Git & GitHub

What is Git?

Git is a distributed version control system (dvcs) that allows developers to keep track of their code changes and collaborate with other developers.

It is by far the most popular version control system for open source projects. Created in 2005 by Linus Torvalds to manage the Linux kernel.

Git is like a time machine for your code. Each committed change to the codebase is tracked in a repository.

Distributed vs Centralized VCS

Unlike some version control systems, Git is distributed. Each “checkout” of a code repository (clone) on a local machine is a full copy.

A local clone can operate (commit, branch, etc...) 100% offline and only optionally push to a remote server.

Installing Git

Binary distributions available for all major platforms (Windows, Mac, Linux)

Available at Git website: http://git-scm.com/

Can also be installed using package managers such as brew (Mac)

```
brew install git
```

Getting started with Git

Tell Git who you are:

```
git config --global user.name "Tony Stark"

git config --global user.email tstartk@gmail.com
```

Create a repository:

From within the directory containing a project codebase, run:

```
git init
```

This will create a hidden “.git” subdirectory containing all internal objects for Git to function. You’ll likely never need to look at its content.

Adding and committing files:

```
git add myfile.txt
git commit -m “First commit”
```

Reviewing history

Each commit in Git is a snapshot that could include changes for multiple files.

To view a paginated list of commits in your repository:

```
git log
```

To include the list of files in each commit:

```
git log —name-status
```

Review the last 5 commits:

```
git log -n 5
```

Lots of options available for the `log` command, run `git help log` to view the full documentation

Cloning a remote repository

To make a local copy of an existing project:

```
git clone <url> <local_directory>

cd <local_directory>
```

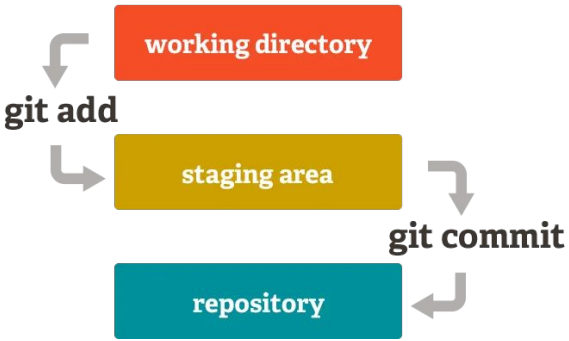
… and start making changes!

Reviewing changes

So you've made a bunch of changes and would like to review them before committing. Simply run:

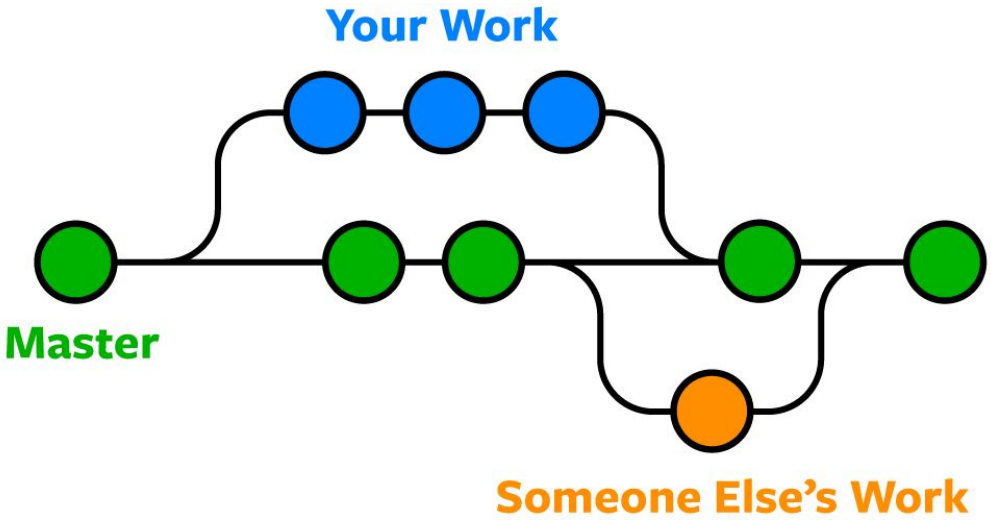
```
git diff
```

This will show changes between the working directory and staging area.



Git Branches

A Git branch is a separate path of development. It allows multiple features to be worked on without interfering with the main branch.



Git Branches (continued)

How to create a branch:

`git branch <branch_name>` (creates the branch without switching to it)

or

`git checkout -b <branch_name>`

GitHub

GitHub provides a web-based platform for developers to host and collaborate on Git repositories.

Repositories hosted on GitHub are just regular Git repositories.

GitHub enhances Git’s capabilities by providing management, collaboration and security tools.

GitHub Features

1. Pull Requests
2. Branch Protection Rules
3. Issues
4. Tags & Releases

Creating a repository

Under “Your Repositories” click “New”

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? Import a repository.

Required fields are marked with an asterisk (*).

Owner * **Repository name ***

laurenty-dev / ie-python-demo

ie-python-demo is available.

Great repository names are short and memorable. Need inspiration? How about **jubilant-dollop** ?

Description (optional)

A test repository

- **Public** — Anyone on the internet can see this repository. You choose who can commit.
- **Private** — You choose who can see and commit to this repository.

Initialize this repository with:

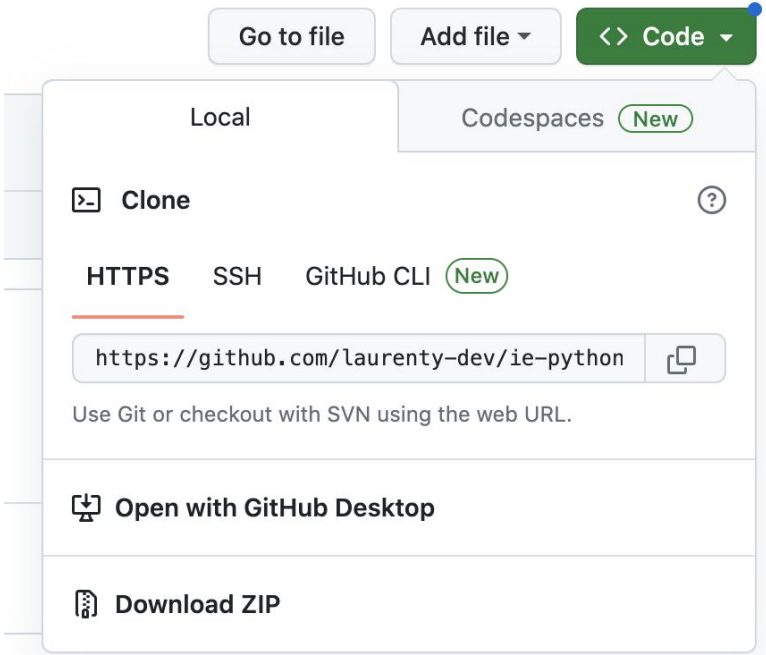
- **Add a README file** — This is where you can write a long description for your project. Learn more about READMEs.

Cloning a GitHub repository

Github offers several methods for cloning repositories:

- HTTPS
- SSH
- GitHub CLI

The end result is the same, it comes down to authentication preferences.



GitHub Pull Request

Demo time!

Questions?

Still awake?

Resources

https://medium.com/@lucasmaurer/git-gud-the-working-tree-staging-area-and-local-repo-a1f0f4822018

Official docs: https://git-scm.com/doc

https://www.atlassian.com/git/tutorials/learn-git-with-bitbucket-cloud (BitBucket)

Branches explained:
https://git-scm.com/book/en/v2/Git-Branching-Branches-in-a-Nutshell