

Create a New Repository*git init***Grab a Local Copy of GitHub Repo***git clone repo-url* (e.g. '*git clone https://github.com/ergsense/DTECTS_hw.git*') **Check Repo Status** (check for changes)*git status***Commit new content***git add ** ('*' for all new content, specific names otherwise)
*git commit***Push to Remote Repo***git push***Pull from a Remote Repo***git fetch* (first to sync)
git pull -all (second to grab)**Reset Your Repo** (i.e. reset back to HEAD)*git reset --hard***Switch to a Previous Commit***git checkout sha-id*

Assumptions

- 'Repo' means local content here unless explicitly stated
 - e.g. "adding to the repo" means adding to your local .git/ repo copy

Notes

- ('-f') Anytime work gets sticky and things won't successfully complete this gets easy, just force it
 - e.g. for the classic eggshell of a failed '*git push*', just use '*git push -f*'!
- ('-a') Be sure to you are on correct branch before interacting with the remote repo.
 - If you want to interact with the whole repo, just use '-a' for all!

1. Create - New Local Repo

Here we will create a new repo locally and use it, adding an amend & reset for illustration.

```
<create>      - Create dir & place init content
               *note: for empty dirs, place an empty.txt inside for repo retention

git init       - Initialize repo
git add *      - Add content
git status     - Confirm correct staging (new files, removed files, ignored content, etc.)
git commit     - Stores the added changes to a new commit, yeilding a sha-id for records
git status     - Confirm commit complete as intended
...
git commit --amend - Update commit's content, or change it's commit message
git reset --hard   - Reset the repo to it's initial state (value of commit <sha-id>)
```

2. Create – New GitHub Repo

Here we will create a new GitHub repo and add initial content, in preparation for future use.

Go to <https://github.com/>

Select 'New Repository' and enter a Repository name

Select 'Create Repository'

Use the created HTTPS value to checkout a local copy

e.g. '*git clone https://github.com/justinmreina/test.git*' <- The repo is now ready for use

3. Add – New Stuff to Repo

Here we update the project a bit with new content, and commit this to the local repo.

```
git add somefile1.txt somefile2.txt lib_dir/*  - add two files and an entire directory
git status                                     - Confirm correct staging
git commit                                     - Store the added changes to a new commit
git status                                     - Confirm commit complete as intended
```

4. Push - To GitHub

Here we push some new content (commits) to the GitHub remote repo.

```
git push      - Push to remote repo. Use '-f' if tricky or painful!
```

5. Pull – From Github

Here we pull current content (commits) from the GitHub remote repo. This is done anytime you want to refresh or catch-up, e.g. to new content from the team, etc.

```
git pull      - Push to remote repo. Use '-f' if tricky or painful!
```

6. Update – Copy of Remote GitHub Repo

Simple, '*git pull*'!