💻 🎓 Git-it is a (Mac, Win, Linux) Desktop App for Learning Git and GitHub

🕐 **776** commits	⑂ **5** branches	📦 **0** packages	🏷 **14** releases	👥 **52** contributors	⚖ BSD-2-Clause

Branch: master ▾   New pull request   Create new file   Upload files   Find file   🔲 🔲 🔲   Clone or download ▾

👤 **jlord** Merge pull request **#257** from jotoeri/add-guilink   ...   ✔ Latest commit `d808d9c` on Mar 13, 2019

📁 assets	make lang button wider	2 years ago
📁 lib	Set es_ES for es alias	17 months ago
📁 menus	Add exrta space #220	2 years ago
📁 resources	Adding a link to the GUI overview	11 months ago
📁 tests	test notes	5 years ago
📄 .gitattributes	make PortableGit to be ignored in language stats	3 years ago
📄 .gitignore	Update .gitignore	4 years ago
📄 .travis.yml	update travis configuration to cache and use a later version of node	3 years ago
📄 CONTRIBUTING.md	Massive, messy merge. Fingers crossed.	3 years ago
📄 LICENSE	Initial commit	5 years ago
📄 README.md	Update README.md	15 months ago
📄 docs.md	assests -> assets	3 years ago
📄 empty-data.json	Add Pick Up Where You Left Off button	4 years ago
📄 empty-saved-dir.json	Created saved dir empty file	4 years ago
📄 main.js	Add locale support in main.js	4 years ago
📄 package-lock.json	Add es_ES to README	17 months ago
📄 package.json	version 4.3.0	2 years ago

📖 **README.md**

Git-it (Desktop App)

`build` `passing` `code style` `standard`
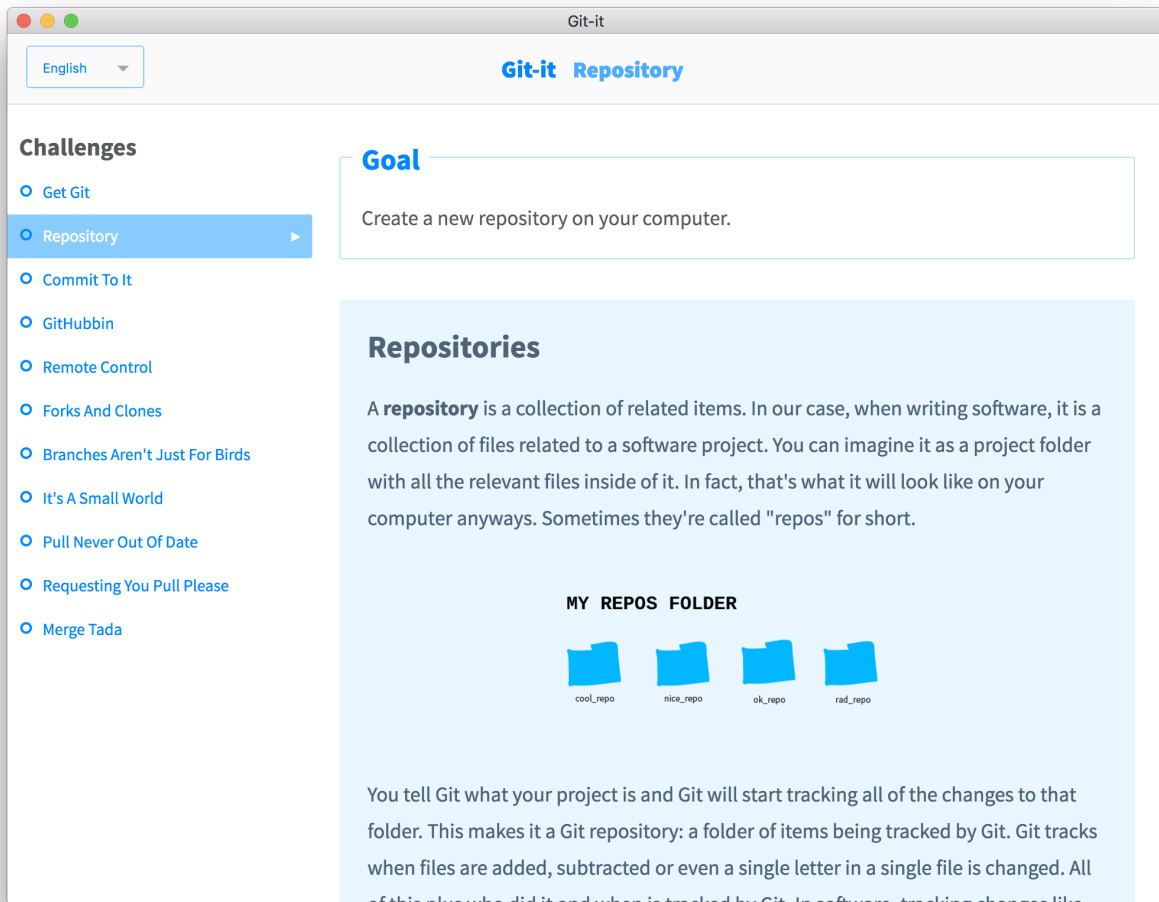
Git-it is a desktop (Mac, Windows and Linux) app that teaches you how to use Git and GitHub on the command line.

