

Web Design & Development Summer Academy 2018

Getting Started with Git and GitHub



Wi-Fi



- **Wi-Fi Network:** Clark_Student
- If prompted for a code enter “blue sky”
- **Username:** g_webdesign
- **Password:** Summer2018

Schedule of Saturday Events

Sign Up Page: <https://goo.gl/HCKrrj>

~~July 14, 2018 – Getting Started with Markdown~~

~~July 28, 2018 – Getting Started with Sass~~

~~August 11, 2018 – Getting Started with Node.js~~

~~August 18, 2018 – Getting Started with Git and GitHub~~

August 25, 2018 – Getting Started with Bootstrap

September 8, 2018 – Getting Started with React.js

September 15, 2018 – Automating Things with Python

GitHub - What is it?

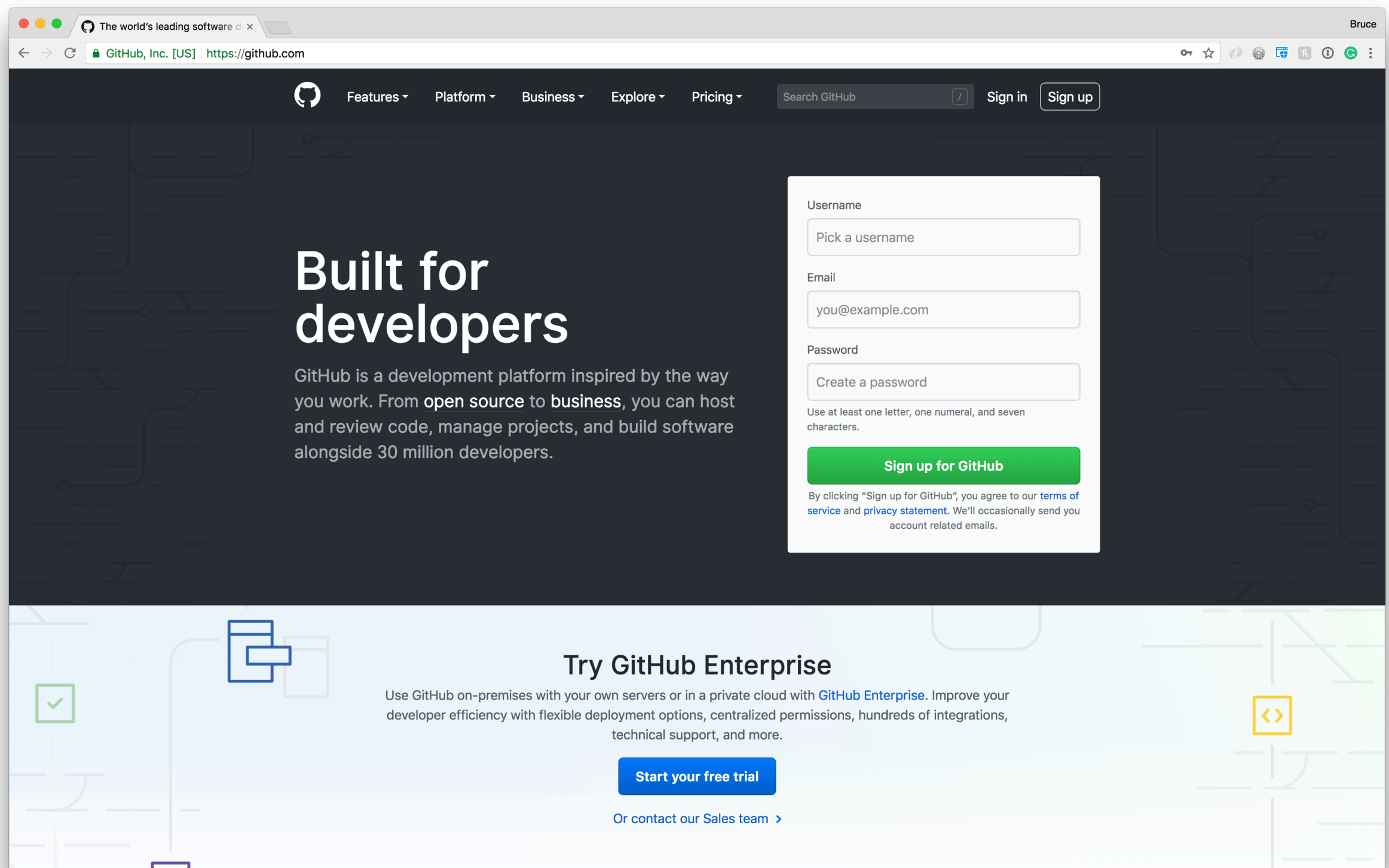
- Web based hosting platform for git repositories or "repos"
- Github is built on top of git
- It allows you to have local and remote copies of your projects
- It's a cloud based git service and way more than that!

GitHub - What is it?

- Web based hosting platform for git repositories or "repos"
- Github is built on top of git
- It allows you to have local and remote copies of your projects
- It's a cloud based git service and way more than that!

GitHub - What is it?

- Web based hosting platform for git repositories or "repos"
- Github is built on top of git
- It allows you to have local and remote copies of your projects
- It's a cloud based git service and way more than that!



<https://github.com>

Setting up a personal account

- Username
- Email
- The email address that you use matters
- Free accounts do not allow you to have "private" repositories
- Once you login notice all of the various plans available

Setting up a an account

- Username
- Email
- The email address that you use matters
- Free accounts do not allow you to have "private" repositories
- Once you login notice all of the various plans available

Hello World · GitHub Guides

Bruce


Secure

https://guides.github.com/activities/hello-world/

☆

GitHub Guides

Video GuidesGitHub HelpGitHub.com



Hello World

🕒 10 minute read

The **Hello World** project is a time-honored tradition in computer programming. It is a simple exercise that gets you started when learning something new. Let's get started with GitHub!

You'll learn how to:

- Create and use a repository
- Start and manage a new branch
- Make changes to a file and push them to GitHub as commits
- Open and merge a pull request

What is GitHub?

GitHub is a code hosting platform for version control and collaboration. It lets you and others work together on projects from anywhere.

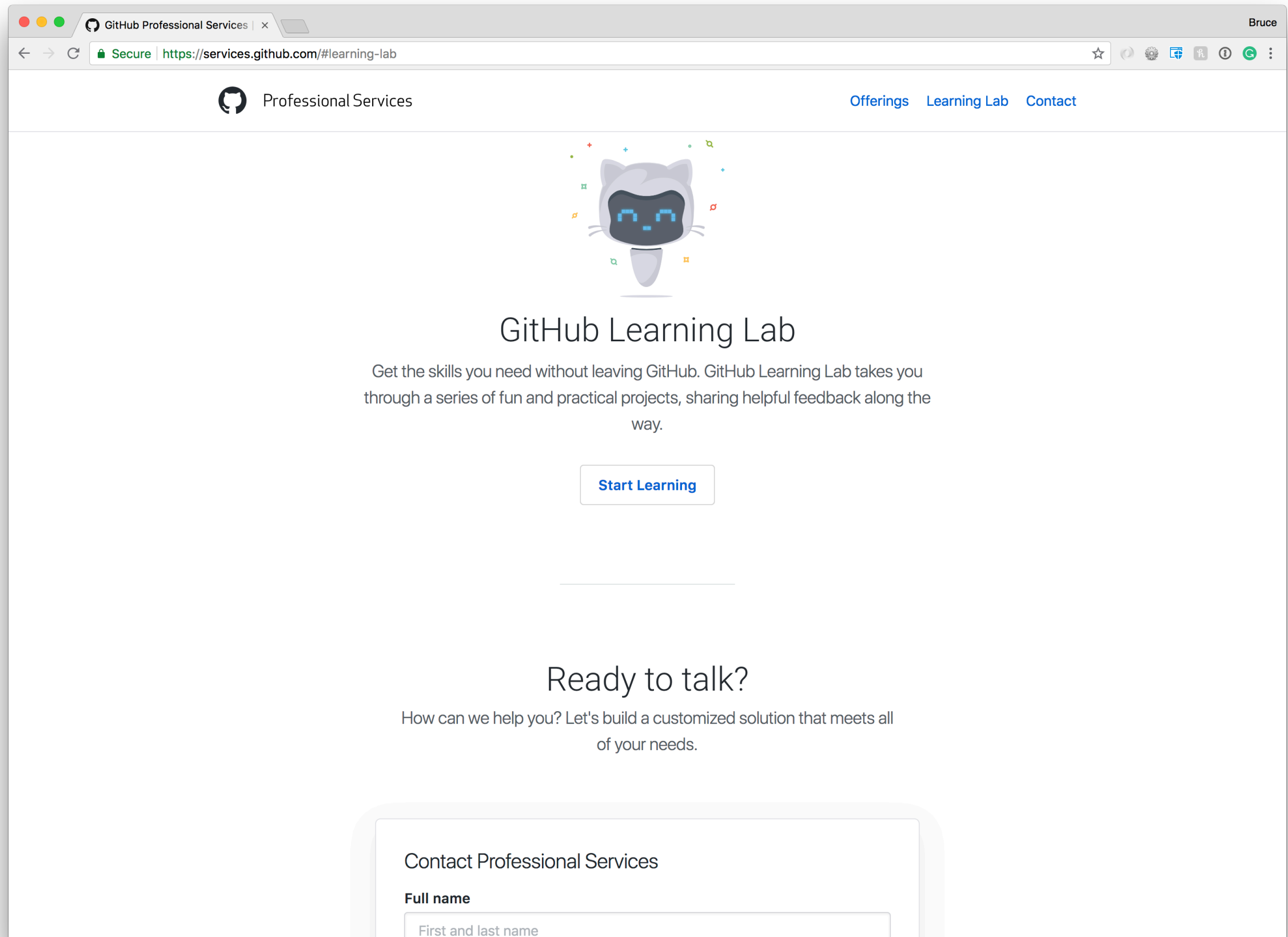
This tutorial teaches you GitHub essentials like *repositories*, *branches*, *commits*, and *Pull Requests*. You'll create your own Hello World repository and learn GitHub's Pull Request workflow, a popular way to create and review code.

No coding necessary

Intro

- [What is GitHub?](#)
- [Create a Repository](#)
- [Create a Branch](#)
- [Make a Commit](#)
- [Open a Pull Request](#)
- [Merge Pull Request](#)

<https://guides.github.com/activities/hello-world/>



<https://services.github.com/#learning-lab>


Git

- The core of what runs Github
- <http://git-scm.com>
- A distributed version control system

Git


https://git-scm.com


Bruce

**git** --distributed-is-the-new-centralized


Git is a **free and open source** distributed version control system designed to handle everything from small to very large projects with speed and efficiency.

Git is **easy to learn** and has a **tiny footprint with lightning fast performance**. It outclasses SCM tools like Subversion, CVS, Perforce, and ClearCase with features like **cheap local branching**, convenient **staging areas**, and **multiple workflows**.


 **Learn Git in your browser for free with Try Git.**

**About**


The advantages of Git compared to other source control systems.

**Documentation**


Command reference pages, Pro Git book content, videos and other material.


**Downloads**


GUI clients and binary releases for all major platforms.


**Community**


Get involved! Bug reporting, mailing list, chat, development and more.


**Pro Git** by Scott Chacon and Ben Straub is available to [read online for free](#). Dead tree versions are available on [Amazon.com](#).

