br branch

Now, can use git co & git br for git checkout & git branch respectively.