⚐ **The app includes translations in these languages: Spanish (ES), Spanish (CO), French, Portugese (BR), Ukrainian, Traditional Chinese, Japanese, and Korean.** ⚐

Previously it ran *in* terminal (the command line application) with a guide online but now the guide *is* the app and it runs on your desktop on its own 🛩 **This will be the only maintained version going forward** →.

Hello future Forkers, Branchers and Pull Requesters!

This application contains challenges for learning Git and GitHub—by using *real* Git and GitHub, not emulators. You'll be learning the awesome (and not so scary) command line and GitHub which means when you finish all of the challenges you'll have *real* repositories on your GitHub account and green squares on your contribution chart.

Your Contributions

	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Jan

M

W

F

Summary of Pull Requests, issues opened and commits. Learn more.

Less ▢ ▢ ▢ ■ More

1,567 Total	**25 days**	**25 days**
Jan 21 2013 - Jan 21 2014	December 28 - January 21	December 28 - January 21
Year of Contributions	**Longest Streak**	**Current Streak**
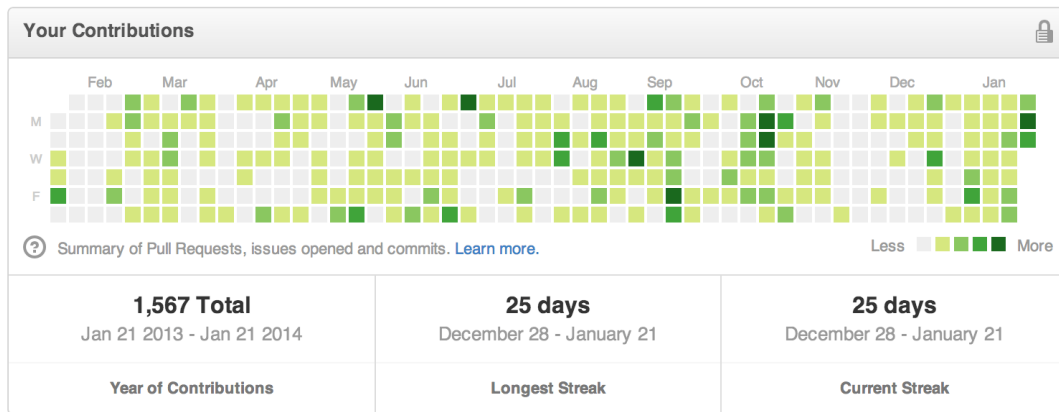
What to Install

You will need this app, Git-it as well as a text editor. You'll also of course need Git and the first challenge in Git-it helps you get that set up. But if you want to get a head start, go for it! You'll continue to use Git and your text editor throughout your bright social coding future.