 **Mac GUIs**

 **Tarballs**

 **Windows Build**

 **Source Code**



Latest source Release
2.18.0
[Release Notes \(2018-06-21\)](#)
[Download 2.18.0 for Mac](#)

Companies & Projects Using Git

GooglefacebookMicrosofttwitterLinked inNETFLIX

androidpenguinRAILSQtGNOMEeclipseKX

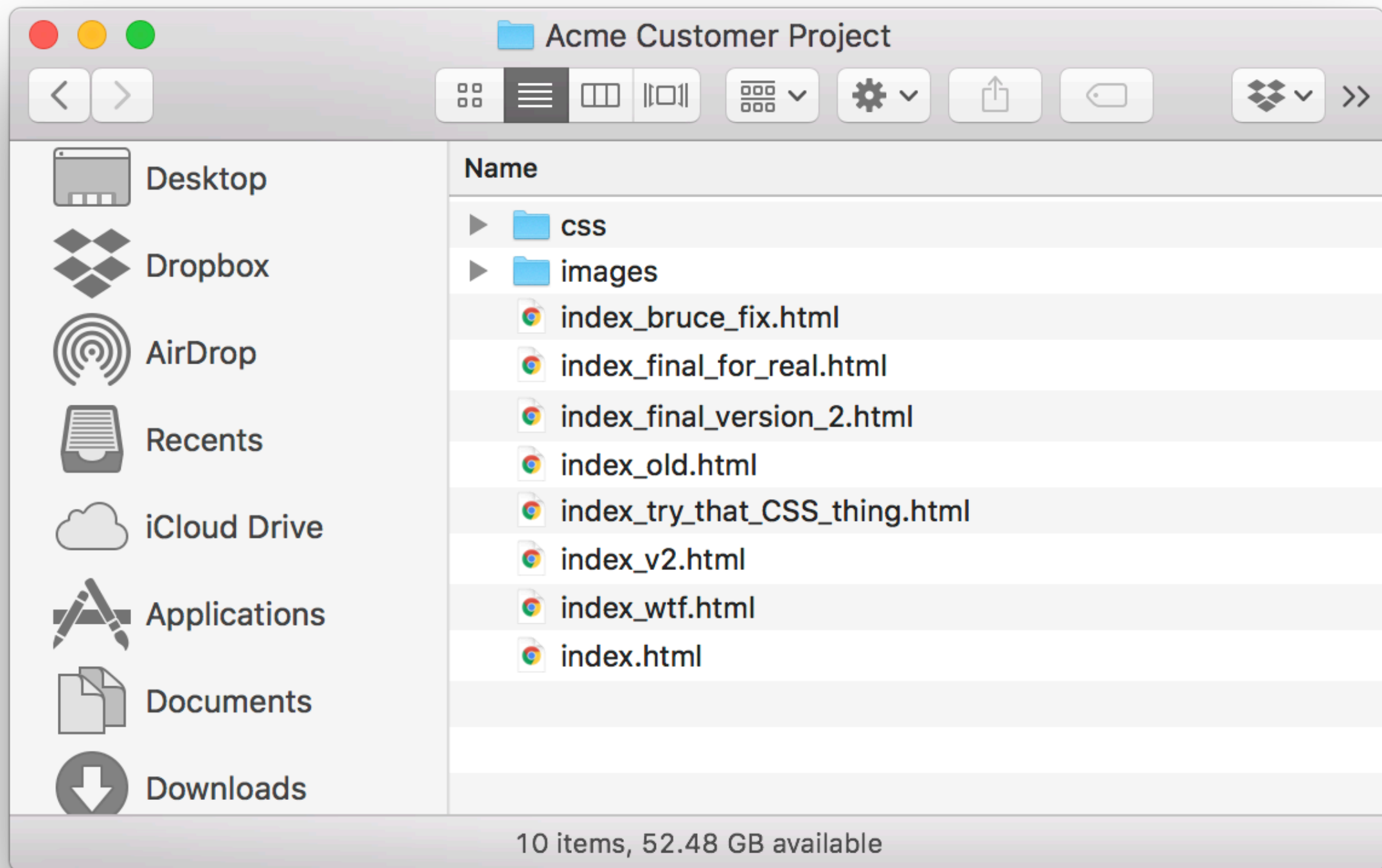
PostgreSQL

Git

- The core of what runs GitHub
- <http://git-scm.com>
- A distributed version control system

The command line is scary
The command line is scary
The command line is scary
The command line is scary
The command line is scary
The command line is scary
The command line is scary
The command line is scary
The command line is scary
The command line is scary
The command line is scary





Git

- Git tracks changes to projects over time
- Allows you to go back to a prior version of your code base
- Ever lose any work?
- I have

Two Types of Version Control Software

Centralized

- Centralized has a central repo where users can check out files and then check them back in when they are finished
- Subversion
- CVS

Distributed

- Git is distributed
- Everyone has a copy of the repo
- You can clone repos if one gets lost or damaged

