

Some Examples of Command Line Git Use

Get a copy of a repo as a specific user: e.g.

```
git clone https://DuncanRowland@github.com/BrionyStudent/Test_1.git
```

- 1) Make some local edits (e.g. add a new image, say "happy.jpg")
- 2) Add that image, so that it is staged (and tracked by git locally): `git add happy.jpg`
- 3) Commit the changes (locally save to git): `git commit -m "Added a happy image"`

If you are a "Collaborator" then you will be able to upload these changes to the repo

```
git push
```

However, if someone else has upload to the repo, you have to download their stuff first...

```
git pull
```

...and this might cause conflicts. To resolve these, either

- 1a) Manually edit the files (which, if text, will now contain a combination of **both** versions)
- 1b) or, checkout either 'their' version or '(y)ours'

```
git checkout --ours -- happy.jpg  
git checkout --theirs -- happy.jpg
```

You then need to stage and commit this change, and again push to the server

```
git commit -a -m "Resolved happy image conflict"  
git push
```

Branches are used all the time in git, locally (for personal use) and remotely (to share).
Keep the 'master' branch is special, see it as the 'release' version.

To make a new (local) branch: `git branch Duncan`

Optionally, to tell the server about this new branch: `git push -u origin Duncan`

To start using the branch: `git checkout Duncan`

- 1) Make some edits (e.g. add a new image, say "happy.txt")
- 2) Add that file, so that it is staged (and tracked by git locally): `git add happy.txt`
- 3) Commit the changes (locally save to git): `git commit -m "Added happy text"`

If the server knows about the branch, you can update that too: `git push`

How to merging changes from one branch into another (e.g. 'Duncan' into 'master')?
I'm assuming here that Duncan is your local branch, whilst master in on the server.

- 1) Switch to the master branch: `git checkout master`
- 2) And make sure it is up to date (resolving/committing any conflicts): `git pull`
- 3) Switch to the Duncan branch: `git checkout Duncan`
- 4) Make sure you are up to date with the master (resolve any conflicts): `git merge master`
- 5) Stage and commit the changes: `git commit -a -m "Duncan up to date with master"`
- 6) Switch to the master branch: `git checkout master`
- 7) Merge in the changes: `git merge Duncan`
- 8) Stage and commit the changes: `git commit -a -m "Merged Duncan"`
- 9) You can update the server: `git push`

Finally, to delete a local branch (but note, committed files still in repo): `git branch -d Duncan`
and to remove it from the server: `git push origin --delete Duncan`

For more, checkout ungit (<https://www.youtube.com/watch?v=hkBVai3oKvo>), and other learning resources: <https://try.github.io/levels/1/challenges/1> and <http://pcottle.github.io/learnGitBranching/>