

Getting started with Git

Ahmed Salijee



Getting started with Git



1. Why and what



3. Let's get cloning



5. What about GitHub



2. Installing the bits



4. I want my own repo



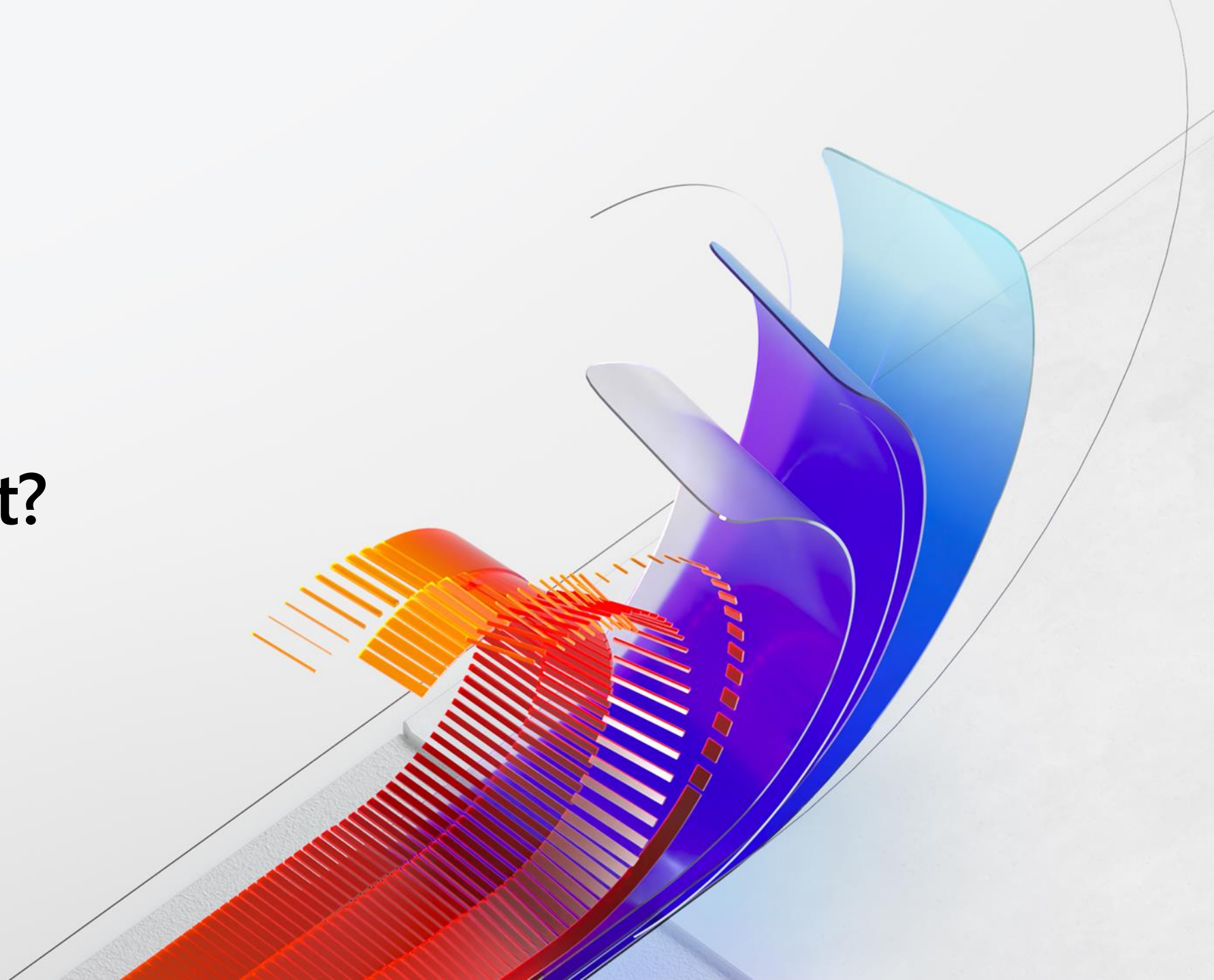
6. Let's do a pull request



Hands on, a bit challenging, let's have fun

Why and what?

Ahmed Saljee



What is Git?

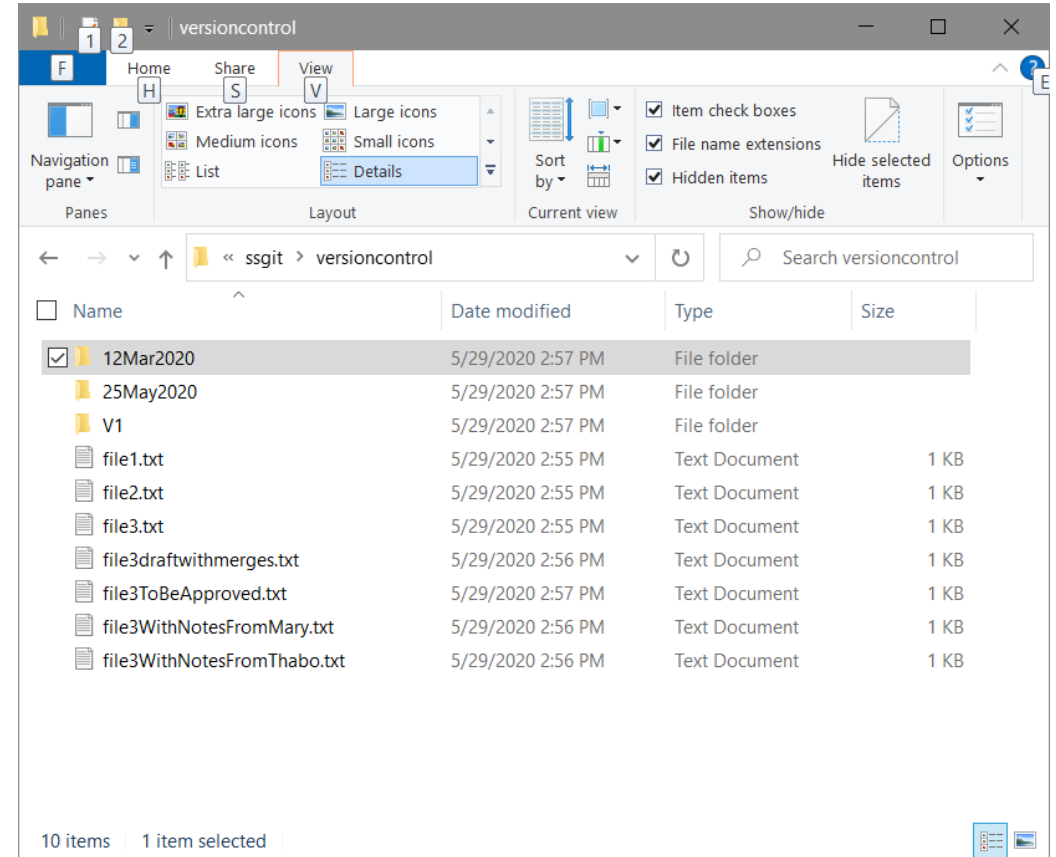
From Wikipedia

Git is a ***distributed version-control system*** for tracking changes in source code during software development.

*Not totally accurate – does not need to be source code or software dev.
It can track and manage changes in general*

Why a version control system

- Keeping track of changes is challenging.
- VCS
 - Tools that help a team manage changes to **source** code (files) over time.
 - keeps track of every modification to the code (files)
 - in a special kind of database.
 - Who made the changes, when – and maybe why
 - Go back in time
 - Work in isolation if needed



I am not a dev – this is not important...

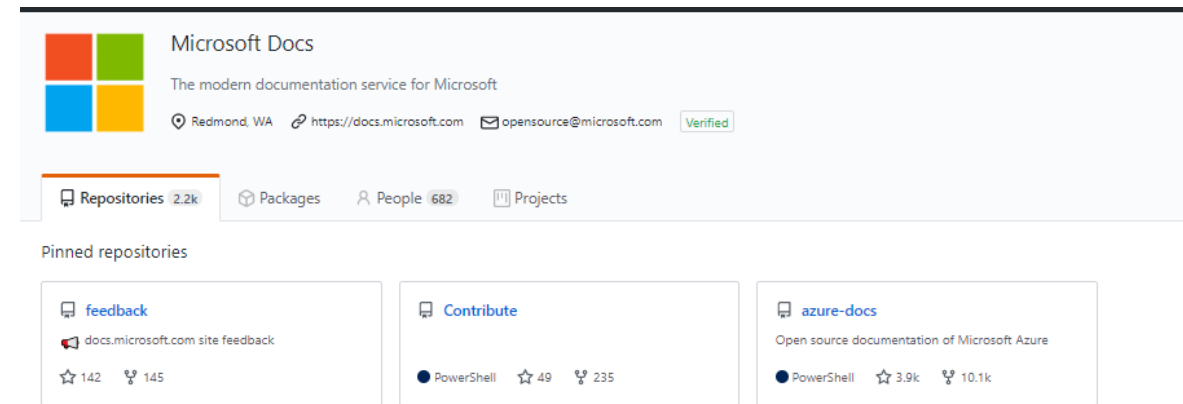
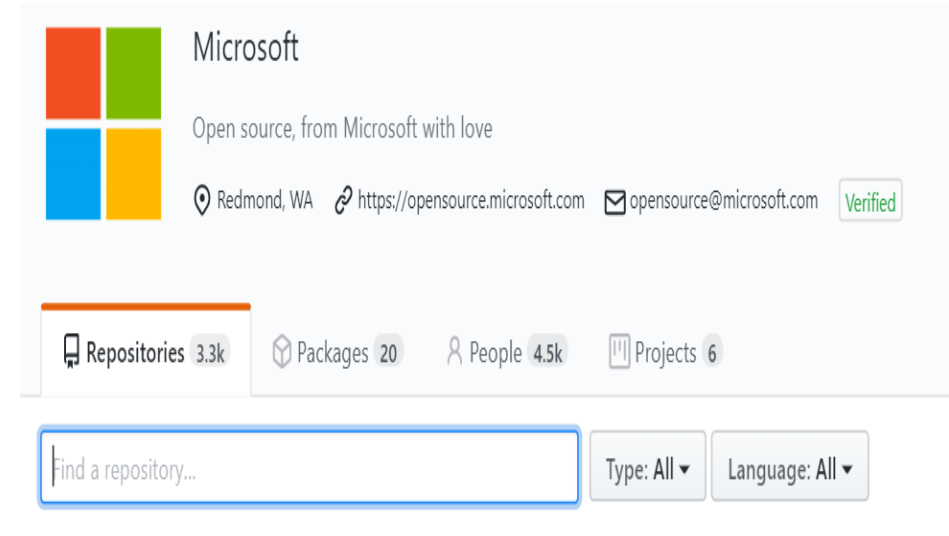
Your customers have devs 😊