The command line on OS X

pwd = Present Working Directory

mkdir = make directory

rm -r folder_name = delete contents of a folder

cd ~ = go back to home folder

ls = list files and folders

The command line on Windows

cd = Current Directory

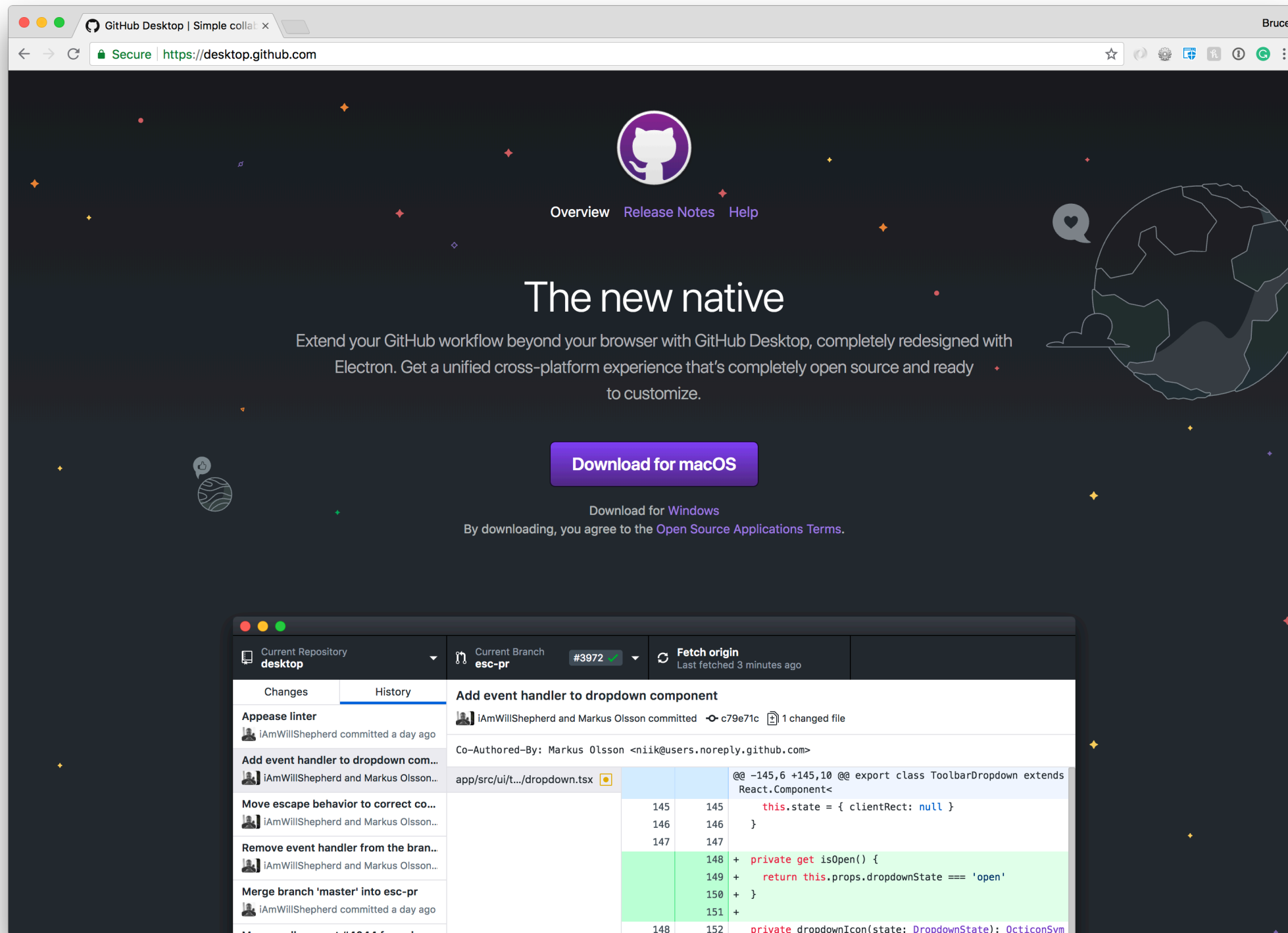
mkdir = make directory

rmdir folder_name = delete contents of a folder

cd .. = go back a folder in the folder hierarchy

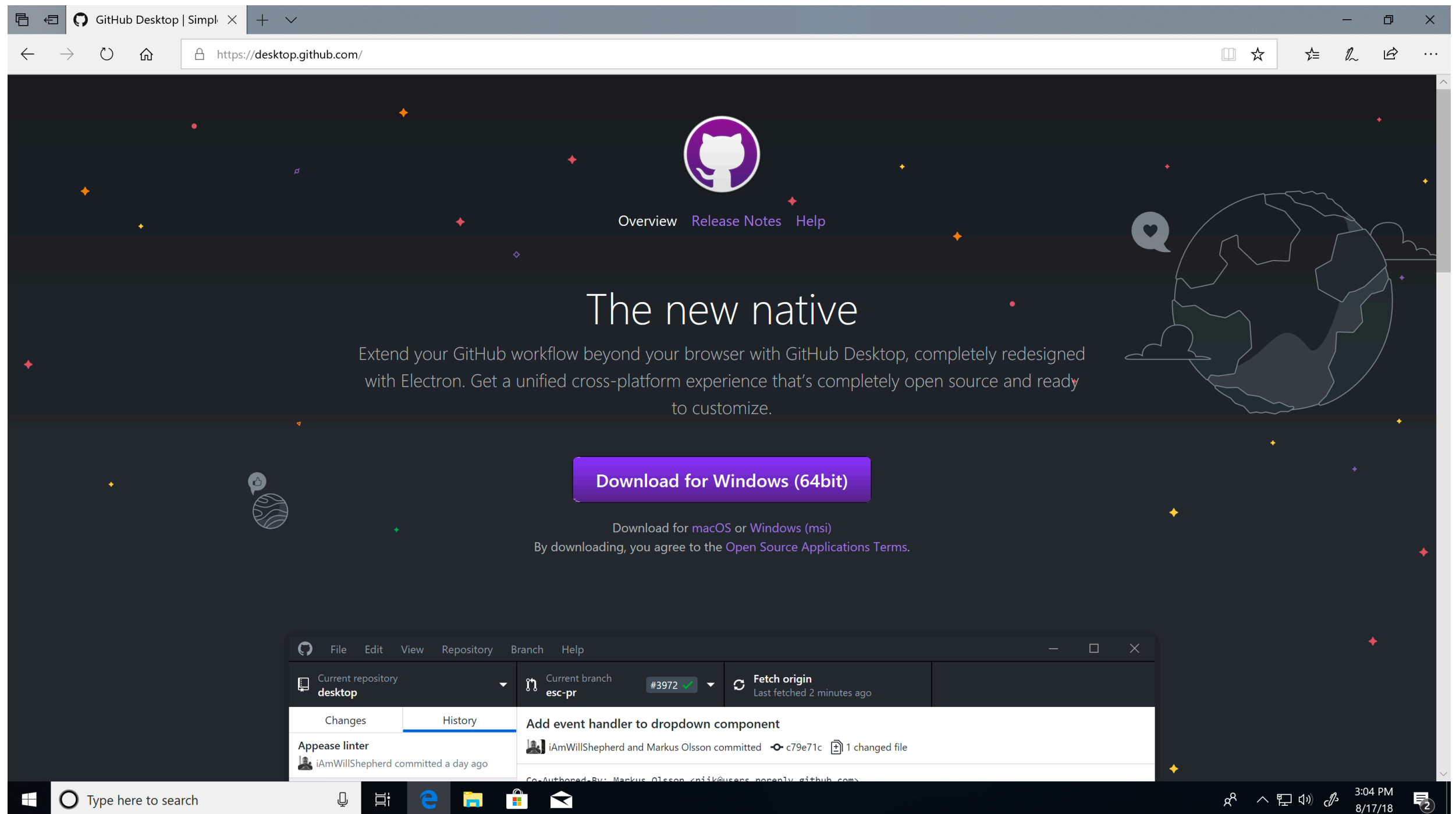
dir = list files and folders

Installing Git and GitHub client on OS X



<https://desktop.github.com/>

Installing Git and GitHub client on Windows



<https://desktop.github.com/>

Installing Git and GitHub client on Windows

- You will need to install git manually on Windows
- <http://git-scm.com>



Current Repository
No Repositories



No Repositories Found



Create a new project and publish it to GitHub

Create New Repository



Add an existing project on your computer and
publish it to GitHub

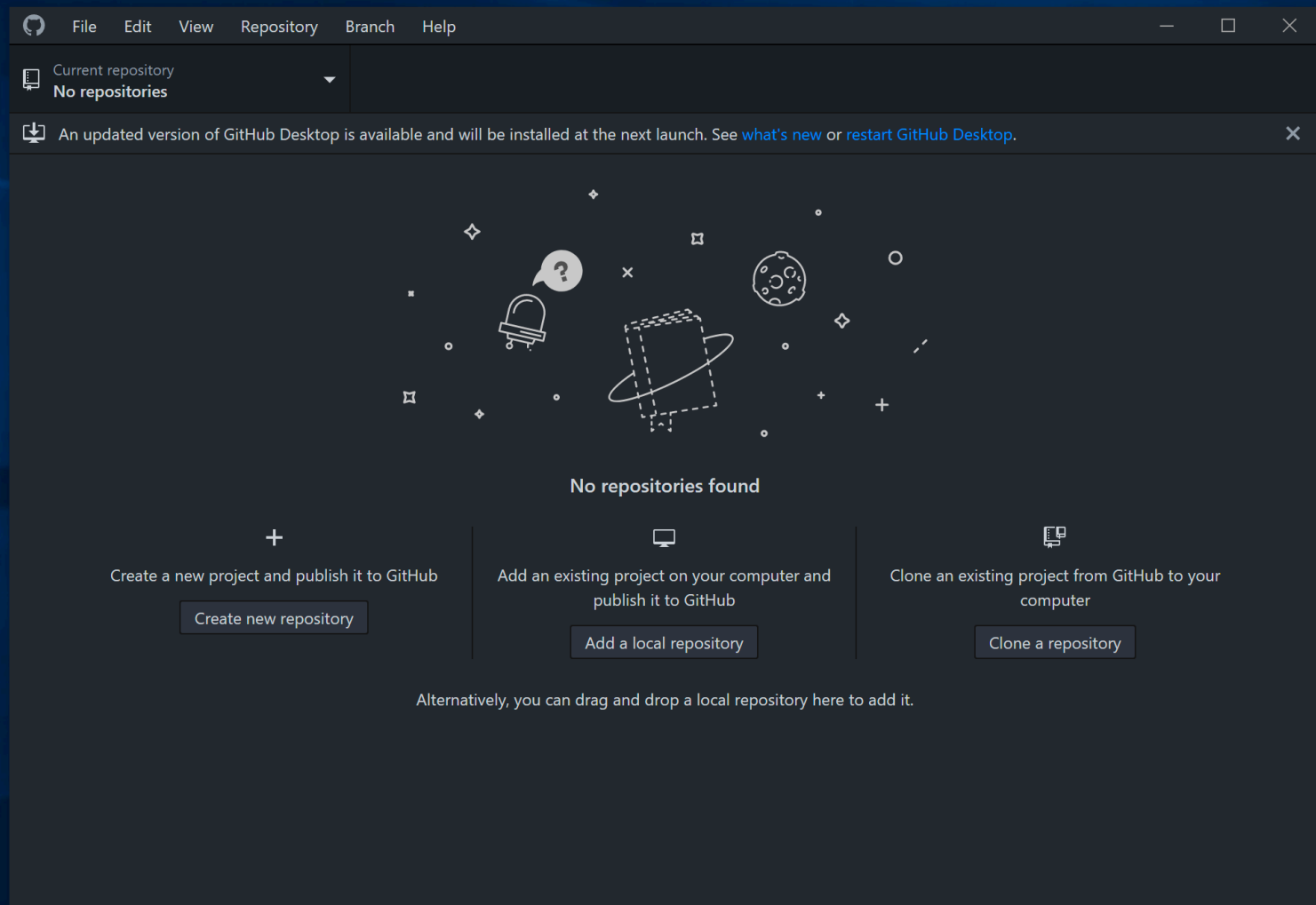
Add a Local Repository



Clone an existing project from GitHub to your
computer

Clone a Repository

Alternatively, you can drag and drop a local repository here to add it.



Type here to search



3:10 PM
8/17/18



How git works

- Git makes snapshots
- Each snapshot is called a “commit”
- Newer commits point back to older commits
- Newer commits only contain the files that have changed
- Each commit uses a 40 character string (hash) to represent each commit

Setting up Global Repository Defaults

Go to a command prompt

- `git config --global user.name "Bruce Elgort"`
- `git config --global user.email "belgort@clark.edu"`
- `git config --global credential.helper osxkeychain`
- `git config --list`
- More info at <https://help.github.com/articles/set-up-git/>

Create a git repo by...

Type in the terminal or command prompt in the folder you want to the repo to be located:

git init

Create a git repo by...


- Working directory contains current working files
- The staging area contains everything you want to be part of your next commit. Added or modified files.
- They go into the git repo once committed

Files exist in one of three stages

- Unmodified
- Modified
- Staged
- You need to manually move a file to the staging area. More on this later...

Creating a repo

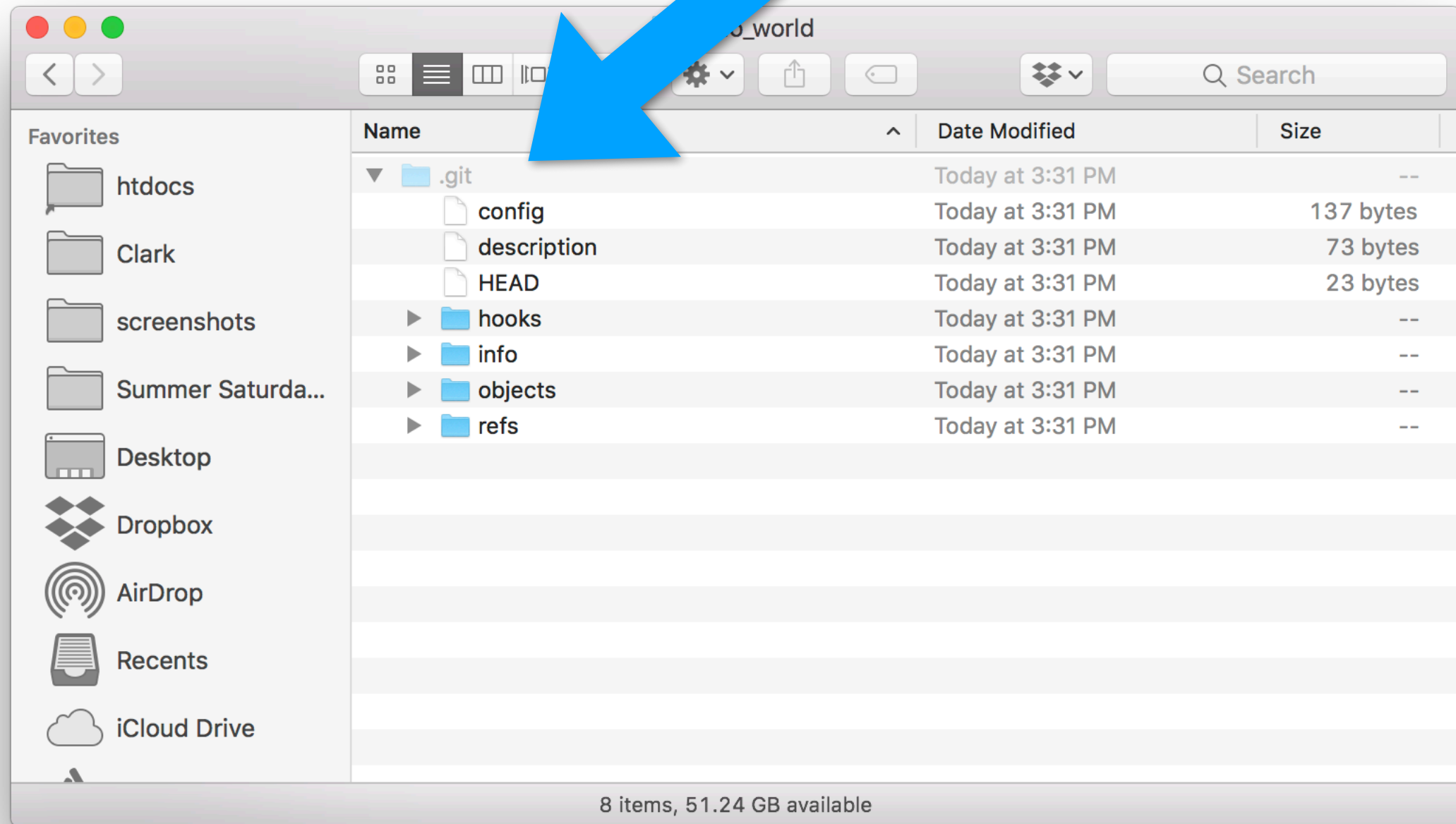
- git init

A terminal window titled 'hello_world — -bash — 81x23' is shown. It displays the command 'git init' being executed in a directory named 'hello_world'. The output indicates that an empty Git repository has been initialized in the path '/Users/bruceelgort/Desktop/hello_world/.git/'.

