Working with Git on the command line can be a little bit challenging because it's a little bit like a black box. I mean, how do you know when you should or shouldn't run certain Git commands? Is Git ready for me to run a command yet? What if I run a command but I think it didn't work...how can I find that out? The answer to all of these questions is the `git status` command!

```
$ git status
```

The `git status` is our key to the mind of Git. It will tell us what Git is thinking and the state of our repository as Git sees it. When you're first starting out, you should be using the `git status` command *all of the time*! Seriously. You should get into the habit of running it after any other command. This will help you learn how Git works and it'll help you from making (possibly) incorrect assumptions about the state of your files/repository.

---

QUESTION 1 OF 2

To answer this quiz, make sure you've `cd` ed into the `course-git-blog-project` project. If you've been following along in this lesson and haven't added any files to this project, then what does running `git status` display?

---

○   Status: good

---

○   On branch master
    Your branch is up-to-date with 'origin/master'.
    Initial commit
    nothing to commit (create/copy files and use "git add" to track)

---

○   master branch
    Please commit some files

---

⊘   On branch master
    Your branch is up-to-date with 'origin/master'.
    nothing to commit, working directory clean

---

[ SUBMIT ]

## Git Status Output

The `git status` command will display a lot of information depending on the state of your files, the working directory, and the repository. You don't need to worry too much about these, though...just run `git status` and it will display the information you need to know.
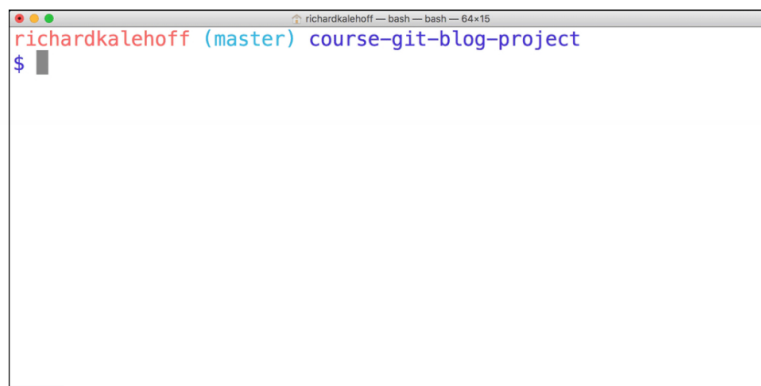
*An animated gif of the Terminal application. The `git status` command is run in the `course-git-blog-project` project.*

## Git Status Explanation

As you can see in the GIF above, running `git status` in the `course-git-blog-project` project produces the following output:

```
On branch master
Your branch is up-to-date with 'origin/master'.
nothing to commit, working directory clean
```

The output tells us two things:

1. `On branch master` – this tells us that Git is on the `master` branch. You've got a description of a branch on your terms sheet so this is the "master" branch (which is the default branch). We'll be looking more at branches in lesson 5
2. `Your branch is up-to-date with 'origin/master'.` – Because `git clone` was used to copy this repository from another computer, this is telling us if our project is in sync with the one we copied from. We won't be dealing with the project on the other computer, so this line can be ignored.
3. `nothing to commit, working directory clean` – this is saying that there are no pending changes.

Think of this output as the "resting state" (that's not an official description - it's how I like to describe it!). This is the resting state because there are no new files, no changes have been made in files, nothing is in the staging area about to be committed...no change or action is pending, so that's why I like to call it the resting state.

So this is what it looks like when running `git status` in a repository that already has commits. Let's switch to the `new-git-project` project to see what the `git status` output will produce.

💡 *Changes in Git v2.14* 💡

*In Git version 2.14, running the* `git status` *command in an empty directory changed the wording of "Inital commit" to the much clearer "No commits yet". So the output would be:*

```
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)
```
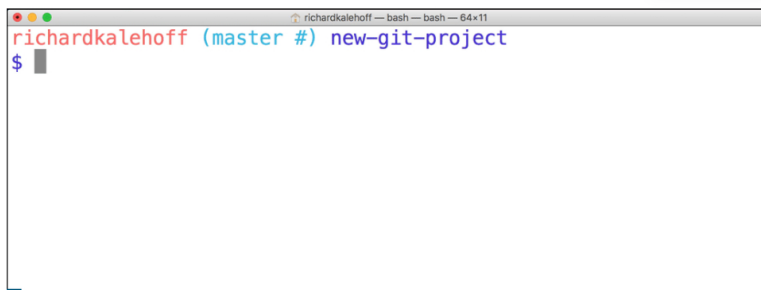
QUESTION 2 OF 2

To answer this quiz, make sure you've `cd` ed into the `new-git-project` project.

If you've been following along in this lesson and haven't added any files to this project, then what does running `git status` display?

○  Status: good

⊘  On branch master
   Initial commit
   nothing to commit (create/copy files and use "git add" to track)

○  master branch
   Please commit some files

○  On branch master
   nothing to commit, working directory clean

SUBMIT



```
richardkalehoff (master #) new-git-project
$ ▮
```

*An animated GIF of the Terminal application. The* `git status` *command is run in the* `new-git-project` *project.*

## Explanation Of Git Status In A New Repo

This is the output of running `git status` in the `new-git-project` project:

```
$ git status
On branch master

Initial commit

nothing to commit (create/copy files and use "git add" to track)
```

To be completely clear, I haven't made any commits in my project yet. If you have made a commit, then your output should look exactly like that of the course-git-blog-project project.

If you compare this to the `git status` output from the course-git-blog-project project, then you'll see that they're pretty similar. The thing to note that's different is that this output includes the line `Initial commit`. This is the tiniest bit confusing because there actually aren't any commits in this repository yet! We haven't discussed making a commit yet, but when we do, we will be able to make an initial commit.

Wanna have a sneak peak of the next lesson and at the same time prove that there aren't any commits in this repo yet? Great, I knew you did! Try running the command `git log` and check out its response:

```
$ git log
fatal: your current branch 'master' does not have any commits yet
```

Well, that's kind of scary looking. "Fatal"? Fortunately, it turns out that just means that the Git program is exiting because it can't find any work to do. Git tells us this as if it were an error, but it's really not a problem. We know we haven't put any commits into this repo yet.

It's pretty clear from the response that there aren't any commits!

We've just taken a very brief look at the `git status` command. Remember that the output of `git status` will change depending on if files have been added/deleted/modified, what's on the staging index, and the state of the repository. We'll be using the `git status` command throughout this entire course, so get comfortable running it!

### Git Status Recap

The `git status` command will display the current status of the repository.

```
$ git status
```

I can't stress enough how important it is to use this command *all the time* as you're first learning Git. This command will:

- tell us about new files that have been created in the Working Directory that Git hasn't started tracking, yet
- files that Git *is* tracking that have been modified
- a whole bunch of other things that we'll be learning about throughout the rest of the course ;-)

### Helpful Links

- Checking the Status of Your Files
- git status docs
- git status Tutorial

NEXT