"Learn how to github"

Prepared by: Masood Sadat

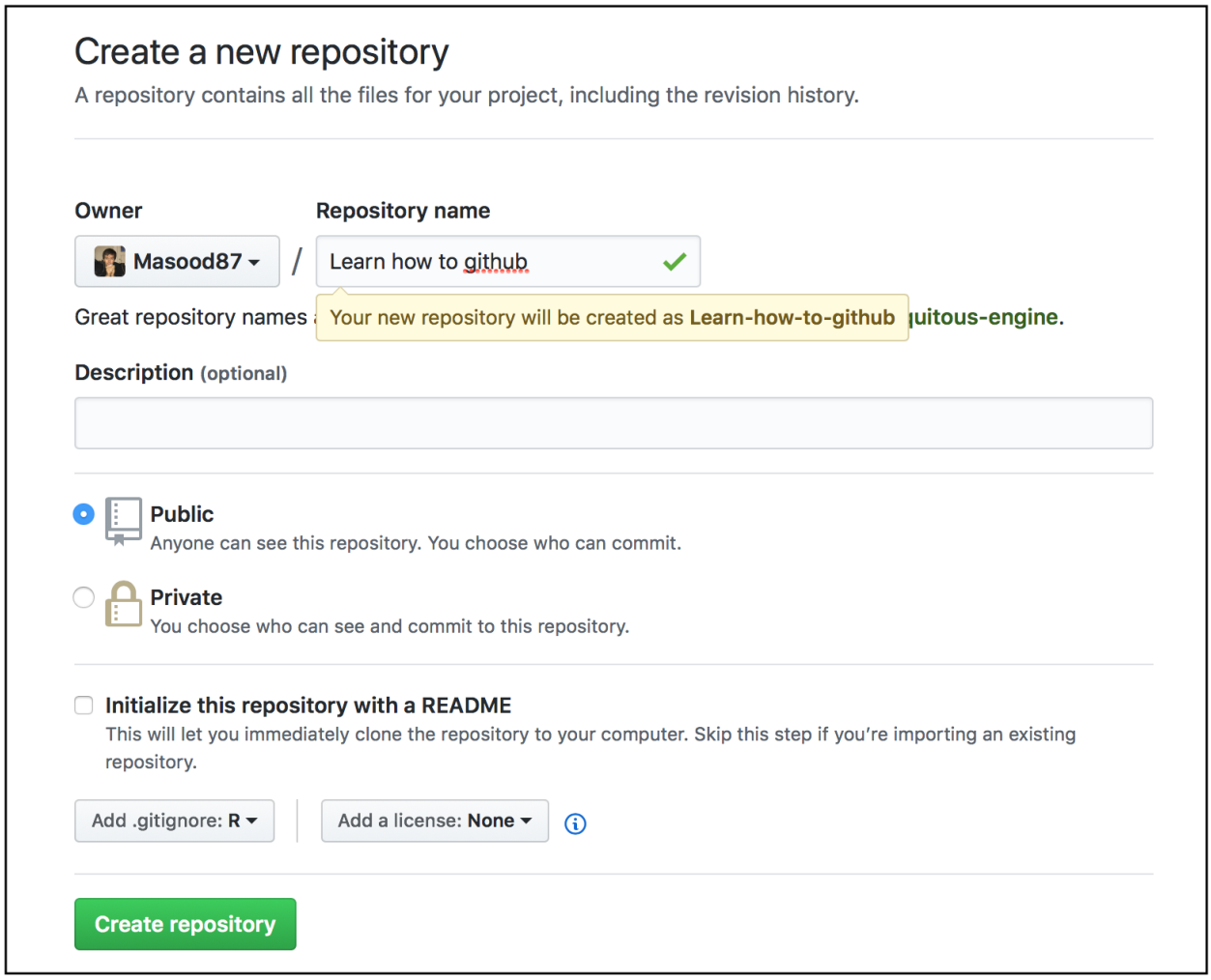
May 20, 2018

Contents

1 Create account and repository on github.com

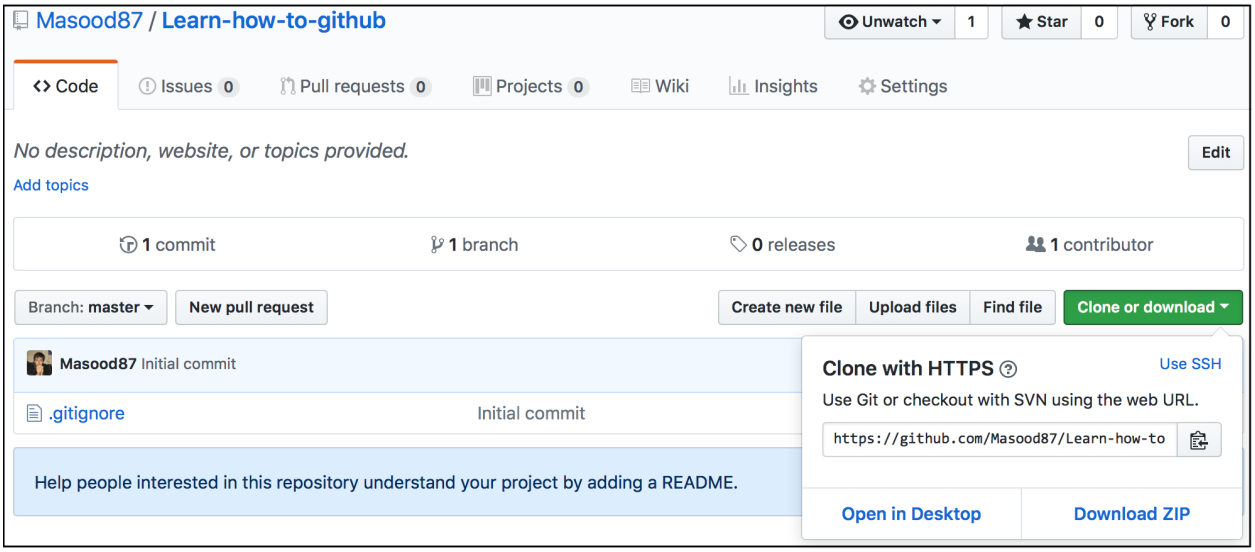
Go to `github.com` and create an account. Once, account is created, create a repository in `Repositories` tab and click on the green button `New`. The following page opens.

> ***NOTE:*** *A repository is something like a project folder.*



2 Copy repository URL

After a new repository is created, copy the `clone url` link. This link is used to clone the repository from server to the computer.



3 Set working directory

On Mac computers, open `Terminal` (the dreaded command line). I am not an expert on `terminal` and it is like a black box to me at this point. But for purpose of using `git`, it is not difficult and we need to know only a few commands.

NOTE: *To find `terminal`, search spotlight OR go to `Applications` > `Utilities` > `Terminal`.*

The following are important commands for working with directory:

- Check working directory: `pwd`
- Use `cd` to change working directory.
  - Go up one folder: `cd ..`
  - Go into a folder: `cd ~/documents/github`
- Check content of workding directory: `ls` (or `dir` for windows)
- Make a new folder: `mkdir [foldername]`

For our project, we go to our desired directory and create a new folder if necessary.