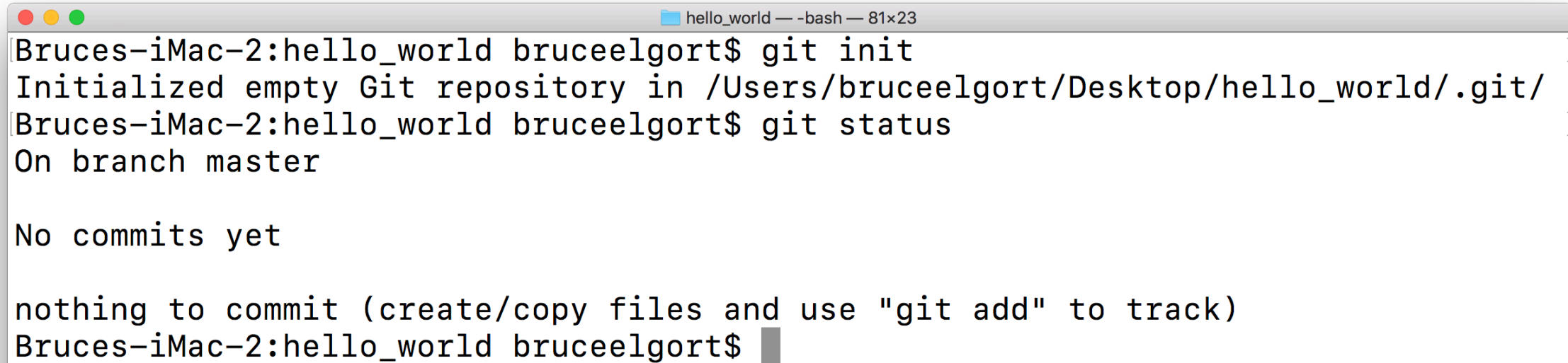
```
Bruces-iMac-2:hello_world bruceelgort$ git init
Initialized empty Git repository in /Users/bruceelgort/Desktop/hello_world/.git/
Bruces-iMac-2:hello_world bruceelgort$
```

Creating a repo

DO NOT MESS WITH THIS FOLDER!



Check the status of a repo

A terminal window titled 'hello_world — -bash — 81x23' with standard macOS window controls (red, yellow, green buttons). The terminal shows the execution of 'git init' and 'git status' commands. The output of 'git init' is 'Initialized empty Git repository in /Users/bruceelgort/Desktop/hello_world/.git/'. The output of 'git status' is 'On branch master' followed by a blank line and 'No commits yet'. A final line of output says 'nothing to commit (create/copy files and use "git add" to track)'. The prompt 'Bruce-iMac-2:hello_world bruceelgort\$' is followed by a cursor.

```
Bruces-iMac-2:hello_world bruceelgort$ git init
Initialized empty Git repository in /Users/bruceelgort/Desktop/hello_world/.git/
Bruce-iMac-2:hello_world bruceelgort$ git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)
Bruce-iMac-2:hello_world bruceelgort$
```

Add a file to the staging area

```
hello_world — -bash — 81x23
[Bruces-iMac-2:hello_world bruceelgort$ git init
Initialized empty Git repository in /Users/bruceelgort/Desktop/hello_world/.git/
[Bruces-iMac-2:hello_world bruceelgort$ git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)
[Bruces-iMac-2:hello_world bruceelgort$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)

    index.html

nothing added to commit but untracked files present (use "git add" to track)
[Bruces-iMac-2:hello_world bruceelgort$
```

Add a file to the staging area

```
hello_world — -bash — 81x23
[Bruces-iMac-2:hello_world bruceelgort$ git init
Initialized empty Git repository in /Users/bruceelgort/Desktop/hello_world/.git/
[Bruces-iMac-2:hello_world bruceelgort$ git status
On branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)
[Bruces-iMac-2:hello_world bruceelgort$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)

    index.html

nothing added to commit but untracked files present (use "git add" to track)
[Bruces-iMac-2:hello_world bruceelgort$ git add index.html
[Bruces-iMac-2:hello_world bruceelgort$
```

Add a file to the staging area

```
hello_world — -bash — 81x23
[Bruces-iMac-2:hello_world bruceelgort$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)

    index.html

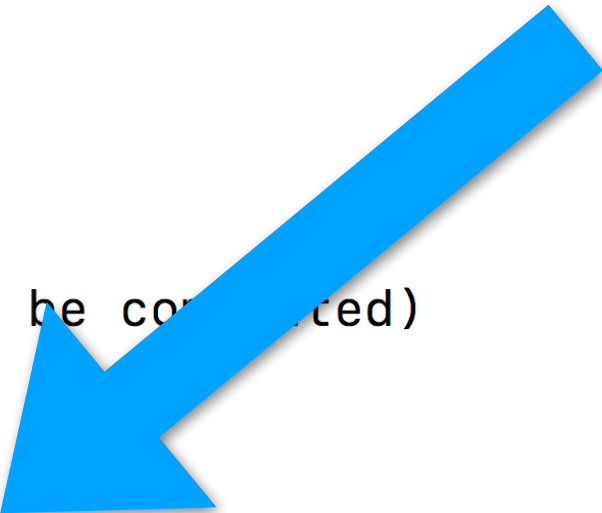
nothing added to commit but untracked files present (use "git add" to track)
[Bruces-iMac-2:hello_world bruceelgort$ git add index.html
[Bruces-iMac-2:hello_world bruceelgort$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

    new file:   index.html

Bruces-iMac-2:hello_world bruceelgort$
```



Add a file to the staging area

```
hello_world — -bash — 81x23
[Bruces-iMac-2:hello_world bruceelgort$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)

        index.html


nothing added to commit but untracked files present (use "git add" to track)
[Bruces-iMac-2:hello_world bruceelgort$ git add index.html
[Bruces-iMac-2:hello_world bruceelgort$ git status
On branch master

No commits yet


Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

        new file:   index.html

Bruces-iMac-2:hello_world bruceelgort$
```




Ready to commit



A terminal window titled "hello_world — -bash — 81x23" showing the execution of a git commit command. The output indicates a successful initial commit with the hash 5d0c6a0, creating a new file index.html with 10 insertions. The prompt returns to the user's shell.

```
Bruces-iMac-2:hello_world bruceelgort$ git commit -m "Initial commit"
[master (root-commit) 5d0c6a0] Initial commit
 1 file changed, 10 insertions(+)
 create mode 100644 index.html
Bruces-iMac-2:hello_world bruceelgort$
```

Ready to commit



```
hello_world — -bash — 81x23
[Bruces-iMac-2:hello_world bruceelgort$ git commit -m "Initial commit"
[master (root-commit) 5d0c6a0] Initial commit
 1 file changed, 10 insertions(+)
 create mode 100644 index.html
[Bruces-iMac-2:hello_world bruceelgort$ git status
On branch master
nothing to commit, working tree clean
Bruces-iMac-2:hello_world bruceelgort$
```

git log



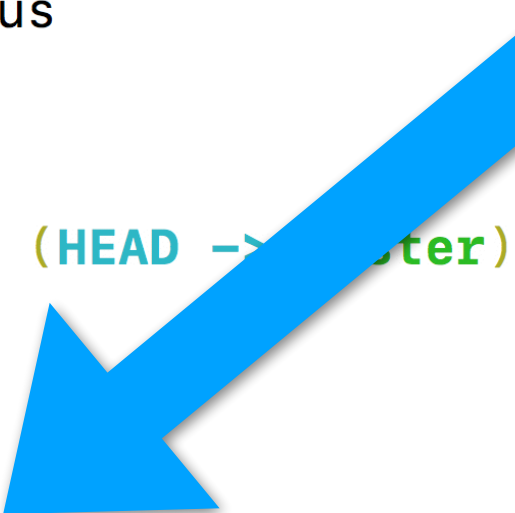
```
hello_world — -bash — 81x23
[Bruces-iMac-2:hello_world bruceelgort$ git commit -m "Initial commit"
[master (root-commit) 5d0c6a0] Initial commit
 1 file changed, 10 insertions(+)
 create mode 100644 index.html
[Bruces-iMac-2:hello_world bruceelgort$ git status
On branch master
nothing to commit, working tree clean
[Bruces-iMac-2:hello_world bruceelgort$ git log
commit 5d0c6a076a481efd58c87f1b91af63ba7a12cbe1 (HEAD -> master)
Author: Bruce Elgort <bruce.elgort@gmail.com>
Date:   Fri Aug 17 15:43:42 2018 -0700

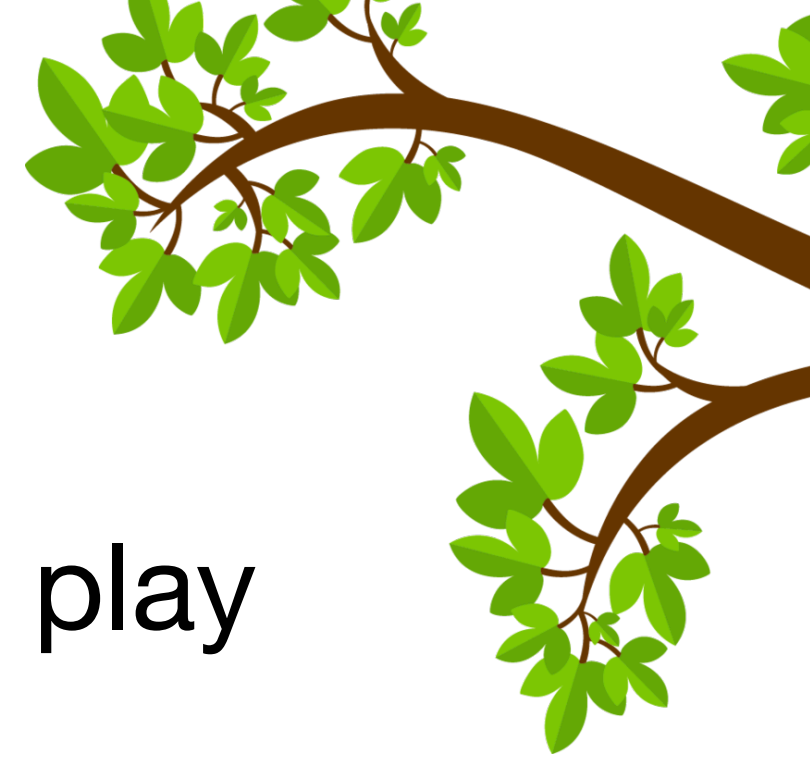
    Initial commit
Bruces-iMac-2:hello_world bruceelgort$
```

git diff

```
hello_world — -bash — 81x23
[Bruces-iMac-2:hello_world bruceelgort$ git status
On branch master
nothing to commit, working tree clean
[Bruces-iMac-2:hello_world bruceelgort$ git log
commit 5d0c6a076a481efd58c87f1b91af63ba7a12cbe1 (HEAD -> master)
Author: Bruce Elgort <bruce.elgort@gmail.com>
Date:   Fri Aug 17 15:43:42 2018 -0700

    Initial commit
[Bruces-iMac-2:hello_world bruceelgort$ git diff
diff --git a/index.html b/index.html
index c2db709..0651544 100644
--- a/index.html
+++ b/index.html
@@ -6,5 +6,6 @@
 </head>
 <body>
     <h1>Hello World</h1>
+    <h2>Goodbye</h2>
 </body>
 </html>
\ No newline at end of file
Bruces-iMac-2:hello_world bruceelgort$
```