Git and GitHub are collaboration tools
Used beyond software development

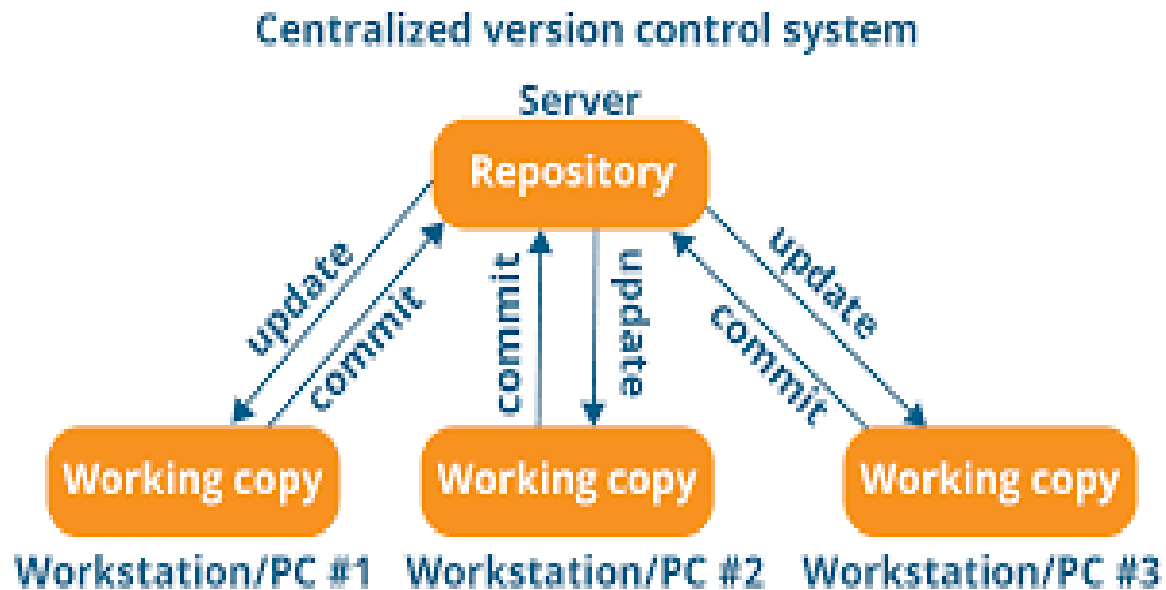
Microsoft docs using Git/GitHub
Many workshops/material posted on GitHub

GitHub (and Azure DevOps) is more than just Git.

Think of it as social network

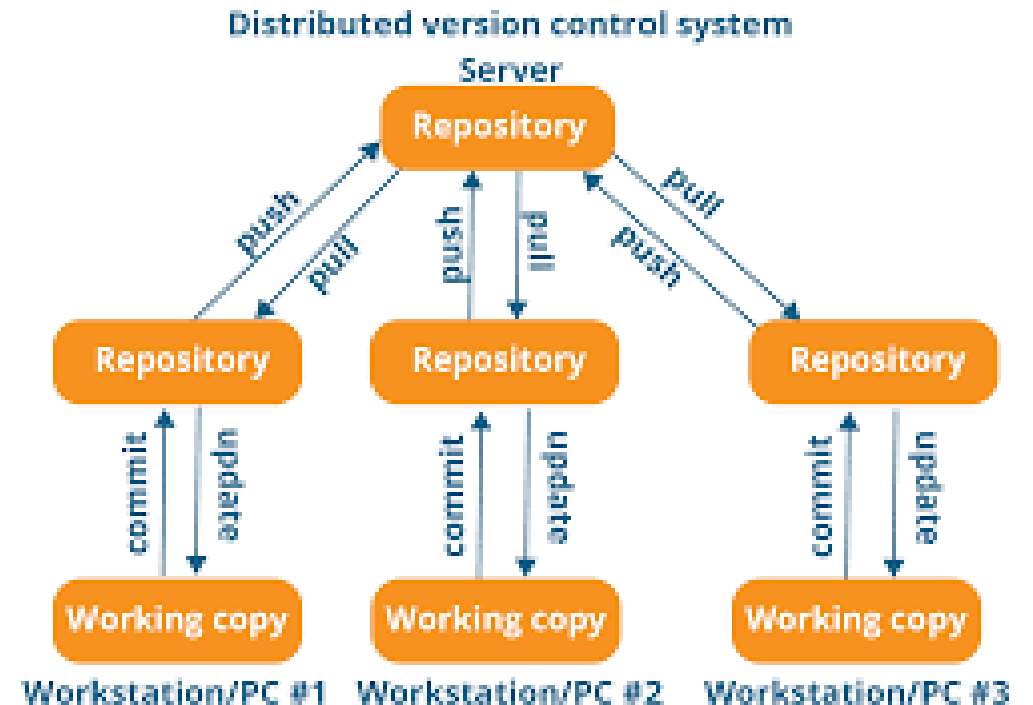


Central vs distributed



A single server that contains all the versioned files, and a number of clients that check out files from that central place

Eg CVS, TFVC, Sourcesafe, Subversion



Clients don't just check out the latest snapshot of the files; rather, they fully mirror (download) the repository, including its full history.

Eg Git, Mercurial, Bazaar or Darcs

Git vs GitHub

Git is a version control system (and associated protocols) that lets you manage and keep track of your files.

GitHub is a cloud-based hosting service that lets you manage Git repositories (and more!)

IMPORTANT - There are other Git hosting services (onprem, cloud based)

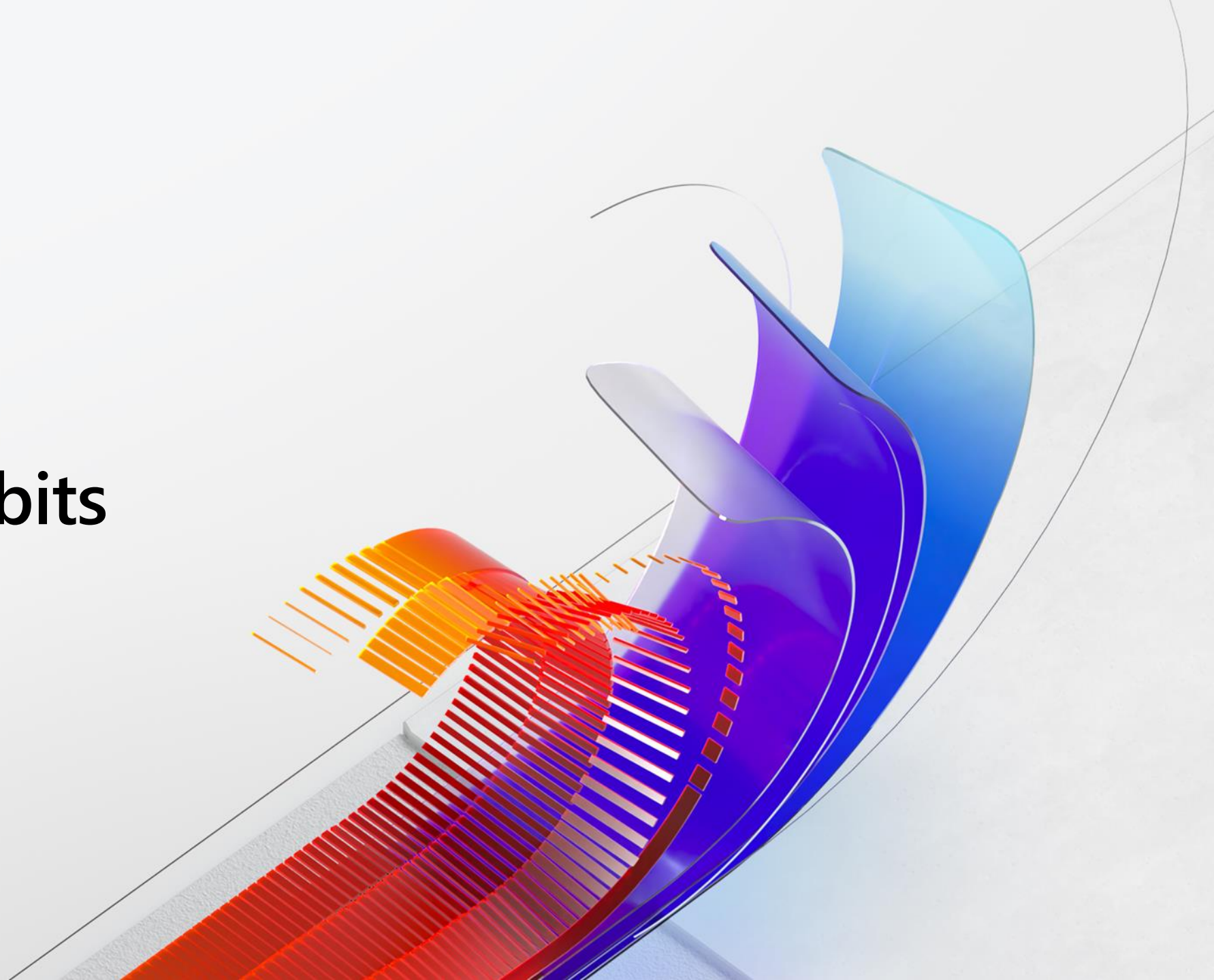
- Azure Repos (part of Azure DevOps)
- BitBucket
- GitLab
-

All of these a lot more services (managing work, issues, test case, deployments etc)

If a customer says they want a Git repo, GitHub is just one of many options for hosting it
- just like hosting a website

Installing the bits

Ahmed Salijee



Installing the Bits

- We are not going to do this..
 - You can visit <https://git-scm.com/downloads>
 - download and install Git
- Let's try something different
 - Install Chocolatey – a package manager
 - STEPS – Visit <https://chocolatey.org/install>
 - Open Powershell in Admin mode (Admin mode!)
 - Copy,paste and run in powershell window the script from website (or the bit below)

```
Set-ExecutionPolicy Bypass -Scope Process -Force; [System.Net.ServicePointManager]::SecurityProtocol =  
[System.Net.ServicePointManager]::SecurityProtocol -bor 3072; iex ((New-Object  
System.Net.WebClient).DownloadString('https://chocolatey.org/install.ps1'))
```

Chocolatey Install:

Individual

Organization

1. First, ensure that you are using an *administrative shell* - you can also install as a non-admin, check out [Non-Administrative Installation](#).
2. Install with powershellExe

NOTE: Please inspect <https://chocolatey.org/install.ps1> prior to running any of these scripts to ensure safety. We already know it's safe, but you should verify the security and contents of *any* script from the internet you are not familiar with. All of these scripts download a remote PowerShell script and execute it on your machine. We take security very seriously. [Learn more about our security protocols](#).

With PowerShell, you must ensure *Get-ExecutionPolicy* is not Restricted. We suggest using *Bypass* to bypass the policy to get things installed or *AllSigned* for quite a bit more security.


- Run *Get-ExecutionPolicy*. If it returns *Restricted*, then run *Set-ExecutionPolicy AllSigned* or *Set-ExecutionPolicy Bypass -Scope Process*.