So, first thing in the `terminal`, we check the working directory

```
pwd
```

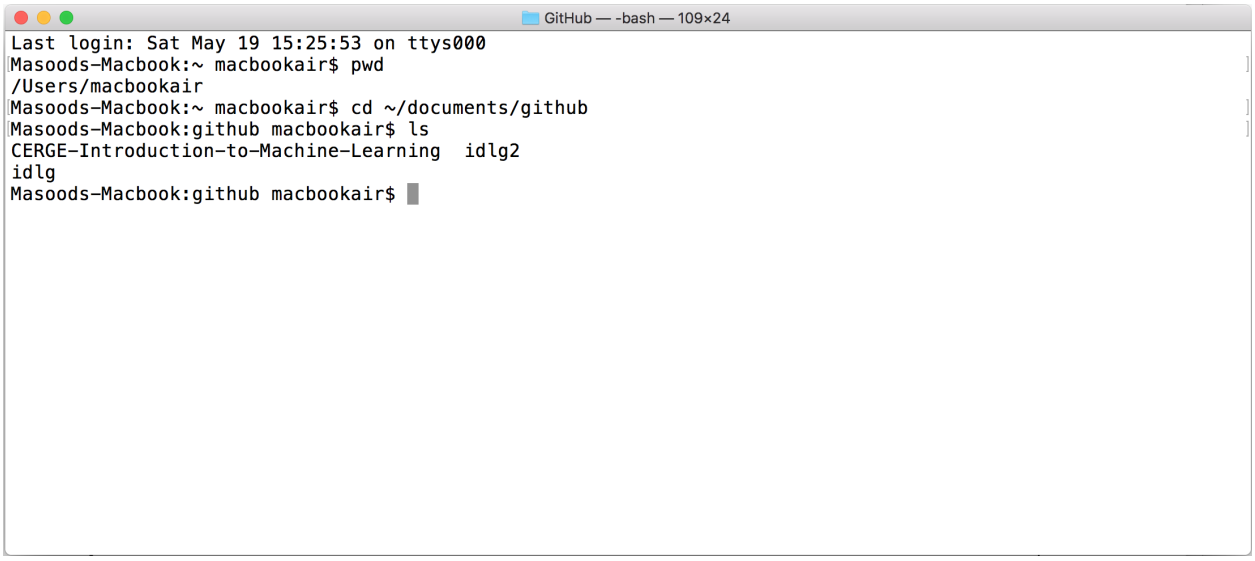
Then, we change the working directory to where we want. I have already a folder called `GitHub` in `Documents`, so I set the working directory there

```
cd ~/documents/github
```

We check the content of our working directory.

```
ls
```

Here is a screen shot of the process. As you can see, there are already three folders in the working directory.



4 Clone the repository to computer

In the `terminal`, clone the repository in the working directory (i.e. from step 3: `~/Documents/GitHub/`). To do that, we use `git clone` with the url in the terminal command line.

```
git clone https://github.com/Masood87/Learn-how-to-github.git
```

> ***NOTE:*** *`git clone` is the first of five git commands we learn here. `git clone` essentially clones the repository into your working directory (in our case it is `~/Documents/GitHub/`).*

Type `ls` in the `terminal` after `git clone` to see the cloned files.

```
ls
```