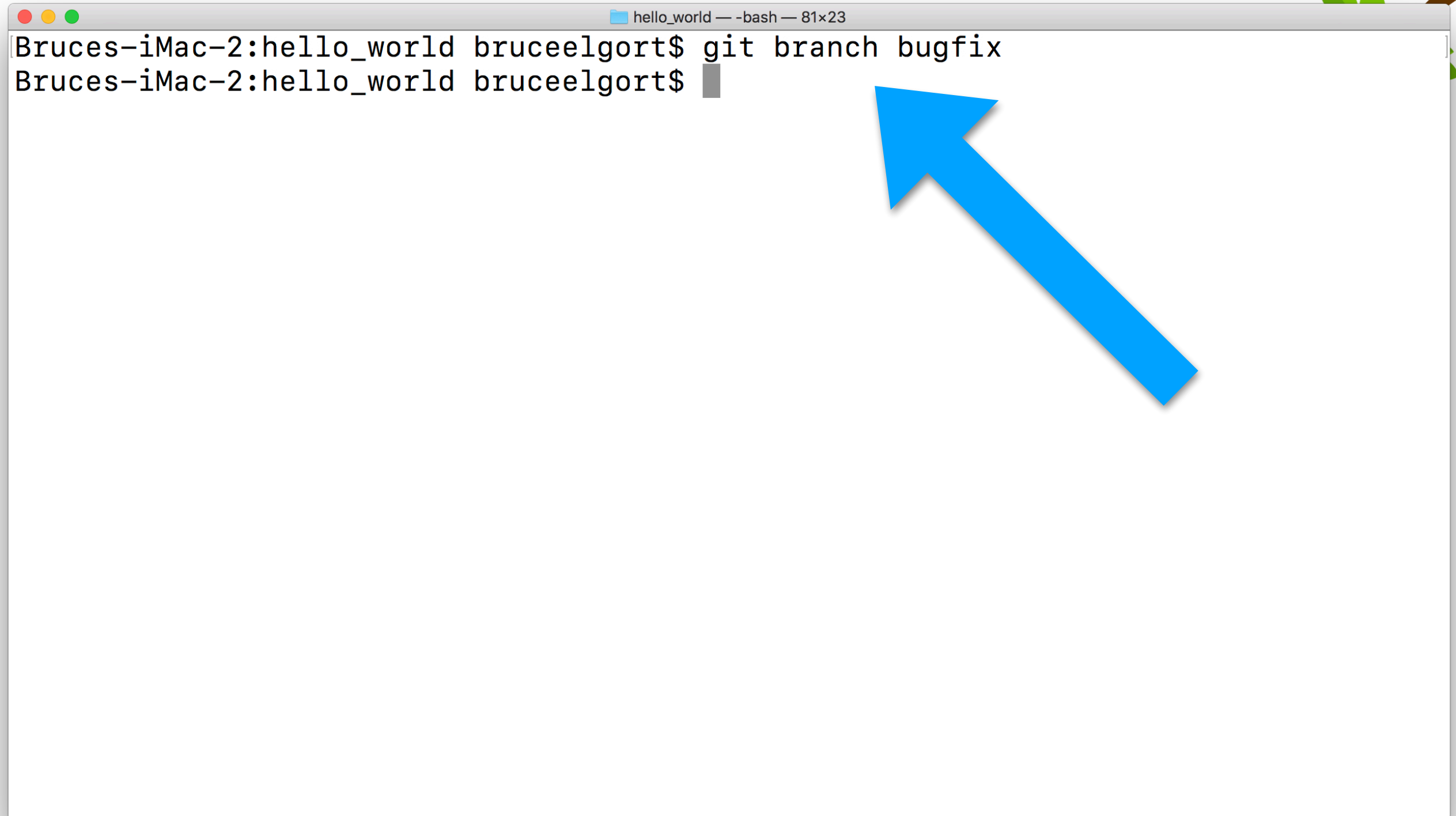




Branching

- Allows you to experiment and play around
- Allows you to not mess up your master branch
- You can then merge a branch back into the master branch
- Or you don't have too
- And yes, you can create branches from branches

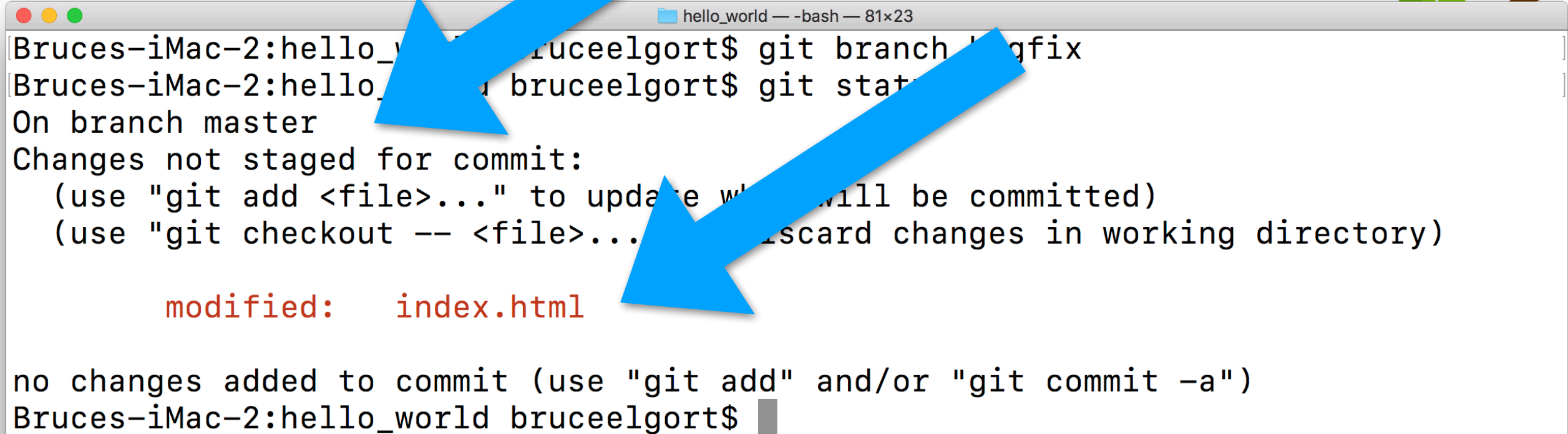
Branching



A terminal window titled "hello_world — -bash — 81x23" is shown. The prompt is "Bruces-iMac-2:hello_world bruceelgort\$". The command "git branch bugfix" has been entered. A large blue arrow points from the bottom right towards the command. The terminal window has a grey title bar with red, yellow, and green window control buttons on the left.

```
hello_world — -bash — 81x23
Bruces-iMac-2:hello_world bruceelgort$ git branch bugfix
Bruces-iMac-2:hello_world bruceelgort$
```

Branching

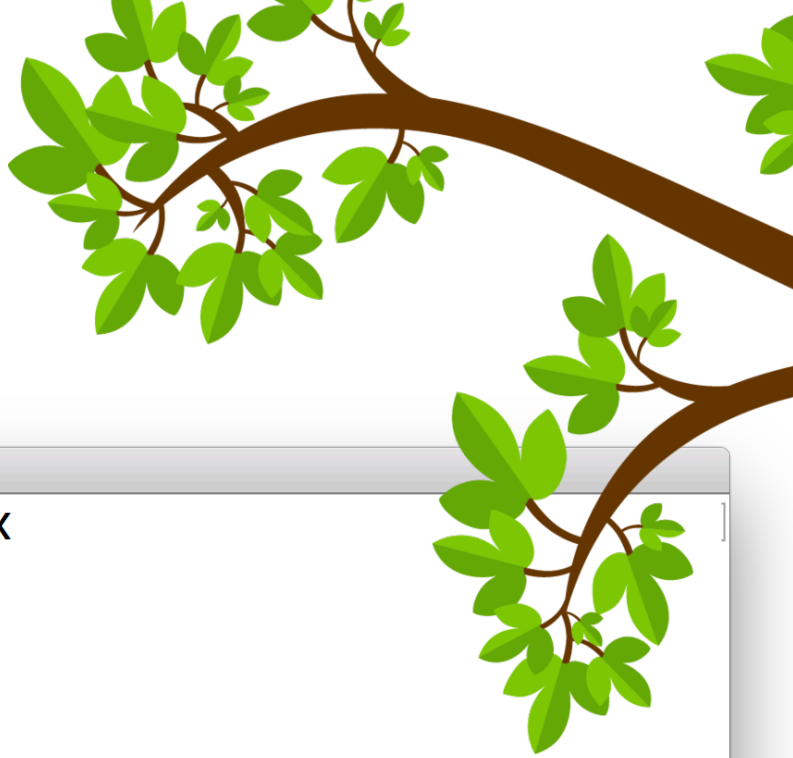



```
hello_world — -bash — 81x23
[Bruces-iMac-2:hello_world] bruceelgort$ git branch logfix
[Bruces-iMac-2:hello_world] bruceelgort$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   index.html

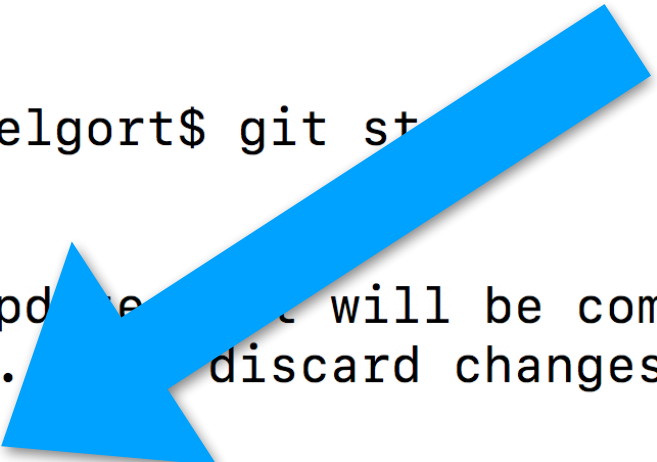
no changes added to commit (use "git add" and/or "git commit -a")
[Bruces-iMac-2:hello_world] bruceelgort$
```


Branching



```
hello_world — -bash — 81x23
Bruces-iMac-2:hello_world bruceelgort$ git checkout bugfix
M      index.html
Switched to branch 'bugfix'
Bruces-iMac-2:hello_world bruceelgort$
```

