

Git Staging Environment

One of the core functions of Git is the concepts of the Staging Environment, and the Commit.

As you are working, you may be adding, editing and removing files. But whenever you hit a milestone or finish a part of the work, you should add the files to a Staging Environment.

Staged files are files that are ready to be committed to the repository you are working on. You will learn more about `commit` shortly.

For now, we are done working with `git_tuto 2`. So we can add it to the Staging Environment:

```
git add git-tuto 2
```

The file should be Staged. Let's check the status::

```
git status
```

```
On branch master
```

```
No commits yet
```

```
Changes to be committed:
```

```
(use "git rm --cached <file>..." to unstage)
```

```
new file:   git_tuto 2.pdf
```

Now the file has been added to the Staging Environment.

Git Add More than One File

You can also stage more than one file at a time. Let's add 2 more files to our working folder. Use the text editor again.

Now add all files in the current directory to the Staging Environment:

```
git add --all
```

Using `--all` instead of individual file names will `stage` all changes (new, modified, and deleted) files.

```
git status
```

```
On branch master
```

```
No commits yet
```

```
Changes to be committed:
```

```
(use "git rm --cached <file>..." to unstage)
```

```
new file:   git_tuto 2.pdf
```

```
new file:   git_tuto 3.pdf
```

Now 2 files are added to the Staging Environment, and we are ready to do our first `commit`.

Note: The shorthand command for `git add --all` is `git add -A`