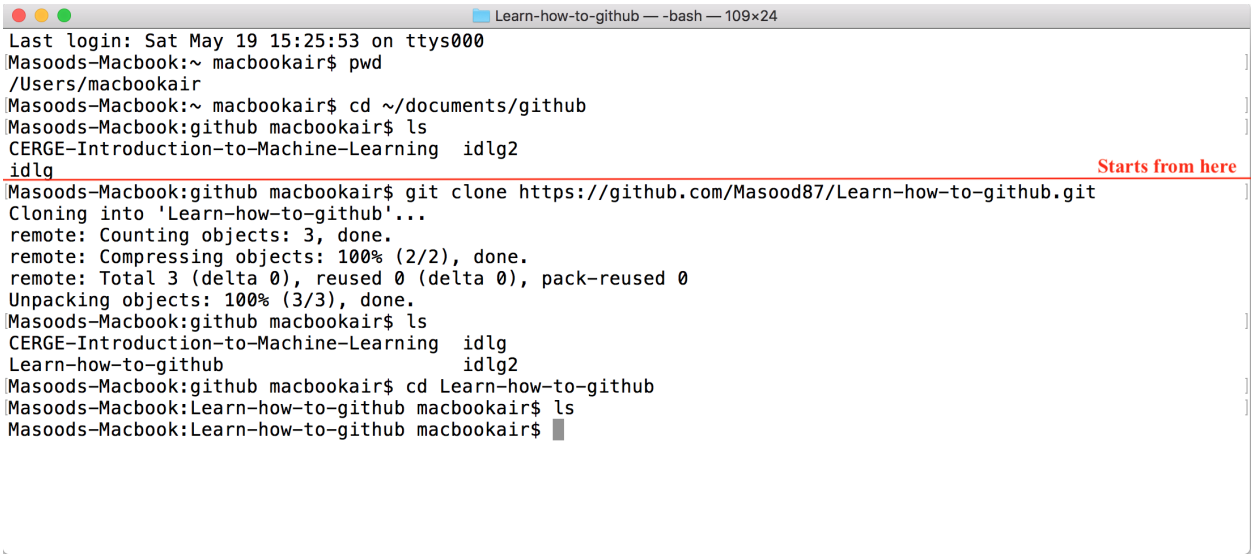
Next, change your working directory to the cloned folder `Learn-how-to-github`

```
cd Learn-how-to-github
```

You can check again the content of `Learn-how-to-github`

```
ls
```

Here is a screen shot of the process. As you can see, `Learn-how-to-github` is added and there is nothing inside it yet.



5 Make a change to repository and `git add`

After repository is cloned, we either add new files or modify an existing file. In this case, we add add a new folder with four files. In the finder, I add a folder called `screenshots` with four `.jpg` files in them.

In the terminal, we check the status of the cloned folder using `git status`.

> ***NOTE:*** *`git status` is used to check any changes including modification of codes inside a file (if any).*

```
git status
```

The changes will be noted in red font. We either *accept* to update the changes in the repository or *ignore* it. To *accept* and upload the changes, there are **three** steps:

- `git add` to ....
- `git commit` to ....
- and `git push` to ....

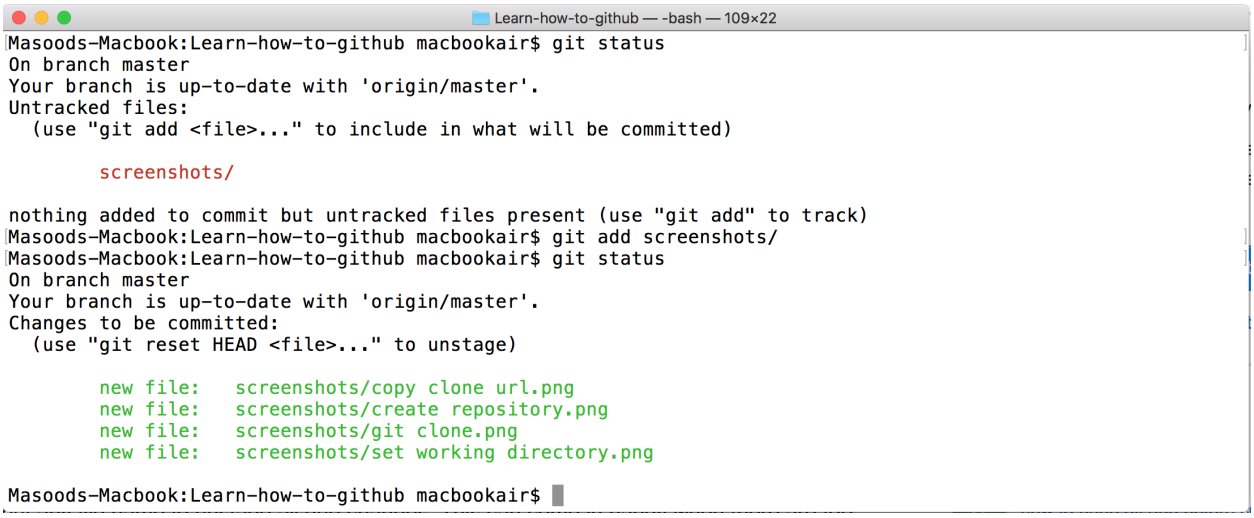
To `git add`, there are three ways in this case. The first one *add screenshots/* only, and the other two *add all*.

```
git add screenshots/

git add -A

git add .
```

Check the status again

```
git status
```

Here is a screen shot of the process. After `git add`, the status shows four files in green font. These files are changes to be *committed*.



