



Git Commands Cheat Sheet

1. Check the status of your repo

```
git status
```

Purpose:

Shows the current branch, staged/unstaged changes, and untracked files.

Example output:

```
On branch main
Your branch is ahead of 'origin/main' by 1 commit.
Changes not staged for commit:
  modified: test.py
```

2. View your remotes

```
git remote -v
```

Purpose:

Lists all remotes configured for fetch and push.

Example output:

```
origin  https://github.com/user/repo.git (fetch)
origin  https://github.com/user/repo.git (push)
```

3. Rename a remote

```
git remote rename origin student
```

Purpose:

Renames the `origin` remote to `student` to distinguish between student and teacher repos.

Output: *(none if successful)*

4. Pull changes from student repo

```
git pull student main --no-edit
```

Purpose:

Updates your local branch with changes from the `student` repo's `main` branch.

- `--no-edit` keeps the default merge message.
- `--no-rebase` forces a merge instead of rebase.

Example output:

```
Updating 123abcd..456efgh
Fast-forward
 test.py | 1 +
 1 file changed, 1 insertion(+)
```

5. Pull changes from teacher repo

```
git pull teacher main --no-edit
```

Purpose:

Updates your local branch with changes from the `teacher` (template) repo.

6. Stage all changes

```
git add .
```

Purpose:

Stages **all modified and new files** in the current directory for the next commit.

Output: *(none if successful)*

7. Stage a specific file

```
git add <filename>
```

Purpose:

Stages only the specified file for commit.

Example:

```
git add test.py
```

8. View your Git global settings

```
git config --global --list
```

Purpose:

Shows configuration settings like username, email, and pull strategy.

Example output:

```
user.name=Bob Slotpole  
user.email=rslotpole@babson.edu
```

9. Add a teacher remote

```
git remote add teacher https://github.com/babson-org/classroom-week00-  
python_class.git
```

Purpose:

Adds a new remote called `teacher` for fetching updates from the teacher's template repo.

Note: Only needs to be done **once per repo**.