Branching

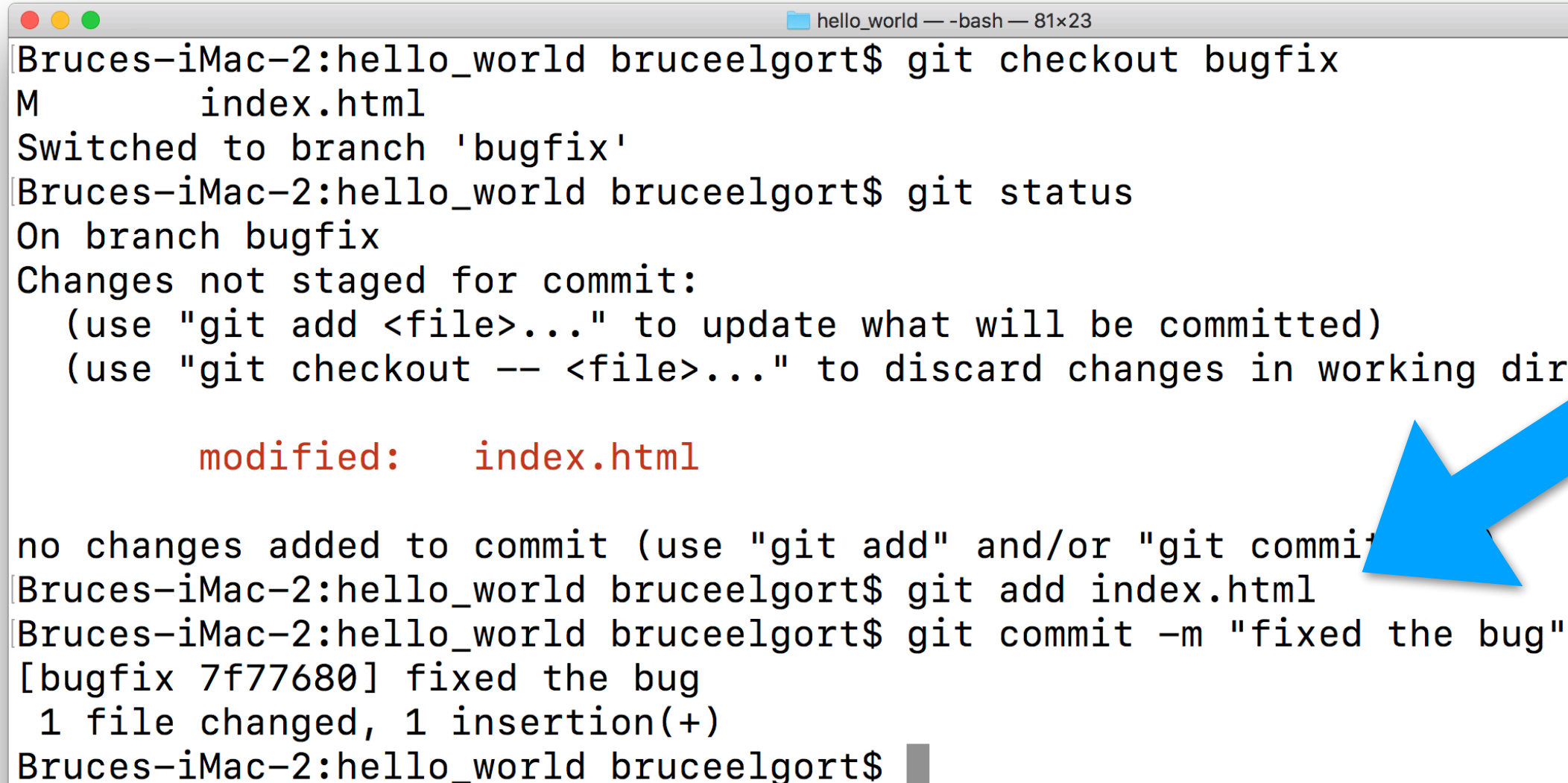


```
hello_world — -bash — 81x23
[Bruces-iMac-2:hello_world bruceelgort$ git checkout bugfix
M       index.html
Switched to branch 'bugfix'
[Bruces-iMac-2:hello_world bruceelgort$ git st
On branch bugfix
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   index.html

no changes added to commit (use "git add" and/or "git commit -a")
Bruces-iMac-2:hello_world bruceelgort$
```

Branching

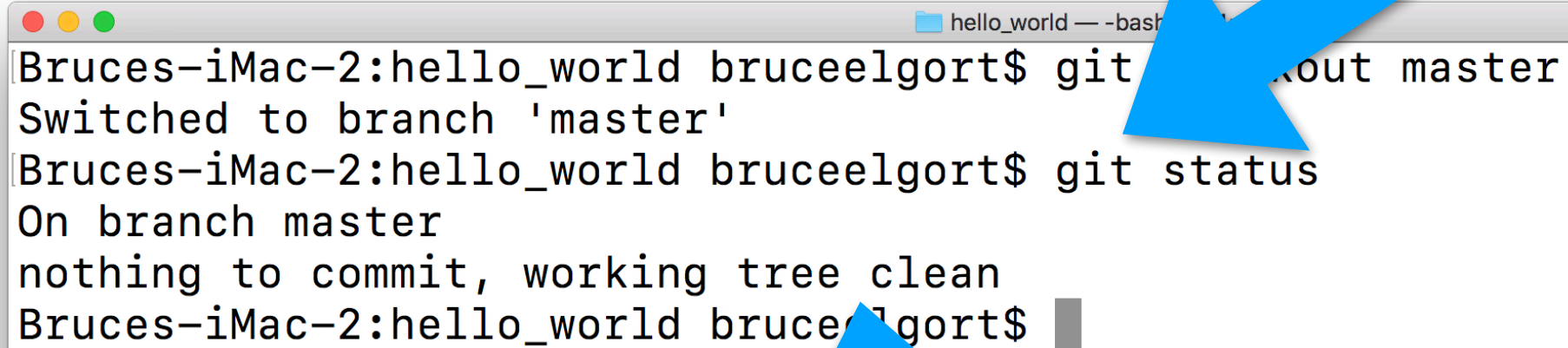


```
hello_world — -bash — 81x23
[Bruces-iMac-2:hello_world bruceelgort$ git checkout bugfix
M      index.html
Switched to branch 'bugfix'
[Bruces-iMac-2:hello_world bruceelgort$ git status
On branch bugfix
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   index.html

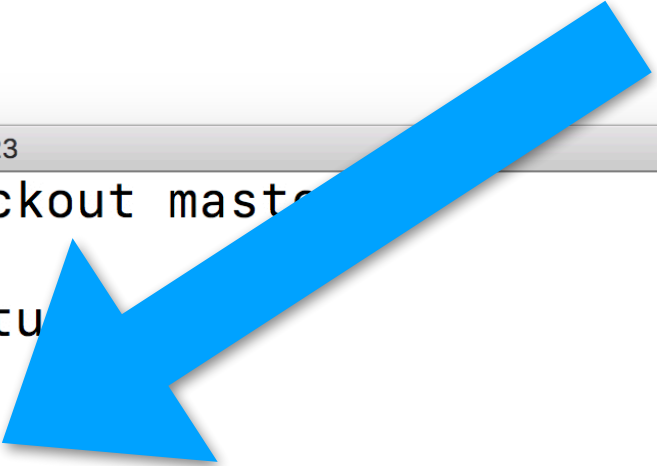
no changes added to commit (use "git add" and/or "git commit" to update)
[Bruces-iMac-2:hello_world bruceelgort$ git add index.html
[Bruces-iMac-2:hello_world bruceelgort$ git commit -m "fixed the bug"
[bugfix 7f77680] fixed the bug
 1 file changed, 1 insertion(+)
Bruces-iMac-2:hello_world bruceelgort$
```

Branching



```
hello_world — -bash
BrucelMac-2:hello_world bruceelgort$ git checkout master
Switched to branch 'master'
BrucelMac-2:hello_world bruceelgort$ git status
On branch master
nothing to commit, working tree clean
BrucelMac-2:hello_world bruceelgort$
```

Get list of branches



```
hello_world — -bash — 81x23
[Bruces-iMac-2:hello_world bruceelgort$ git checkout master
Switched to branch 'master'
[Bruces-iMac-2:hello_world bruceelgort$ git status
On branch master
nothing to commit, working tree clean
[Bruces-iMac-2:hello_world bruceelgort$ git branch
  bugfix
* master
[Bruces-iMac-2:hello_world bruceelgort$
```

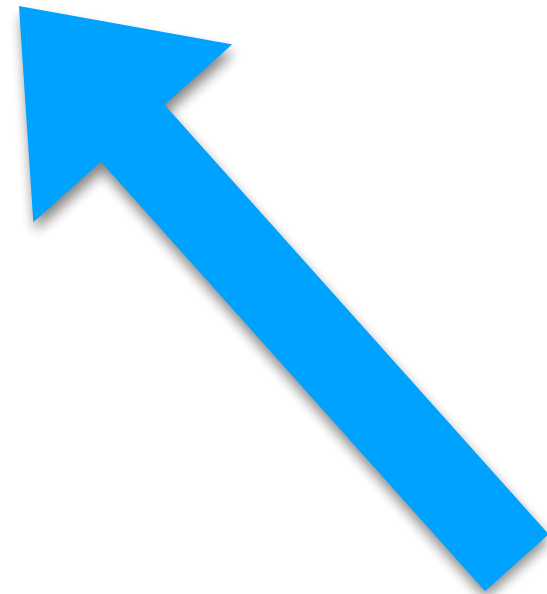
Get list of merged branches

```
hello_world — -bash — 81x23
Bruces-iMac-2:hello_world bruceelgort$ git branch --merged
* master
Bruces-iMac-2:hello_world bruceelgort$
```



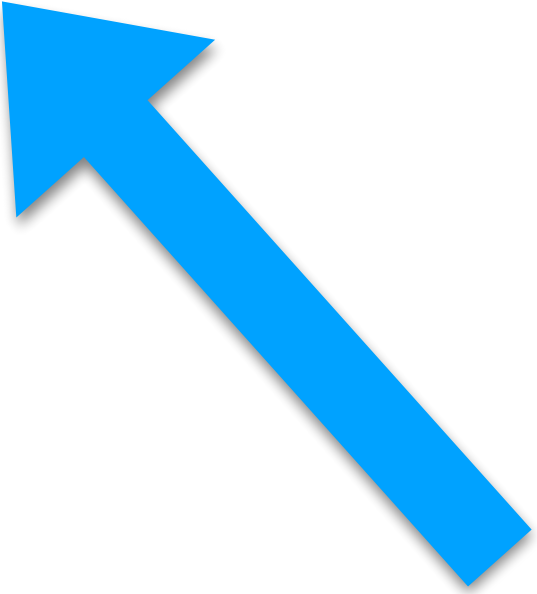

Get list of un-merged branches

```
hello_world — -bash — 81x23
[Bruces-iMac-2:hello_world bruceelgort$ git branch --merged
* master
[Bruces-iMac-2:hello_world bruceelgort$ git branch --no-merge
  bugfix
[Bruces-iMac-2:hello_world bruceelgort$
```



Deleting a branch

```
hello_world — -bash — 81x23
[Bruces-iMac-2:hello_world bruceelgort$ git branch --merged
* master
[Bruces-iMac-2:hello_world bruceelgort$ git branch --no-merge
  bugfix
[Bruces-iMac-2:hello_world bruceelgort$ git branch -D bugfix
Deleted branch bugfix (was 7f77680).
[Bruces-iMac-2:hello_world bruceelgort$
```



Merging branches


```
Bruces-iMac-2:hello_world bruceelgort$  
Bruces-iMac-2:hello_world bruceelgort$ git merge master bugfix  
Updating 5d0c6a0..ba55e7a  
Fast-forward  
 index.html | 1 +  
 1 file changed, 1 insertion(+)  
Bruces-iMac-2:hello_world bruceelgort$ █
```

Merging branches with a conflict demo

Adding a remote origin

```
git remote add origin <server>
```


Adding a remote origin



A terminal window titled "hello_world — -bash — 71x23" with standard macOS window controls (red, yellow, green buttons). The terminal shows the following commands and output:

