

Introduction

Now that we've talked a bit about branches, let's go ahead and put it into practice. You will learn about organizing branches: adding, deleting, renaming branches, and so on.

Listing Branches

In your terminal, type

```
git branch -a
```

This lists all our current branches.

Organizing branches

Let's make a new branch with the command

```
git branch featureA
```

Do

```
git branch -a
```

again. You can see the new branch was added. The asterisk besides `master` denotes that we're currently on the master branch.

To switch to the new `featureA` branch type

```
git checkout featureA
```

The `checkout` command, followed by the branch name, will switch over to that particular branch, so checkout is Git's way of saying, "switch to".

Typing in

```
git status
```

tells us that we're currently on the `featureA` branch, and that there's nothing to commit.

Now, let's make another branch:

```
git branch featureB
```

Type

```
git branch -a
```

You will see the new `featureB` branch that we just made.

Let's switch to the `featureB` branch with the `checkout` command:

```
git checkout featureB
```

There is a faster way of making and checking out a branch all in one command:

```
git checkout -b featureC
```

Now we've made and checked out into the `featureC` branch.

Practice that once again:

```
git checkout -b extra
```

Now let's rename our `extra` branch to `branch_to_delete`.

```
git branch -m extra branch_to_delete
```

Now, running

```
git branch -a
```

displays the new name.

You can't delete a branch that you're currently checked out on. Therefore switch to the `master` branch:

```
git checkout master
```

Delete our branch:

```
git branch -d branch_to_delete
```

Running

```
git branch -a
```

shows us that the branch was deleted.