Now run the following command:





```
Set-ExecutionPolicy Bypass -Scope Process -Force; [System.Net.ServicePointManager]::SecurityProtocol = [System.Net.ServicePointManager
```

All Apps Documents Web More ▾ 15


Best match

 **Windows PowerShell**
App

Apps

-  WinSCP >
-  Windows Terminal >
-  Windows Media Player >
-  Windows Fax and Scan >

Search the web


 win - See web results >


Settings (4+)

Documents - SharePoint (4+)


People (1+)


Documents - OneDrive for Business (1+)


 Windows PowerShell





Windows PowerShell
App

 Open

 Run as Administrator

 Run ISE as Administrator

 Windows PowerShell ISE

 Administrator: Windows PowerShell

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

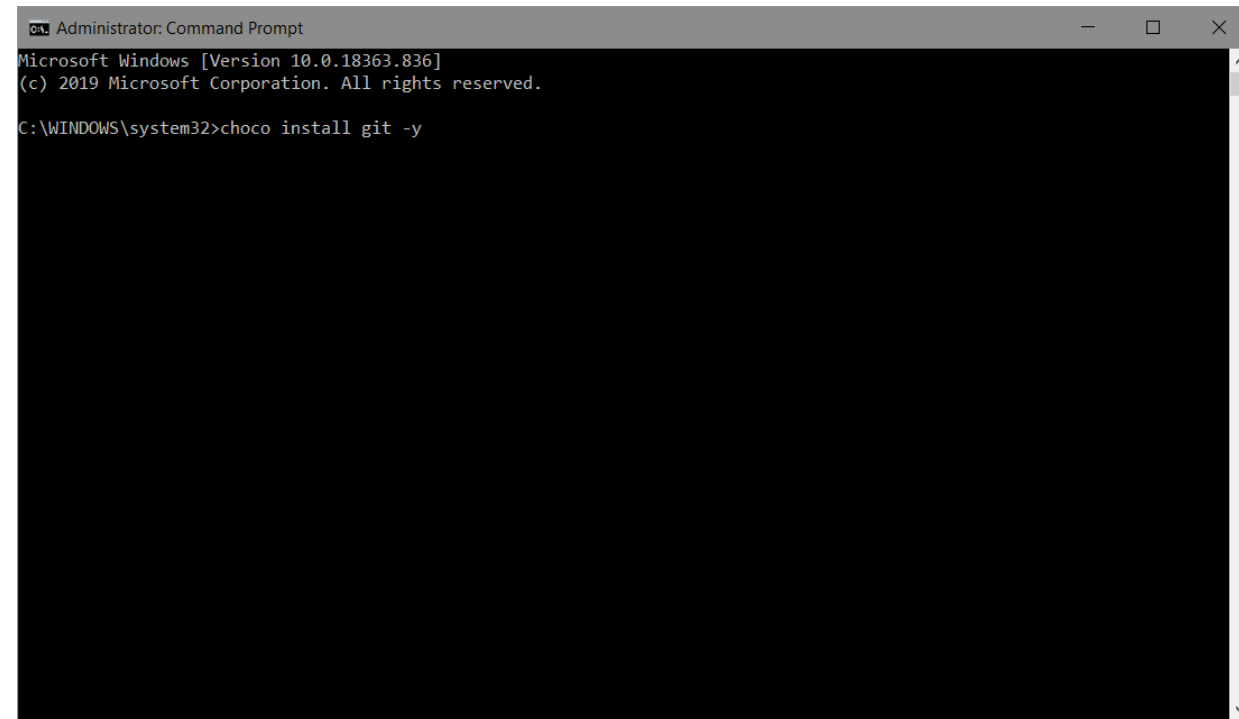
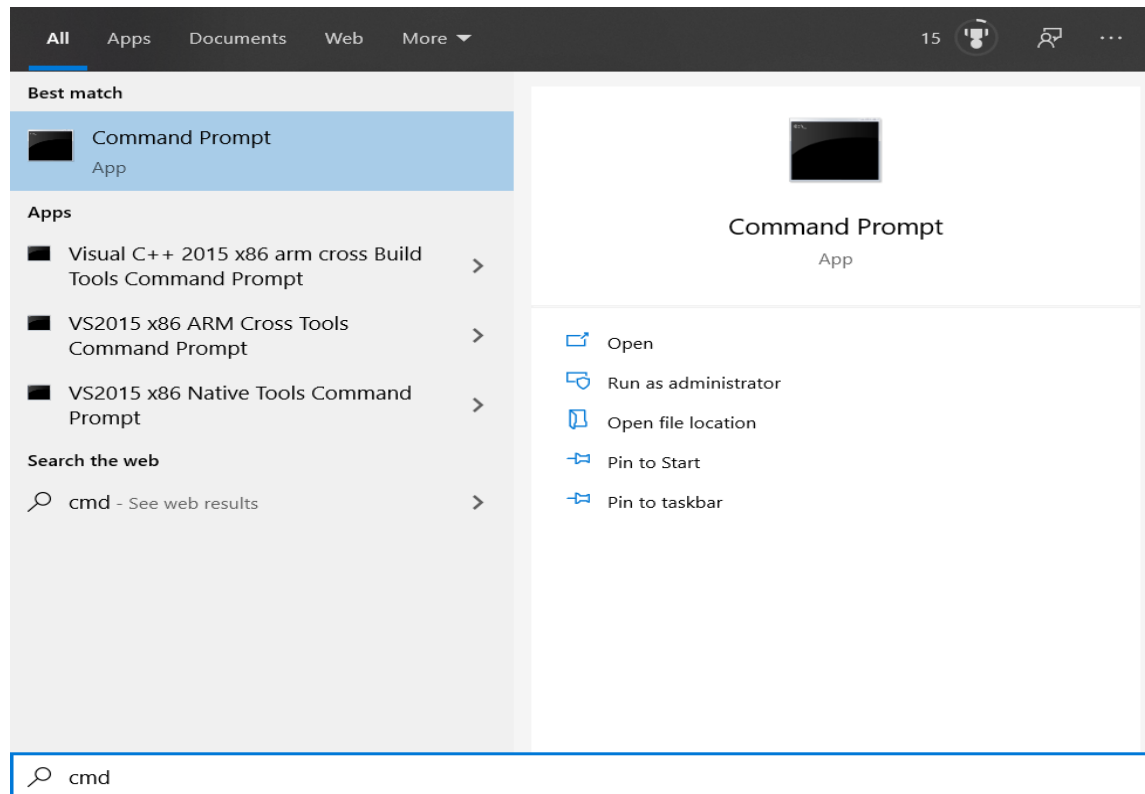
Try the new cross-platform PowerShell <https://aka.ms/pscore6>

```
PS C:\WINDOWS\system32> Set-ExecutionPolicy Bypass -Scope Process -Force; [System.Net.ServicePointManager]::SecurityProtocol = [System.Net.ServicePointManager]::SecurityProtocol -bor 3072; iex ((New-Object System.Net.WebClient).DownloadString('https://chocolatey.org/install.ps1'))
Getting latest version of the Chocolatey package for download.
Getting Chocolatey from https://chocolatey.org/api/v2/package/chocolatey/0.10.15.
Downloading 7-Zip commandline tool prior to extraction.
Extracting C:\Users\ahmeds\AppData\Local\Temp\chocolatey\chocInstall\chocolatey.zip to C:\Users\ahmeds\AppData\Local\Temp\chocolatey\chocInstall...
Installing chocolatey on this machine
Creating ChocolateyInstall as an environment variable (targeting 'Machine')
  Setting ChocolateyInstall to 'C:\ProgramData\chocolatey'
WARNING: It's very likely you will need to close and reopen your shell
  before you can use choco.
```

Installing the bits

- Open command prompt in admin mode (type cmd)

```
Choco install git -y  
Choco install vscode -y  
Echo test
```



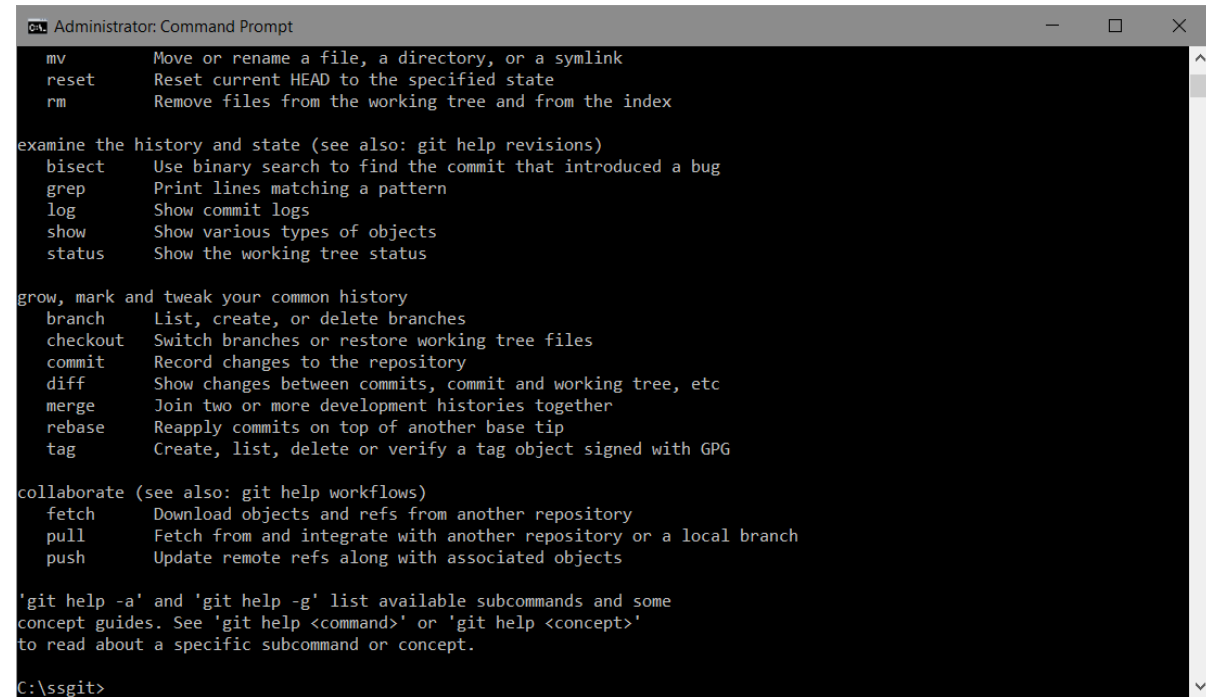
Make sure git is installed and get directory in place