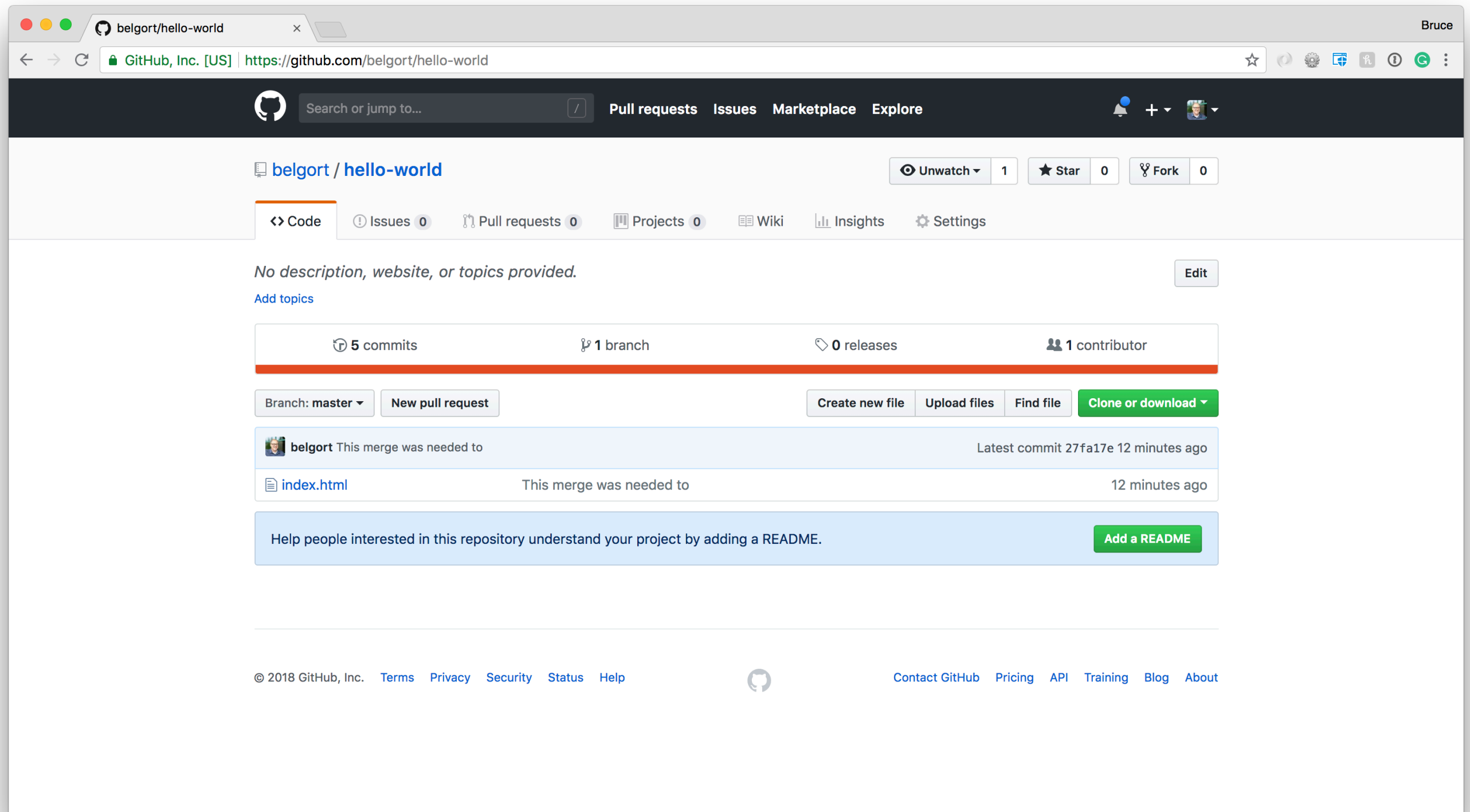
```
Bruces-iMac-2:hello_world bruceelgort$ git remote add origin https://github.com/belgort/hello-world.git
Bruces-iMac-2:hello_world bruceelgort$
```

Pushing a branch to the origin



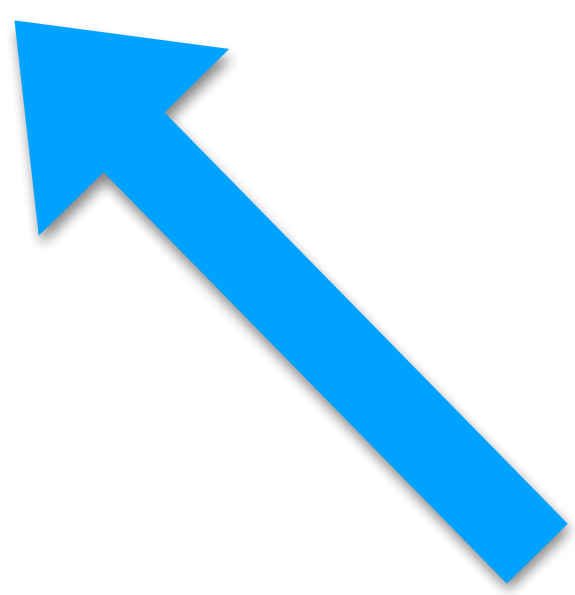
```
hello_world — -bash — 71x23
[Bruces-iMac-2:hello_world bruceelgort$ git remote add origin https://github.com/belgort/hello-world.git
[Bruces-iMac-2:hello_world bruceelgort$ git push origin master
Counting objects: 15, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (10/10), done.
Writing objects: 100% (15/15), 1.40 KiB | 1.40 MiB/s, done.
Total 15 (delta 5), reused 0 (delta 0)
remote: Resolving deltas: 100% (5/5), done.
To https://github.com/belgort/hello-world.git
 * [new branch]      master -> master
Bruces-iMac-2:hello_world bruceelgort$
```

Pushing to the origin

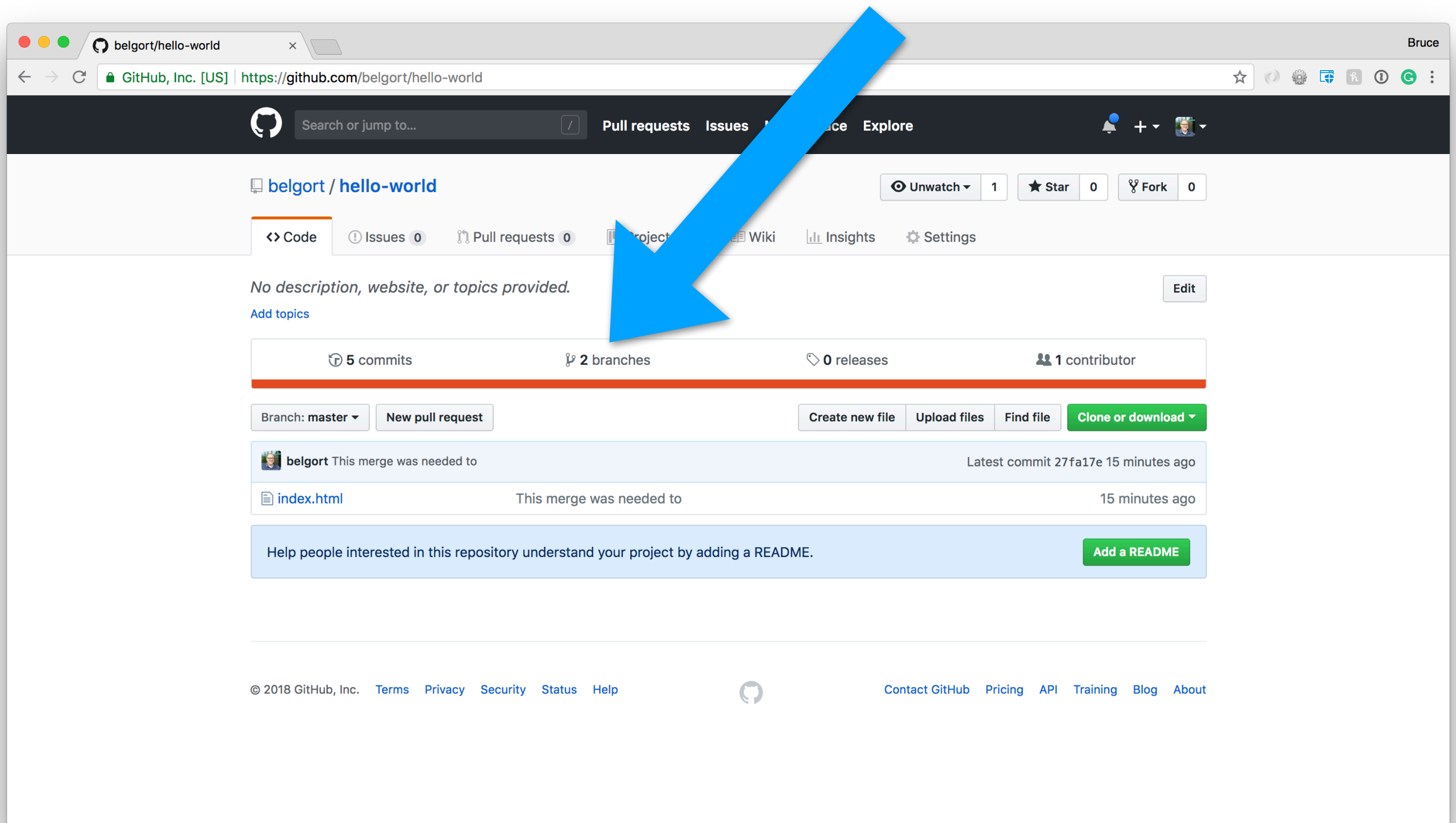


Pushing all branches to the origin

```
hello_world — -bash — 71x23
Bruces-iMac-2:hello_world bruceelgort$ git push origin --all
Total 0 (delta 0), reused 0 (delta 0)
To https://github.com/belgort/hello-world.git
 * [new branch]      bugfix -> bugfix
Bruces-iMac-2:hello_world bruceelgort$
```



Pushing all branches to the origin



The screenshot shows the GitHub interface for the repository 'belgort/hello-world'. A large blue arrow points from the top right towards the '2 branches' status in the repository statistics section.

Repository: belgort / hello-world

Search or jump to... Pull requests Issues Projects Wiki Insights Settings

No description, website, or topics provided. [Add topics](#) [Edit](#)


5 commits 2 branches 0 releases 1 contributor

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

belgort This merge was needed to Latest commit 27fa17e 15 minutes ago

[index.html](#) This merge was needed to 15 minutes ago

Help people interested in this repository understand your project by adding a README. [Add a README](#)

© 2018 GitHub, Inc. [Terms](#) [Privacy](#) [Security](#) [Status](#) [Help](#)  [Contact GitHub](#) [Pricing](#) [API](#) [Training](#) [Blog](#) [About](#)

Pull from the origin

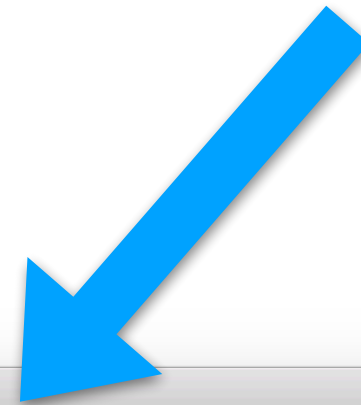
```
git pull
```

Pull from the origin



```
hello_world — -bash — 71x23
[Brucel-iMac-2:hello_world bruceelgort$ git pull origin master
From https://github.com/belgort/hello-world
 * branch          master      -> FETCH_HEAD
Already up to date.
Brucel-iMac-2:hello_world bruceelgort$
```

Show remote info



```
hello_world — -bash — 71x23
[Bruces-iMac-2:hello_world bruceelgort$ git remote -v
origin  https://github.com/belgort/hello-world.git (fetch)
origin  https://github.com/belgort/hello-world.git (push)
Bruces-iMac-2:hello_world bruceelgort$
```