6 Commit changes in repository with `git commit`

The next step after `git add` that adds the changes is to *lock* the changes. This *locking* does not mean applying the changes, which is done using `git push`.

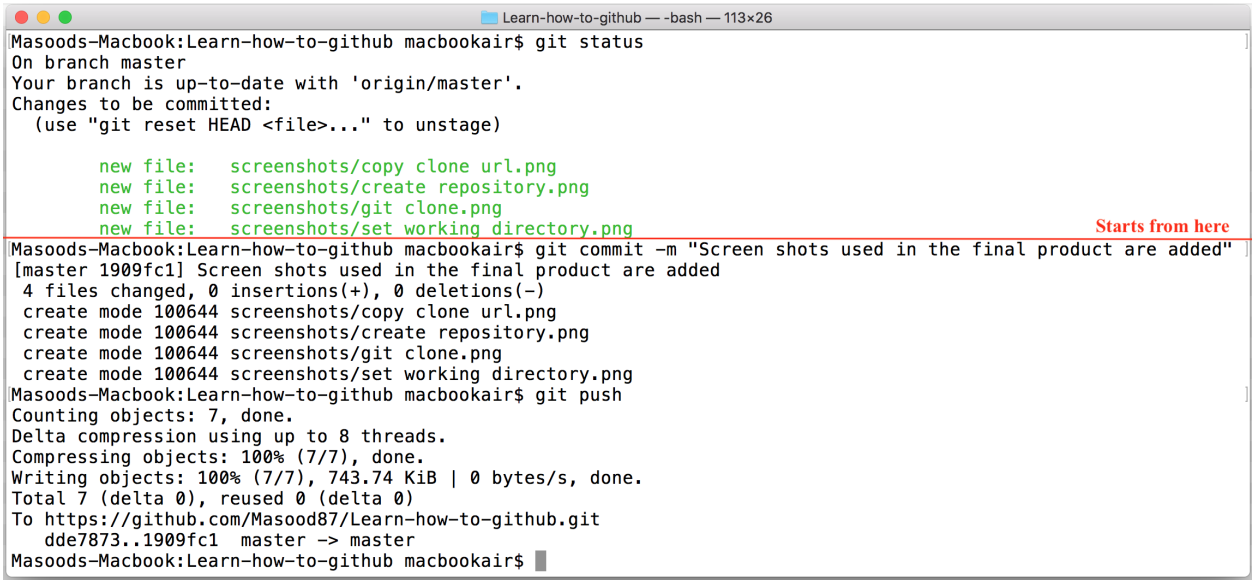
To commit changes, type `git commit` and `-m` with a message inside quotes.

```
git commit -m "Screen shots used in the final product are added"
```

At this stage, the change is committed but it is still on the computer and not synchronized with github.com repository. To sync, type `git push`

```
git push
```

Here is a screen shot of the process. After `git add`, there is `git commit` and finally `git push`. Now the files are synchronized with github.com



7 Workflow

So far, all we discussed was to set up a new repository / project, and commit changes. Other times, we have to `pull` changes others make in the project.

The first thing one does when starting the day to work on a collaborative project is to change directory to the github repository/project using `cd` and request a `pull` of all the changes.

```
cd ~/documents/github/learn-how-to-github
git pull
```

More details coming up...

8 Others

To see all commands we can use and what they do, just type `git`.