- Close current command prompt
- Open a new command prompt
- Type

```
git
```

- You should get something back
- Copy and paste

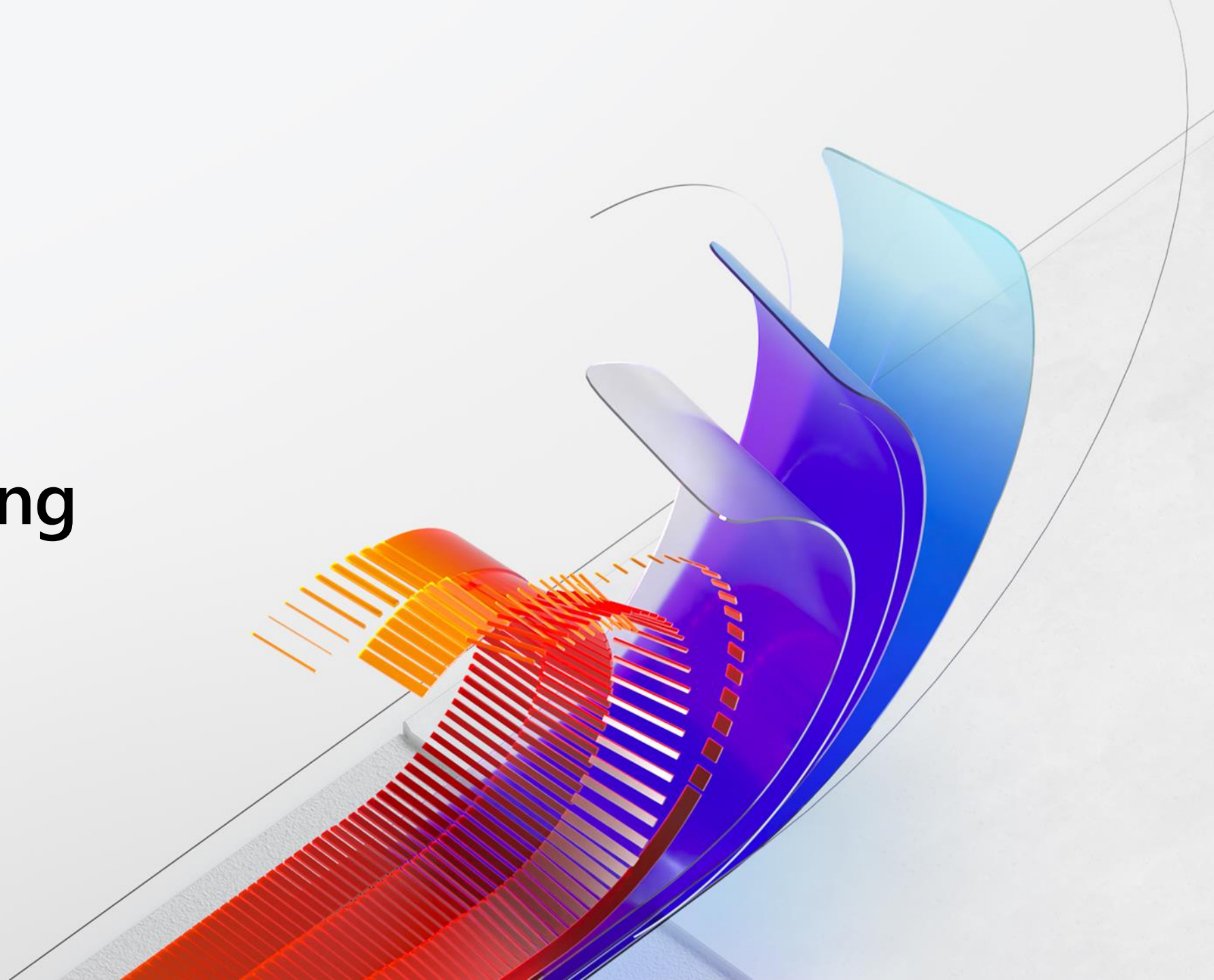
```
cd \  
mkdir ssgit  
cd ssgit  
mkdir mygitrepo  
dir
```



The screenshot shows a Windows Command Prompt window titled "Administrator: Command Prompt". The output of the 'git' command is displayed, listing various subcommands and their descriptions. The commands are grouped into sections: basic file operations, examining history and state, growing and tweaking history, and collaborating. The prompt at the bottom shows the current directory as 'C:\ssgit'.

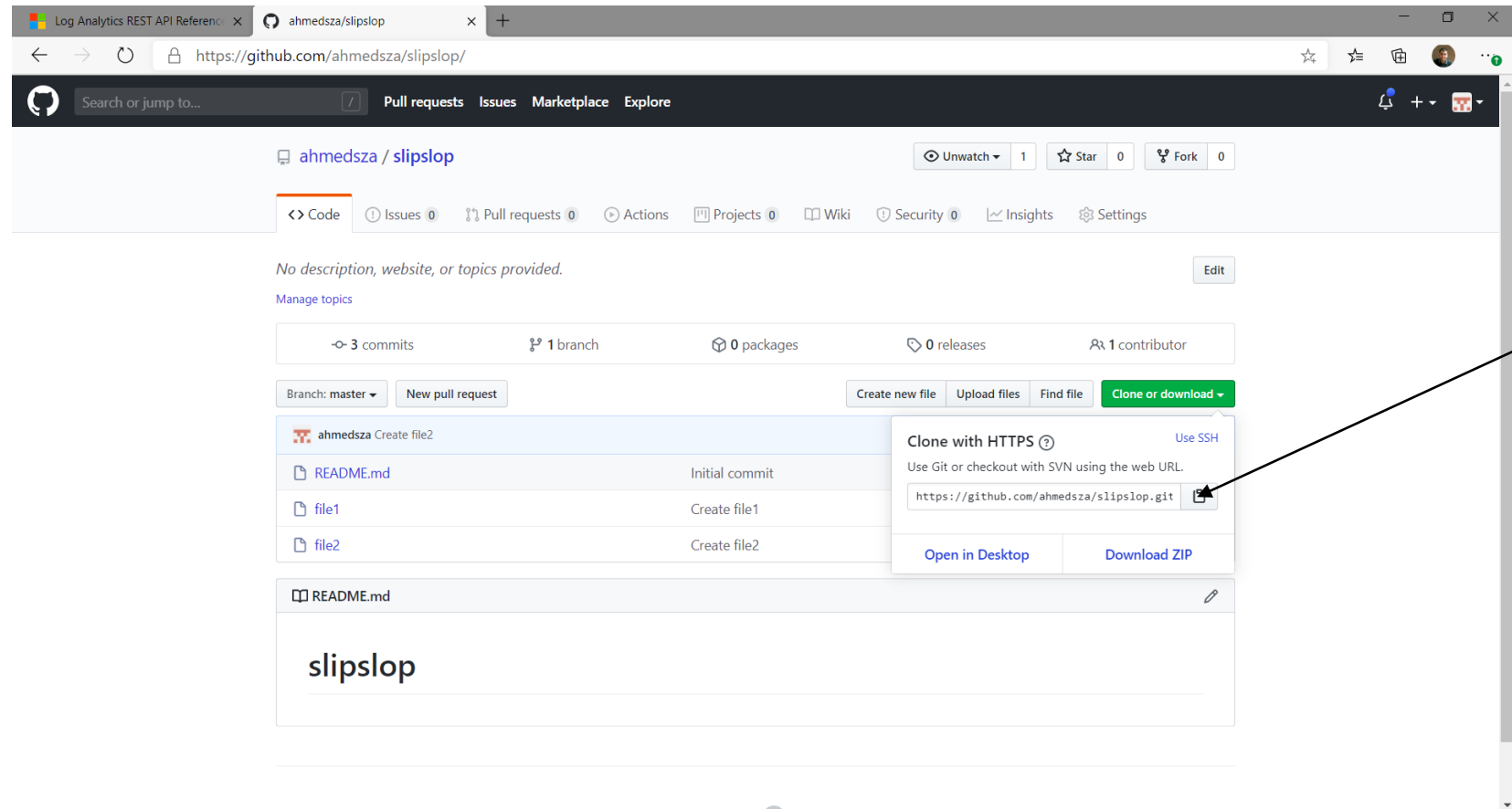
```
Administrator: Command Prompt  
  
mv      Move or rename a file, a directory, or a symlink  
reset   Reset current HEAD to the specified state  
rm      Remove files from the working tree and from the index  
  
examine the history and state (see also: git help revisions)  
bisect   Use binary search to find the commit that introduced a bug  
grep     Print lines matching a pattern  
log      Show commit logs  
show     Show various types of objects  
status   Show the working tree status  
  
grow, mark and tweak your common history  
branch   List, create, or delete branches  
checkout Switch branches or restore working tree files  
commit   Record changes to the repository  
diff     Show changes between commits, commit and working tree, etc  
merge    Join two or more development histories together  
rebase   Reapply commits on top of another base tip  
tag      Create, list, delete or verify a tag object signed with GPG  
  
collaborate (see also: git help workflows)  
fetch    Download objects and refs from another repository  
pull     Fetch from and integrate with another repository or a local branch  
push     Update remote refs along with associated objects  
  
'git help -a' and 'git help -g' list available subcommands and some  
concept guides. See 'git help <command>' or 'git help <concept>'  
to read about a specific subcommand or concept.  
  
C:\ssgit>
```


Let's get cloning



Let's get cloning

- Visit <https://github.com/ahmedsza/slipslop/>
- Click the green clone or download button
- Copy URL

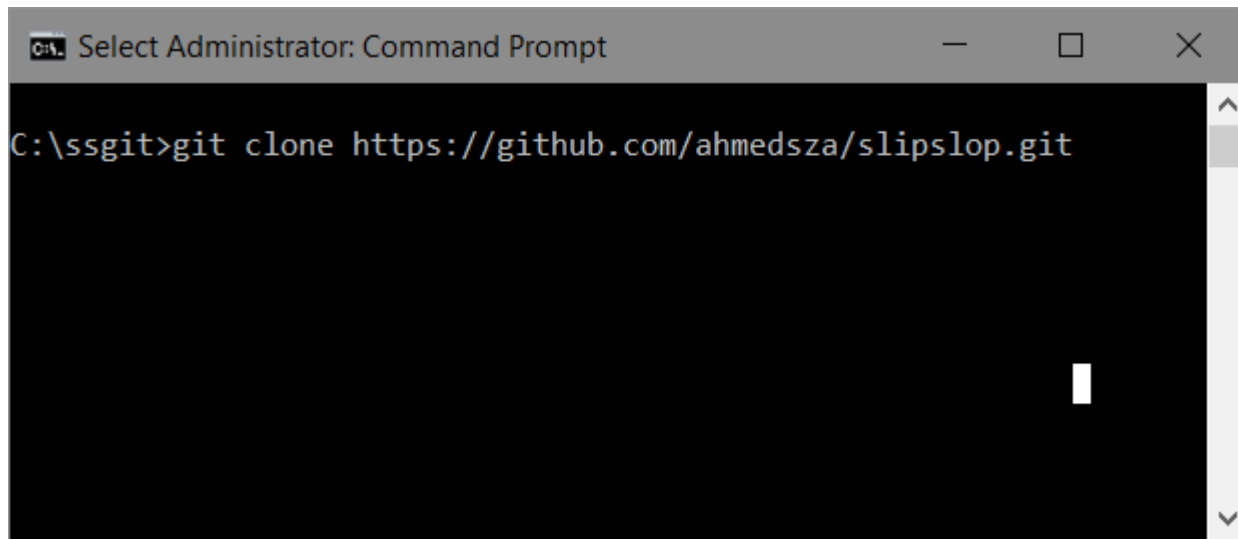


Git clone

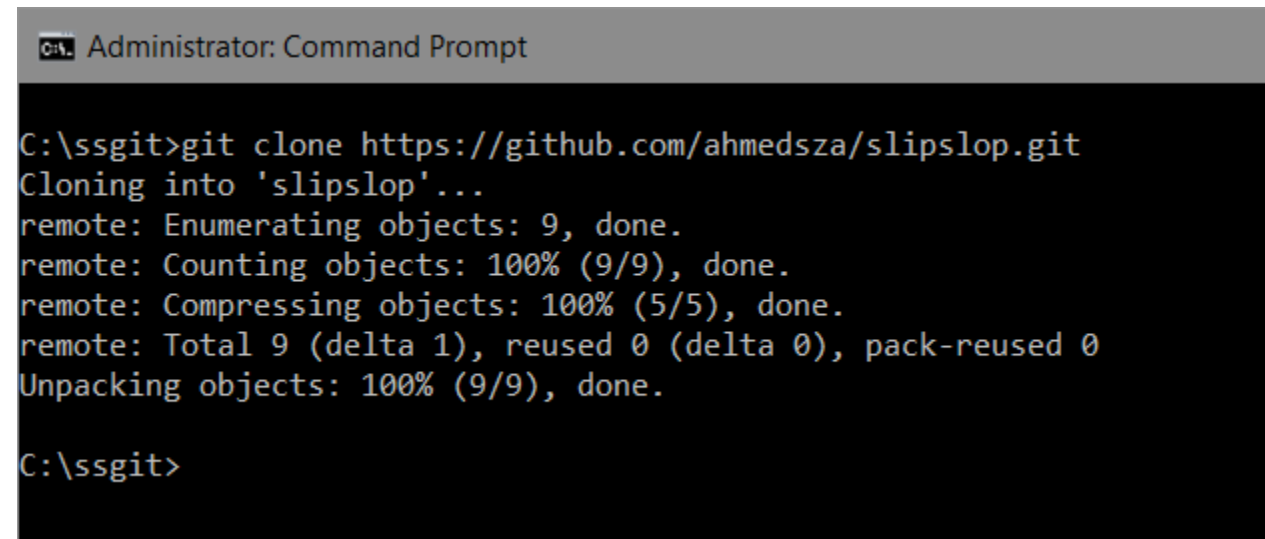
- In your command prompt window, type

```
git clone
```

