Git commands

Tobias Slade-Harajda

August 2023

1 Order of using git

1. Make/copy repo
2. Change to repo dir
3. Change to/create branch if necessary, §5
4. Work in repo, edit files and make changes
5. Add changes ready to commit, §3
6. Commit changes, §3
7. Push changes, §4
8. If working on branch, follow final steps in §5

2 Creating a git repo

2.1 Local repo

Creating local repo (only need to run once on first ever git repo creation)

```
git config --global init.defaultBranch main
```

Making a repo and changing dir

```
mkdir <repo name>
cd <repo name>
git init
```

See §6 for other commands.

Include a README.md, privacy statement, licensing file and accessing rights. Also include a .gitignore file which will ignore certain file extensions or files which, when changes are made, aren't pushed to the GitHub.

2.2 Copied repo

Add a copy of a repo already made on GitHub

```
git remote add origin <YOUR REPO URL from GITHUB>
```

3 Changes to repo

Add specific files of those changes made in the repo

```
git add <name of file>
```

Add all changes made in repo

```
git add --all
```

Commit changes with a message

```
git commit -m "<message>"
```

For pushing see §4

If local repo is deleted or want a copy of a repo already on GitHub

```
git clone https://github.com/<username>/<repo-name>
```

4 Pushing

First time pushing

```
git push --set-upstream origin main
```

Subsequent pushing

```
git push
```

5 Creating & switching to new branch

Creating a branch locally

```
git switch --create <branch name>
```

Pushes of new branch

```
git push --set-upstream origin <branch name>
```

Subsequent pushes of branch follows §4

Switching between branches

```
git switch <branch name>
```

To merge branches, once all changes are made, you need to open a merge request on GitHub. Once the branches are merged, head back to the local repo, switch to main/master branch and delete old branch. Follow these steps

```
git switch main
git pull
git branch -d <branch name>
```

6 Other commands

Check status of repo, compared to the linked GitHub

```
git status
```

Setting name and email address associated with git repo

```
git config --global user.name "YOUR NAME"
git config --global user.email "your@email.address"
```

If you want to edit files in a particular editor (e.g. “nano”)

```
git config --global core.editor "<name of editor>"
```

Comparing changes between local files and those stored on the repo

```
git diff
```

Undo changes that aren't staged/committed

```
git restore <file>
```

Undoing staged changes (files that have been added)

```
git restore --staged <file>
```

Identify all commits and pushes, see activity in the repo

```
git log
```

Revert back to a specific push/commit

```
git revert <first 7 digits of commit id>
```