As a part of the challenges you'll also create a (free) account on GitHub. If you've already got one, high-five!

Git

We recommend installing GitHub Desktop (free) because it installs Git in the most consistent way across supported operating systems (Windows and Mac). We won't use the app itself in the challenges (but it is really useful once you get going in Git!), just the Git that it install for you.

Note—If you're using Windows, you should use the **Git Shell** app as your terminal—it is installed with GitHub Desktop. In Mac and Linux you can use the app **Terminal**, which is already on your computer.

Text Editor

I'm quite partial (I'm on the team!) but Atom is a great text editor (also free) and it is built on Electron.js, just like Git-it. Visit the website and download the version for your operating system. Sublime, Microsoft VS Code and Adobe Brackets are other options.

Git-it

You can view the releases section of this repository to find the download for your operating system. Click to download your version. Once it has downloaded to your default Downloads directory, unzip the folder and run the Git-it executable. The executables for Mac, Windows and Linux:
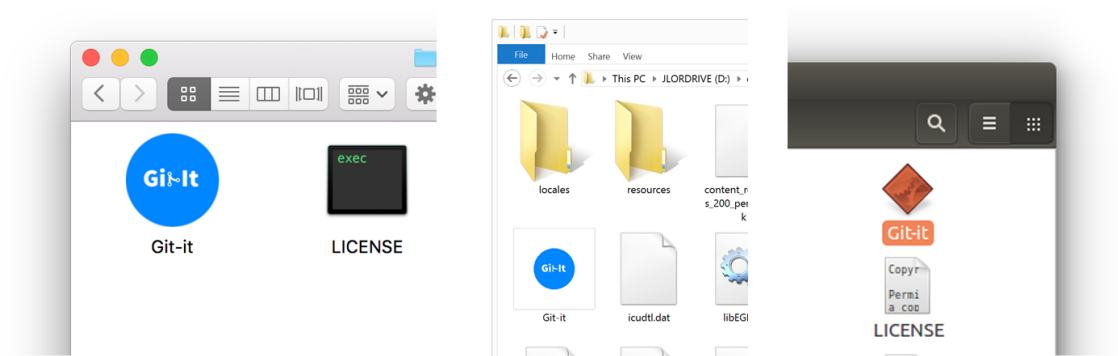
- **Mac** Right (control) click the Git-it icon, select Open and then Open again. You can drag the icon into your Applications directory if you want, but it's not required. If needed, more detailed installation instructions can be found in this issue.
- **Windows** Double-click the executable.

- **Linux** Double-click the executable. If the app isn't running, see this issue.

Get Started!

Open Git-it and click the button to begin the first challenge. Have your terminal and text editor open, too. Follow along with the instructions in each challenge and use the terminal or editor as instructed.

When you've completed the steps in a challenge click 'Verify'. Depending on the challenge you may need to also select the folder you did your work in for Git-it to verify.

Questions? Open an issue on this repository.

Want to contribute or build locally? See the contributing documentation

Tips For Getting Started

Code snippets often times look like `$ some code-stuff --here`. The dollar sign identifies the line as one a user would enter into the command line, but you don't actually include it when you type it into terminal. In this case, you'd actually just type `some code-stuff --here`.

Variables are indicated by `<VARIABLENAME>` in code snippets. When you actually use the line of code, replace `<VARIABLENAME>`, with your variable. For instance to make a new folder in terminal the format is, `mkdir <FOLDERNAME>`, so if you wanted to make a folder named 'octocat', you'd type: `mkdir octocat`.

Command line, terminal and bash all basically mean the same thing: the MS-DOS, Doogie Howser looking screen full of words and numbers. It's awesomely powerful and allows you to control your computer with text commands.

You can do a lot of things from your terminal like delete, rename, copy or create files and folders; run scripts and send things back and forth between servers (like the ones storing things on GitHub.com) and your computer (also a server!).