- And then paste the URL (make sure there is a space). Press enter



```
C:\ssgit>git clone https://github.com/ahmedsza/slipslop.git
```



```
C:\ssgit>git clone https://github.com/ahmedsza/slipslop.git
Cloning into 'slipslop'...
remote: Enumerating objects: 9, done.
remote: Counting objects: 100% (9/9), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 9 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (9/9), done.

C:\ssgit>
```

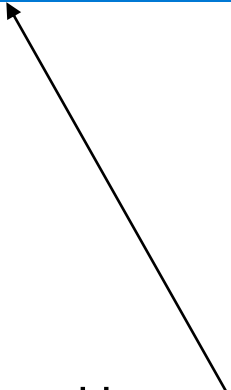
There should be a directory slipslop, check it out - C:\ssgit\slipslop

You might need this..

```
git config --global user.name "Your Name"
```

```
git config --global user.email "your_email@youremail.com"
```

Use email you signed up
with.



What if I add or make changes to the repository?

- What if you want to get updates
- Make sure you are in slipslop directory

```
cd slipslop
```

```
Administrator: Com...  
C:\ssgit>cd slipslop
```

- Run git pull

```
Git pull
```

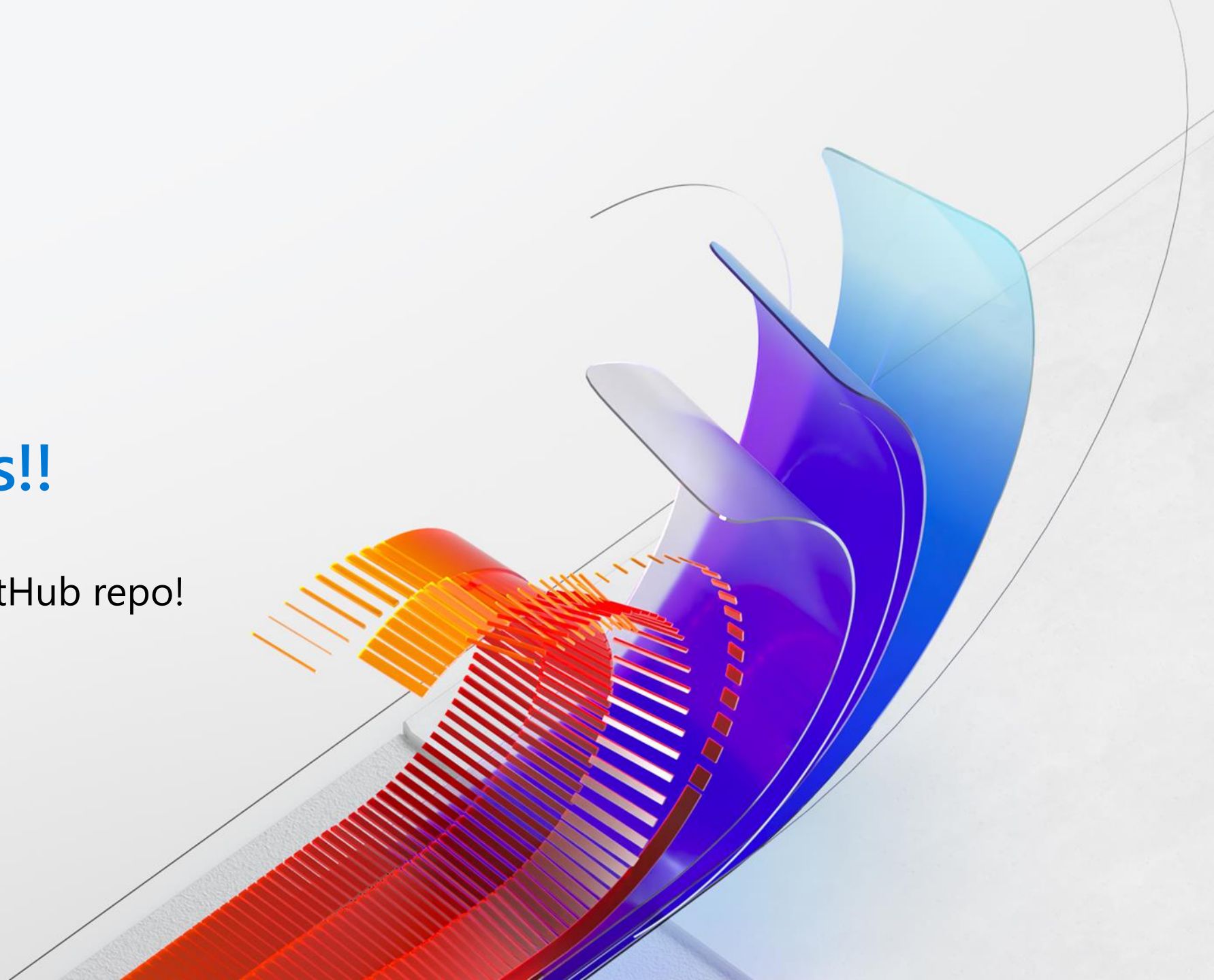
- Run dir

```
dir
```

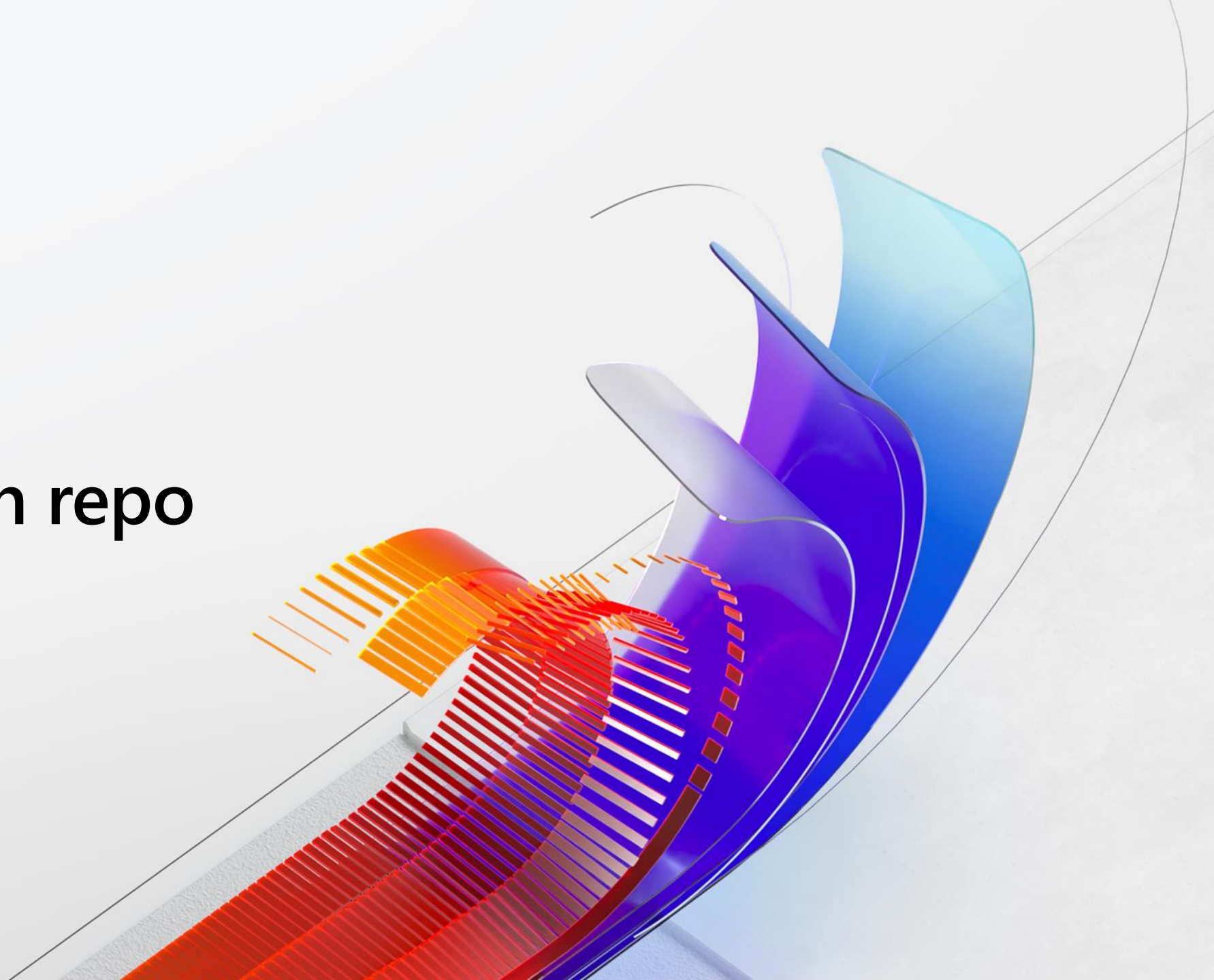
```
Administrator: Command Prompt  
C:\ssgit\slipslop>git pull  
remote: Enumerating objects: 4, done.  
remote: Counting objects: 100% (4/4), done.  
remote: Compressing objects: 100% (2/2), done.  
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0  
Unpacking objects: 100% (3/3), done.  
From https://github.com/ahmedsza/slipslop  
    0384896..748245c  master    -> origin/master  
Updating 0384896..748245c  
Fast-forward  
 file3.txt | 1 +  
1 file changed, 1 insertion(+)  
create mode 100644 file3.txt  
C:\ssgit\slipslop>
```

Congratulations!!

Try it out with another GitHub repo!