Cloning a repo

The screenshot shows a web browser window displaying the GitHub repository page for 'belgort/Clark_Arboretum'. The repository has 6 commits, 3 branches, 0 releases, and 2 contributors. The 'Clone or download' button is highlighted, and a dropdown menu is open, showing the 'Clone with HTTPS' option selected. A blue arrow points to the 'Clone with HTTPS' option. The repository name is 'belgort / Clark_Arboretum'. The repository description is 'Clark College Arboretum Site'. The repository has 6 commits, 3 branches, 0 releases, and 2 contributors. The repository is currently on the 'master' branch. The repository has a 'Design Files' folder, a 'css' folder, an 'img' folder, a 'js' folder, a 'pdf' folder, a 'php' folder, and an 'index.html' file. The repository was last updated 4 years ago. The repository has a 'README' file. The repository has a 'README' button.

git - the simple guide - no dee x belgort/Clark_Arboretum: Clark x

GitHub, Inc. [US] https://github.com/belgort/Clark_Arboretum

Search or jump to... Pull requests Issues Marketplace Explore

belgort / Clark_Arboretum Unwatch 2 Star 0 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

Clark College Arboretum Site Edit

Add topics

6 commits 3 branches 0 releases 2 contributors

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

belgort-clark Modified CSS

Design Files	AI Design Files are Now Included
css	Modified CSS
img	Revised image tiles for new map
js	Revised image tiles for new map
pdf	Revised image tiles for new map
php	Minor changes to About Page
index.html	Cleaned up HTML

4 years ago 4 years ago 4 years ago

Help people interested in this repository understand your project by adding a README. Add a README

export_20180817.csv 36c3c8ea63c7593....png 36c3c8ea63c7593....png Show All

Cloning a repo



```
Bruces-iMac-2:trees bruceelgort$ git clone https://github.com/belgort/Clark_Arbo  
retum.git  
Cloning into 'Clark_Arboretum'...  
remote: Counting objects: 3525, done.  
remote: Total 3525 (delta 0), reused 0 (delta 0), pack-reused 3525  
Receiving objects: 100% (3525/3525), 116.34 MiB | 22.87 MiB/s, done.  
Resolving deltas: 100% (288/288), done.  
Bruces-iMac-2:trees bruceelgort$
```

Name	Date Modified	Size
Clark_Arboretum	Today at 8:27 PM	--
└─ .git	Today at 8:27 PM	--
└─ css	Today at 8:27 PM	--
└─ Design Files	Today at 8:27 PM	--
└─ img	Today at 8:27 PM	--
└─ index.html	Today at 8:27 PM	23 KB
└─ js	Today at 8:27 PM	--
└─ pdf	Today at 8:27 PM	--
└─ php	Today at 8:27 PM	--

Learning Resources

Git SCM Site

<http://git-scm.com>

Pro Git Book

<https://git-scm.com/book/en/v2>

Oh shit, git!

<https://ohshitgit.com/>

Git - The Simple Guide

<http://rogerdudler.github.io/git-guide/>

Visual Studio Code Source Control

<https://code.visualstudio.com/docs/editor/versioncontrol>

Git Apps for Mac

Git - GUI Clients

Secure | https://git-scm.com/download/gui/mac

git --distributed-even-if-your-workflow-isnt

Search entire site...

About

Documentation

Downloads

GUI Clients

Logos

Community

The entire **Pro Git book** written by Scott Chacon and Ben Straub is available to [read online for free](#). Dead tree versions are available on [Amazon.com](#).

GUI Clients

Git comes with built-in GUI tools for committing ([git-gui](#)) and browsing ([gitk](#)), but there are several third-party tools for users looking for platform-specific experience.

If you want to add another GUI tool to this list, just [follow the instructions](#).

All

Windows

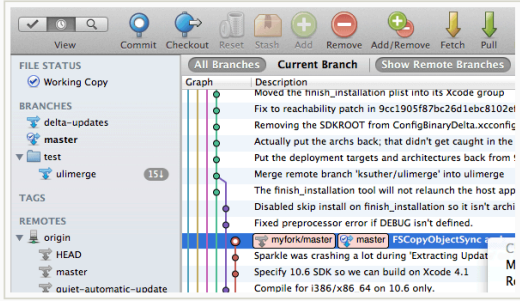
Mac

Linux

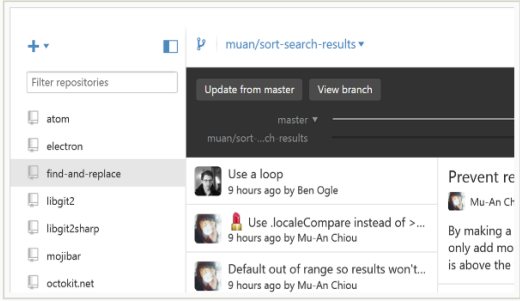
Android

iOS

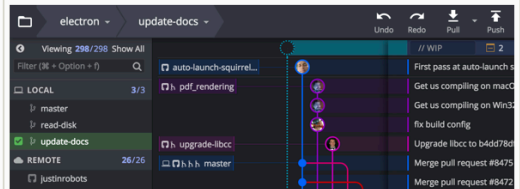
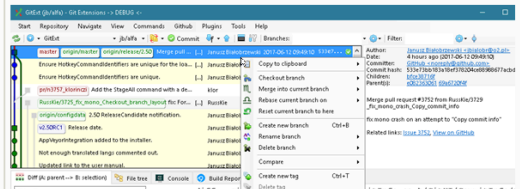
19 Mac GUIs are shown below ↓



SourceTree
Platforms: Mac, Windows
Price: Free
License: Proprietary

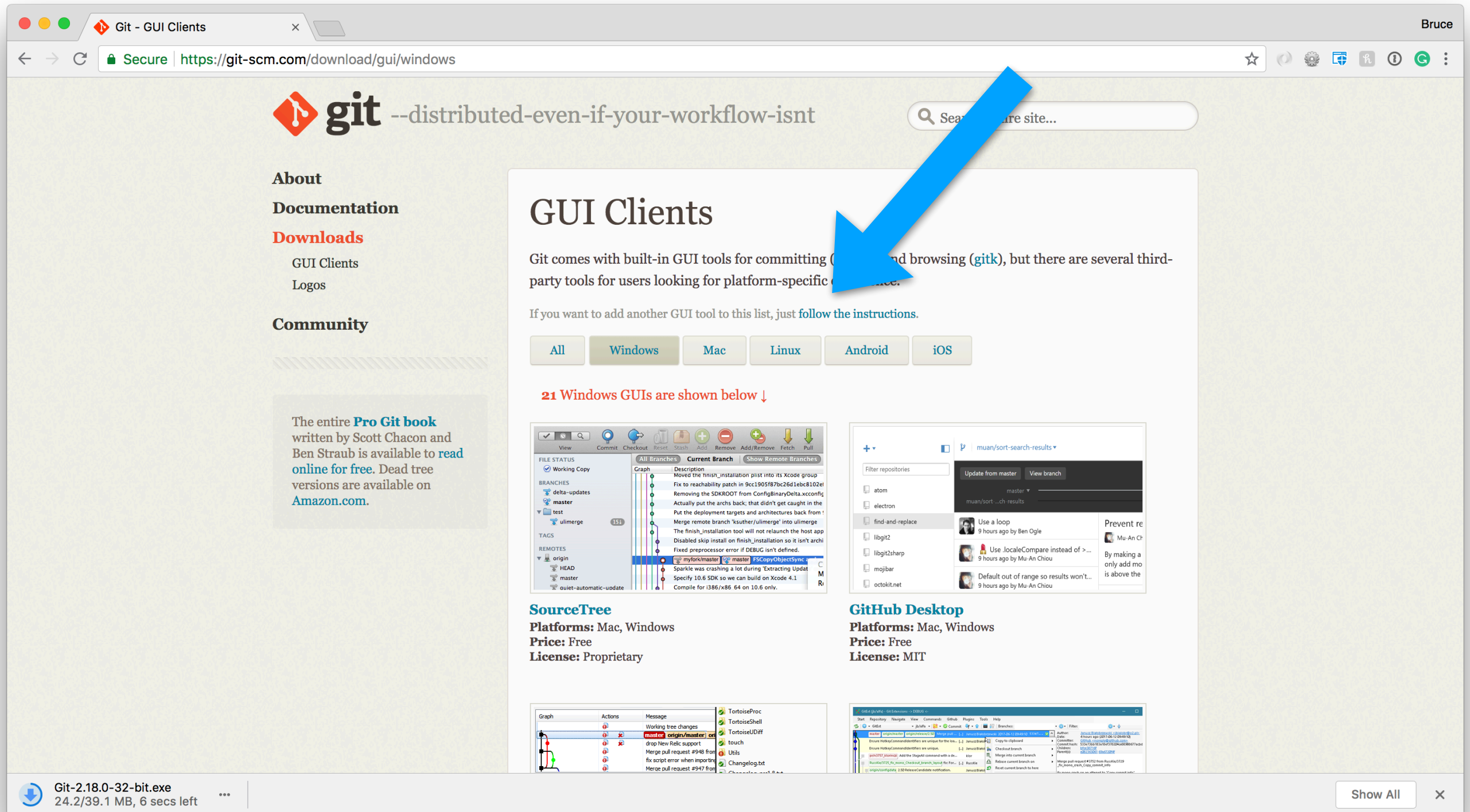


GitHub Desktop
Platforms: Mac, Windows
Price: Free
License: MIT



<https://git-scm.com/download/gui/mac>

Git Apps for Windows



The screenshot shows a web browser window titled "Git - GUI Clients" with the URL <https://git-scm.com/download/gui/windows>. The page features the Git logo and the tagline "--distributed-even-if-your-workflow-isnt". A sidebar on the left contains links for "About", "Documentation", "Downloads" (highlighted in red), and "Community". The "Downloads" section lists "GUI Clients" and "Logos". The main content area is titled "GUI Clients" and explains that Git includes built-in GUI tools like `gitk` and `git-gui`, but also lists several third-party tools. A blue arrow points to the "GUI Clients" heading. Below this, there are filter buttons for "All", "Windows" (selected), "Mac", "Linux", "Android", and "iOS". A red text box states "21 Windows GUIs are shown below ↓". The page displays a grid of GUI client thumbnails, including SourceTree, GitHub Desktop, and TortoiseGit. At the bottom, a download bar shows "Git-2.18.0-32-bit.exe" with a size of 24.2/39.1 MB and 6 seconds left. A "Show All" button is in the bottom right corner.

git --distributed-even-if-your-workflow-isnt

About
Documentation
Downloads
GUI Clients
Logos
Community

The entire **Pro Git book** written by Scott Chacon and Ben Straub is available to **read online for free**. Dead tree versions are available on [Amazon.com](https://www.amazon.com/).

GUI Clients

Git comes with built-in GUI tools for committing (`gitk`) and browsing (`gitk`), but there are several third-party tools for users looking for platform-specific GUIs.

If you want to add another GUI tool to this list, just [follow the instructions](#).

All **Windows** **Mac** **Linux** **Android** **iOS**

21 Windows GUIs are shown below ↓

SourceTree
Platforms: Mac, Windows
Price: Free
License: Proprietary

GitHub Desktop
Platforms: Mac, Windows
Price: Free
License: MIT

Git-2.18.0-32-bit.exe
24.2/39.1 MB, 6 secs left

Show All

<https://git-scm.com/download/gui/windows>