I want my own repo



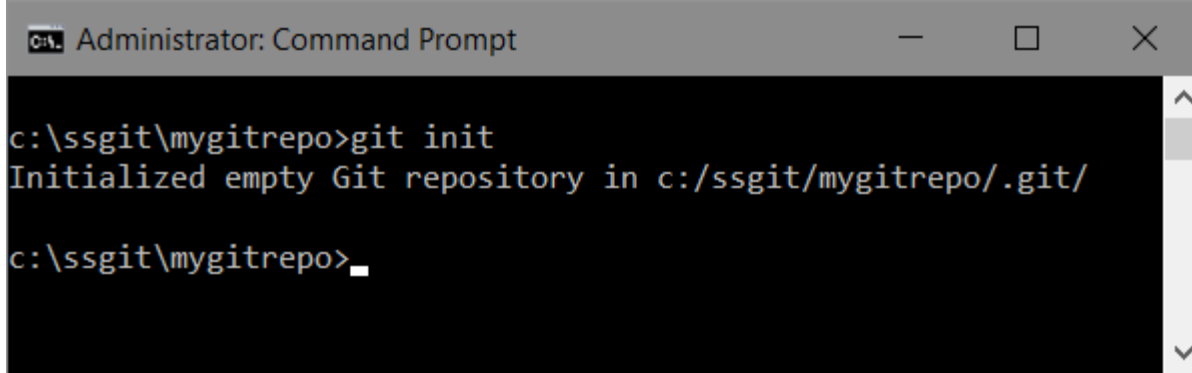
My own git repo

- Make sure we are in mygit repo directory

```
cd c:\ssgit\mygitrepo
```

- To make this a git repo

```
git init
```

A screenshot of a Windows Command Prompt window titled "Administrator: Command Prompt". The window has standard Windows window controls (minimize, maximize, close) in the title bar. The command prompt shows the following text:
c:\ssgit\mygitrepo>git init
Initialized empty Git repository in c:/ssgit/mygitrepo/.git/
c:\ssgit\mygitrepo>
The text is displayed in a monospaced font on a black background. A vertical scrollbar is visible on the right side of the command prompt window.

```
Administrator: Command Prompt  
c:\ssgit\mygitrepo>git init  
Initialized empty Git repository in c:/ssgit/mygitrepo/.git/  
c:\ssgit\mygitrepo>
```

- A hidden directory is created

```
dir /A:H
```

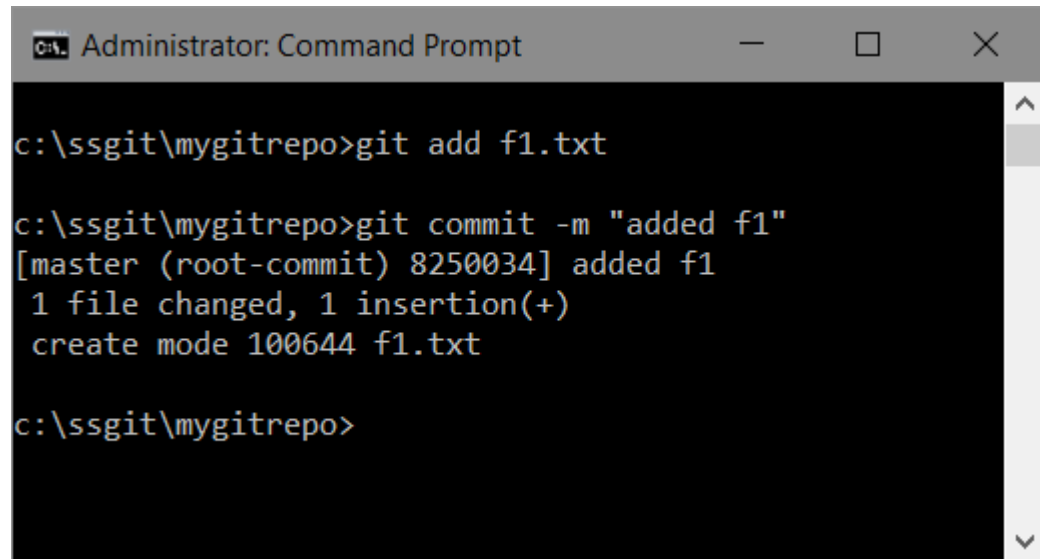
My own git repo

- Create a file in that directory, put some text – and save it

```
notepad f1.txt
```

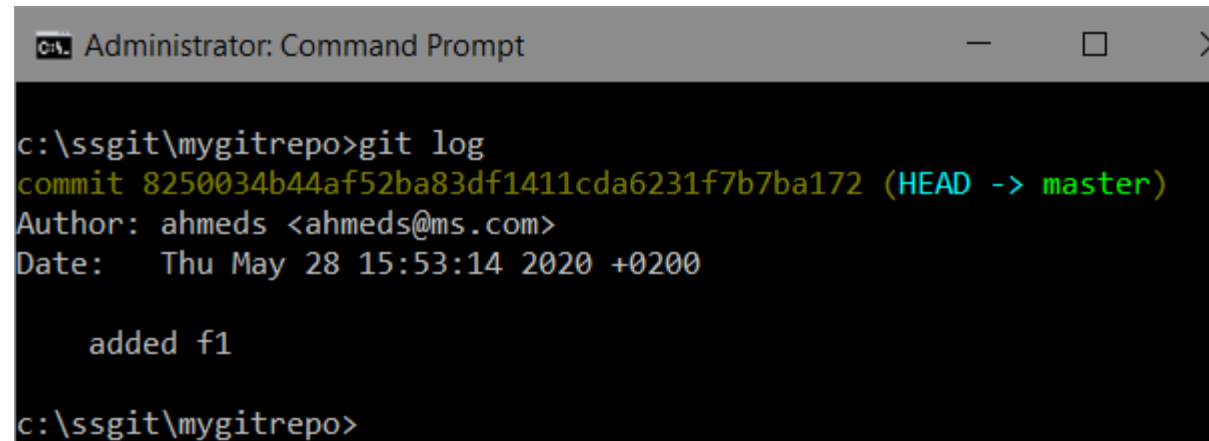
- Now we must let git know about it

```
git add f1.txt  
git commit -m "added my first git file, put whatever comment"
```



```
Administrator: Command Prompt  
c:\ssgit\mygitrepo>git add f1.txt  
  
c:\ssgit\mygitrepo>git commit -m "added f1"  
[master (root-commit) 8250034] added f1  
1 file changed, 1 insertion(+)  
create mode 100644 f1.txt  
  
c:\ssgit\mygitrepo>
```

```
git log
```



```
Administrator: Command Prompt  
  
c:\ssgit\mygitrepo>git log  
commit 8250034b44af52ba83df1411cda6231f7b7ba172 (HEAD -> master)  
Author: ahmeds <ahmeds@ms.com>  
Date: Thu May 28 15:53:14 2020 +0200  
  
    added f1  
  
c:\ssgit\mygitrepo>
```

Rinse and repeat

- Add another file, and commit it

```
echo "f2" >> f2.txt  
git add f2.txt  
git commit -m "added my second git file, put whatever comment"  
git log
```

Rinse and repeat

- Create a file in that directory, put some text – and save it
- One step at a time.

```
echo "f3" >> f3.txt  
notepad f1.txt  
git add .  
git commit -m "added f3 and changed f1"  
git log --name-status  
git log --oneline
```


Mini-advanced stuff



Mini advanced stuff

Back to the future..

- Do this one at a time

```
git log --oneline
dir
Git checkout <enter the number>
Dir
Git checkout master
```

```
Command Prompt

c:\ssgit\mygitrepo>git log --oneline
178e31a (HEAD -> mybranch, master) f2
8250034 added f1

c:\ssgit\mygitrepo>dir
Volume in drive C is OSDisk
Volume Serial Number is DCFC-90BB

Directory of c:\ssgit\mygitrepo

05/28/2020  04:04 PM    <DIR>          .
05/28/2020  04:04 PM    <DIR>          ..
05/28/2020  03:51 PM                6 f1.txt
05/28/2020  04:04 PM                4 f2.txt
                2 File(s)                10 bytes
                2 Dir(s)  22,847,279,104 bytes free

c:\ssgit\mygitrepo>git checkout 8250034
Note: checking out '8250034'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by performing another checkout.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -b with the checkout command again. Example:

    git checkout -b <new-branch-name>

HEAD is now at 8250034 added f1

c:\ssgit\mygitrepo>dir
Volume in drive C is OSDisk
Volume Serial Number is DCFC-90BB

Directory of c:\ssgit\mygitrepo

05/28/2020  04:07 PM    <DIR>          .
05/28/2020  04:07 PM    <DIR>          ..
05/28/2020  03:51 PM                6 f1.txt
                1 File(s)                6 bytes
                2 Dir(s)  22,847,127,552 bytes free

c:\ssgit\mygitrepo>
```

Mini advanced stuff

Branching

```
Command Prompt
c:\ssgit\mygitrepo>git branch mybranch

c:\ssgit\mygitrepo>git checkout mybranch
Switched to branch 'mybranch'

c:\ssgit\mygitrepo>
c:\ssgit\mygitrepo>
```

<one at a time>

git branch mybranch

git checkout mybranch

Notepad f3.txt (save it)

git add f3.txt

git commit -m "added f3 in branch"

Dir

git checkout master

Dir

```
Command Prompt
c:\ssgit\mygitrepo>notepad f3.txt

c:\ssgit\mygitrepo>git add f3.txt

c:\ssgit\mygitrepo>git commit -m "added f3 file in new branch"
[mybranch de2fed9] added f3 file in new branch
1 file changed, 1 insertion(+)
create mode 100644 f3.txt

c:\ssgit\mygitrepo>dir
Volume in drive C is OSDisk
Volume Serial Number is DCFC-90BB

Directory of c:\ssgit\mygitrepo

05/28/2020  04:11 PM    <DIR>          .
05/28/2020  04:11 PM    <DIR>          ..
05/28/2020  03:51 PM                6 f1.txt
05/28/2020  04:10 PM                4 f2.txt
05/28/2020  04:11 PM                2 f3.txt
               3 File(s)              12 bytes
               2 Dir(s)  22,841,503,744 bytes free

c:\ssgit\mygitrepo>git checkout master
Switched to branch 'master'

c:\ssgit\mygitrepo>dir
Volume in drive C is OSDisk
Volume Serial Number is DCFC-90BB

Directory of c:\ssgit\mygitrepo

05/28/2020  04:12 PM    <DIR>          .
05/28/2020  04:12 PM    <DIR>          ..
05/28/2020  03:51 PM                6 f1.txt
05/28/2020  04:10 PM                4 f2.txt
               2 File(s)              10 bytes
               2 Dir(s)  22,849,486,848 bytes free

c:\ssgit\mygitrepo>
```

Mini advanced

Merge your changes

```
git merge mybranch
Dir
git status
git log --oneline
```

Command Prompt

```
c:\ssgit\mygitrepo>git merge mybranch
```

```
Updating 178e31a..de2fed9
```

```
Fast-forward
```

```
f3.txt | 1 +
```

```
1 file changed, 1 insertion(+)
```

```
create mode 100644 f3.txt
```

```
c:\ssgit\mygitrepo>dir
```

```
Volume in drive C is OSDisk
```

```
Volume Serial Number is DCFC-90BB
```

```
Directory of c:\ssgit\mygitrepo
```

05/28/2020	04:13 PM	<DIR>	.
05/28/2020	04:13 PM	<DIR>	..
05/28/2020	03:51 PM		6 f1.txt
05/28/2020	04:10 PM		4 f2.txt
05/28/2020	04:13 PM		2 f3.txt
		3 File(s)	12 bytes
		2 Dir(s)	22,854,336,512 bytes free

```
c:\ssgit\mygitrepo>git status
```

```
On branch master
```

```
nothing to commit, working tree clean
```

```
c:\ssgit\mygitrepo>git log --one-line
```

```
fatal: unrecognized argument: --one-line
```

```
c:\ssgit\mygitrepo>git log --oneline
```

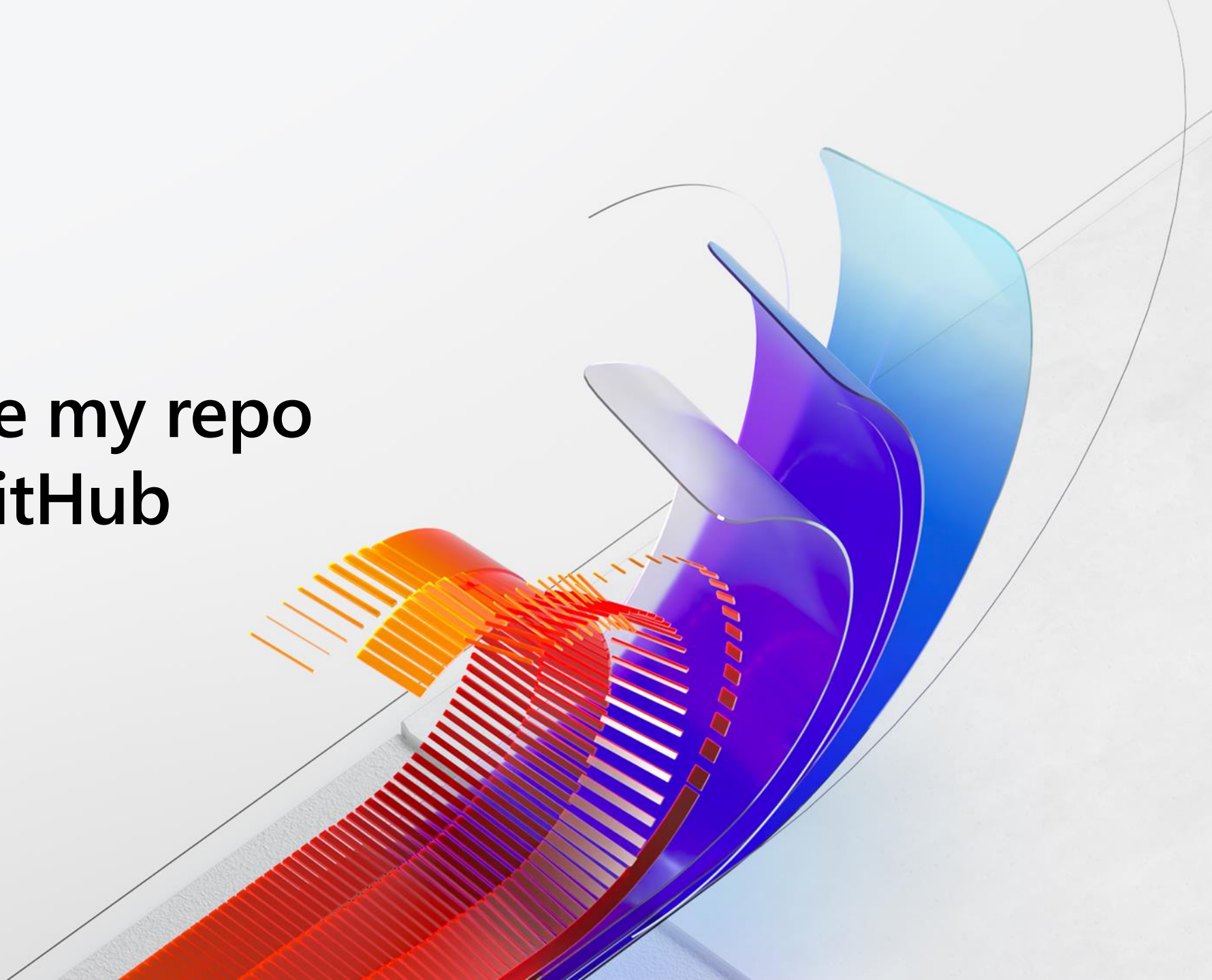
```
de2fed9 (HEAD -> master, mybranch) added f3 file in new branch
```

```
178e31a f2
```

```
8250034 added f1
```

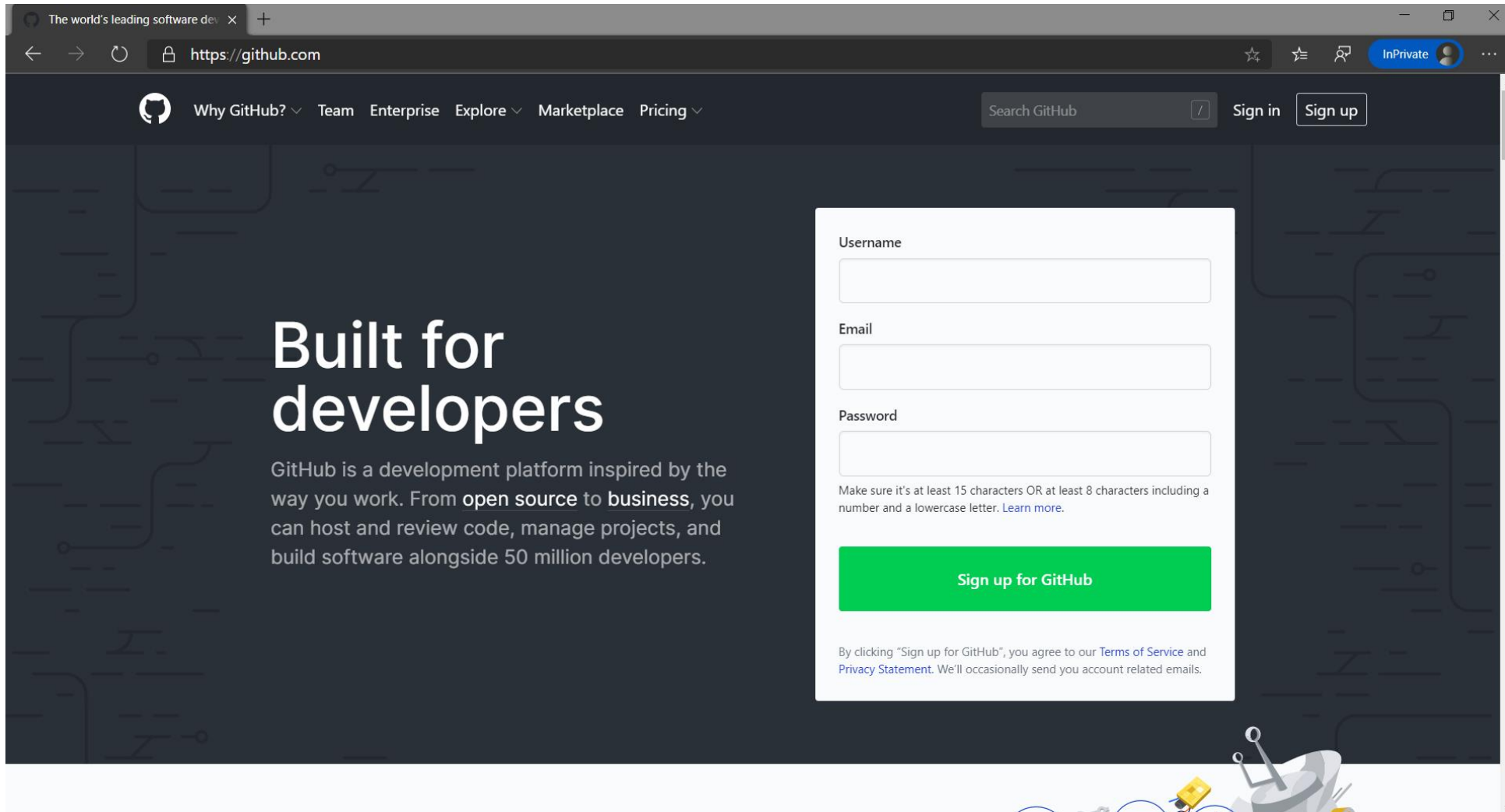
```
c:\ssgit\mygitrepo>_
```

I want to share my repo
aka where's GitHub



Sign up for GitHub

Visit www.github.com – create an account

A screenshot of the GitHub website's sign-up page. The browser's address bar shows 'https://github.com'. The page has a dark background with a faint circuit-like pattern. On the left, the text 'Built for developers' is prominently displayed, followed by a paragraph describing GitHub as a development platform. On the right, there is a white sign-up form with fields for 'Username', 'Email', and 'Password'. Below the password field is a note about password requirements and a link to 'Learn more'. A large green button labeled 'Sign up for GitHub' is positioned below the form. At the bottom of the form, there is a line of text stating that by signing up, the user agrees to the Terms of Service and Privacy Statement. The browser's top bar includes navigation icons, a search bar, and 'Sign in' and 'Sign up' buttons. The user's profile icon in the top right corner is labeled 'InPrivate'.

Create a repo

Overview

Repositories 45

Projects 0

Packages 2

Stars 0

Followers 0

Following 0

Find a repository...

Type: All ▾

Language: All ▾

 New

A repository contains all project files, including the revision history. Already have a project repository elsewhere?
[Import a repository.](#)

Repository template


Start your repository with a template repository's contents.

No template ▾

• f

Owner

Repository name *

 ahmedsza ▾

/

Great repository names are short and memorable. Need inspiration? How about [bookish-guide](#)?

Description (optional)



Public

Anyone can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.

Skip this step if you're importing an existing repository.

☐ **Initialize this repository with a README**

This will let you immediately clone the repository to your computer.

Add .gitignore: **None** ▾

Add a license: **None** ▾



Grant your Marketplace apps access to this repository

You are subscribed to 3 Marketplace apps



Dependabot Preview

Automated dependency updates for Ruby, JavaScript, Python, Go, PHP, Elixir, Rust, Java and .NET



Azure Pipelines

Continuously build, test, and deploy to any platform and cloud



WhiteSource Bolt

Detect open source vulnerabilities in real time with suggested fixes for quick remediation

Fill this in and
click create
repository

Push our files to GitHub

The screenshot shows the GitHub repository setup page for 'ahmedsza/mygitrepo'. The browser's address bar shows the URL 'https://github.com/ahmedsza/mygitrepo'. The page has a navigation bar with links for Code, Issues (0), Pull requests (0), Actions, Projects (0), Wiki, Security (0), Insights, and Settings. The main content area is divided into four sections: 'Quick setup — if you've done this kind of thing before', '...or create a new repository on the command line', '...or push an existing repository from the command line', and '...or import code from another repository'. The 'Quick setup' section includes a 'Set up in Desktop' button and a text input field with the URL 'https://github.com/ahmedsza/mygitrepo.git'. The '...or create a new repository on the command line' section contains a code block with the following commands:

```
echo "# mygitrepo" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/ahmedsza/mygitrepo.git
git push -u origin master
```

 The '...or push an existing repository from the command line' section contains a code block with the following commands:

```
git remote add origin https://github.com/ahmedsza/mygitrepo.git
git push -u origin master
```

 The '...or import code from another repository' section includes an 'Import code' button. A blue box with the text 'Copy this' and an arrow points to the code block in the '...or push an existing repository from the command line' section. The bottom of the page shows a taskbar with a PDF file 'Azure data Factory-....pdf' open.

Quick setup — if you've done this kind of thing before

[Set up in Desktop](#) or [HTTPS](#) [SSH](#)

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# mygitrepo" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/ahmedsza/mygitrepo.git
git push -u origin master
```

...or push an existing repository from the command line

```
git remote add origin https://github.com/ahmedsza/mygitrepo.git
git push -u origin master
```

...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

[Import code](#)

```
Developer Command Prompt for VS 2017

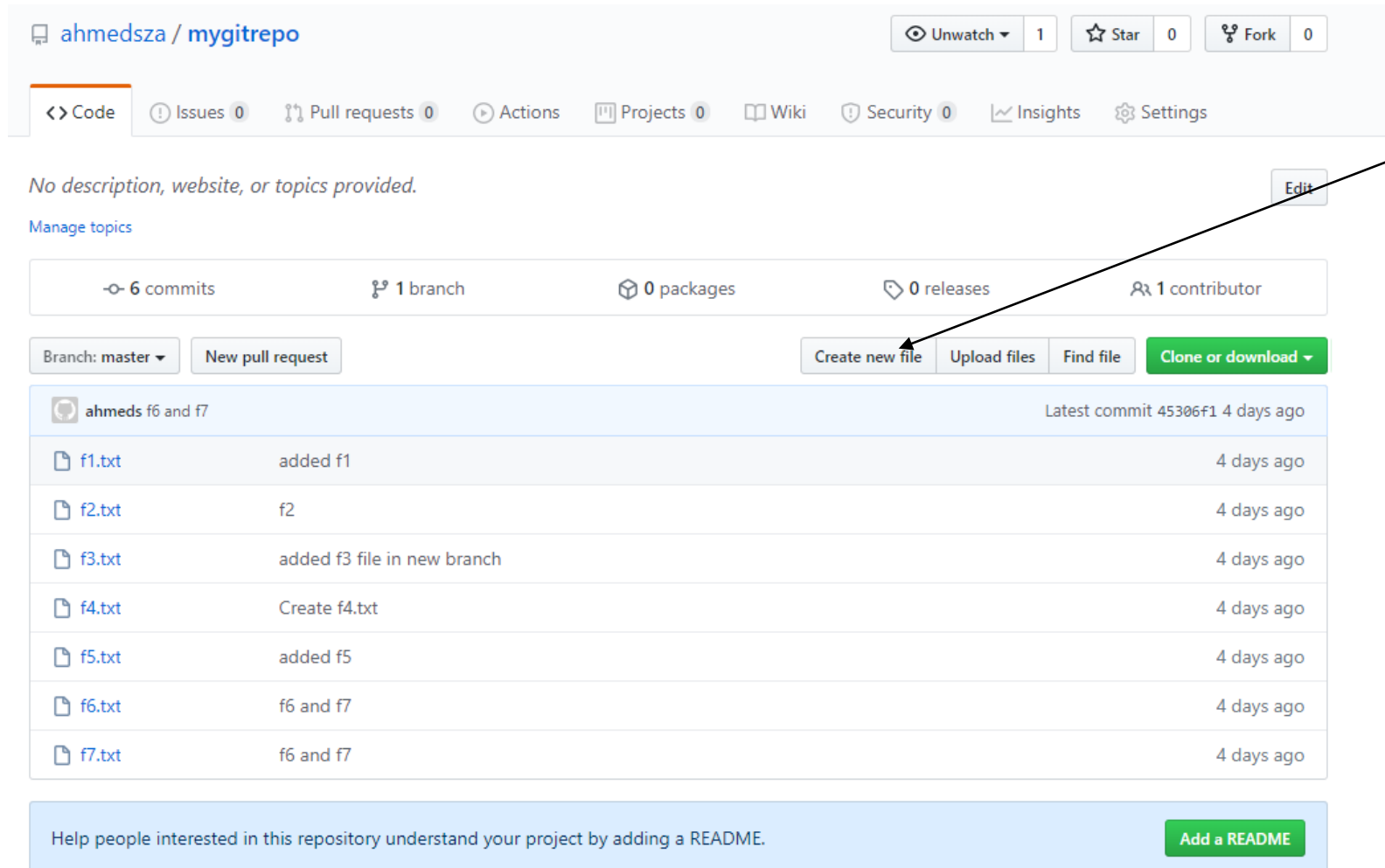
C:\ssgit\mygitrepo>git remote add origin https://github.com/ahmedsza/mygitrepo.git
```

```
Developer Command Prompt for VS 2017

C:\ssgit\mygitrepo>git push -u origin master
Everything up-to-date
Branch 'master' set up to track remote branch 'master' from 'origin'.

C:\ssgit\mygitrepo>
```

Make some changes in the browser



The screenshot shows the GitHub interface for the repository 'ahmedsza / mygitrepo'. At the top, there are buttons for 'Unwatch' (1), 'Star' (0), and 'Fork' (0). Below this is a navigation bar with links for 'Code', 'Issues' (0), 'Pull requests' (0), 'Actions', 'Projects' (0), 'Wiki', 'Security' (0), 'Insights', and 'Settings'. The repository description states 'No description, website, or topics provided.' with an 'Edit' button. Below the description, statistics show '6 commits', '1 branch', '0 packages', '0 releases', and '1 contributor'. A row of buttons includes 'Branch: master', 'New pull request', 'Create new file' (highlighted by an arrow), 'Upload files', 'Find file', and 'Clone or download'. The commit history table shows the following entries:

File	Commit Message	Time
f1.txt	added f1	4 days ago
f2.txt	f2	4 days ago
f3.txt	added f3 file in new branch	4 days ago
f4.txt	Create f4.txt	4 days ago
f5.txt	added f5	4 days ago
f6.txt	f6 and f7	4 days ago
f7.txt	f6 and f7	4 days ago

At the bottom, a blue banner encourages adding a README with an 'Add a README' button.


mygitrepo / anewfile.txt Cancel

<> Edit new file

Preview

Spaces 2 No wrap

1 some wonderful comment



Commit new file

Create anewfile.txt

Add an optional extended description...

☒ Commit directly to the master branch.
☐ Create a new branch for this commit and start a pull request. [Learn more about pull requests.](#)

Commit new file Cancel

Give a filename

Add some contents

Commit (this is "add and commit" from browser)

Get the changes locally

Git pull

```
Developer Command Prompt for VS 2017

C:\ssgit\mygitrepo>git pull
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), done.
From https://github.com/ahmedsza/mygitrepo
   45306f1..ae92998  master    -> origin/master
Updating 45306f1..ae92998
Fast-forward
  anewfile.txt | 1 +
  1 file changed, 1 insertion(+)
  create mode 100644 anewfile.txt
```

No description, website, or topics provided.

Edit

Manage topics

8 commits

1 branch

0 packages

0 releases

1 contributor

Branch: master ▾

New pull request

Create new file

Upload files

Find file

Clone or download ▾

ahmedsza Update f1.txt		Latest commit df02db0 4 minutes ago
anewfile.txt	Create anewfile.txt	7 minutes ago
f1.txt	Update f1.txt	4 minutes ago
f2.txt	f2	4 days ago
f3.txt	added f3 file in new branch	4 days ago
f4.txt	Create f4.txt	4 days ago
f5.txt	added f5	4 days ago
f6.txt	f6 and f7	4 days ago
f7.txt	f6 and f7	4 days ago

Click on commits

Click on a filename.

ahmedsza Update f1.txt

df02db0 6 minutes ago

1 contributor

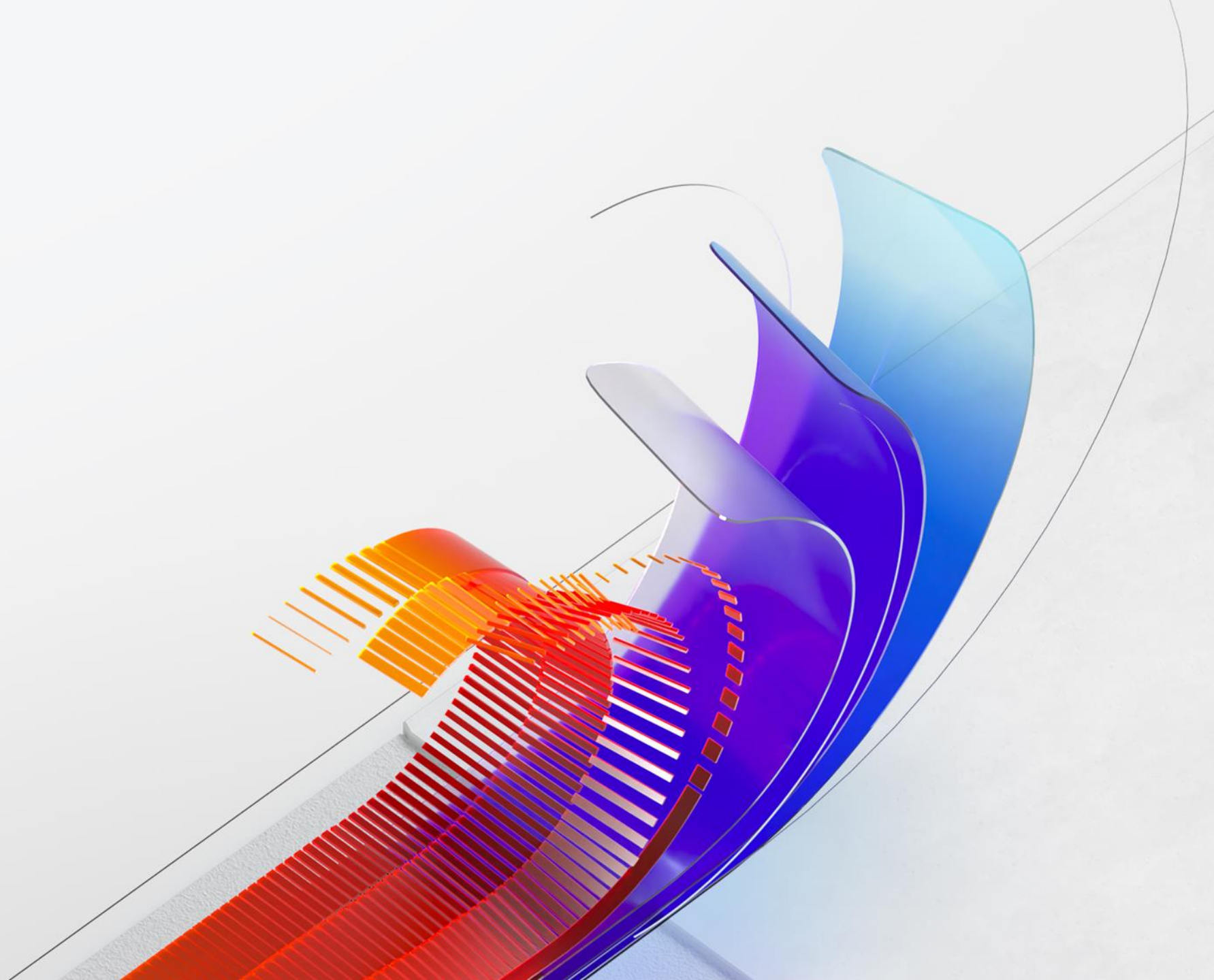
3 lines (2 sloc) | 21 Bytes

RawBlameHistory

```
1 fsafsa
2
3 cghange nade
```

“Play” with these options

Collaboration



Add collaborators

The screenshot shows the GitHub repository settings page for 'ahmedsza / mygitrepo'. The top navigation bar includes links for Code, Issues (0), Pull requests (0), Actions, Projects (0), Wiki, Security (0), Insights, and Settings. The 'Settings' tab is selected. On the left sidebar, the 'Manage access' option is highlighted. The main content area is titled 'Who has access' and shows two sections: 'PUBLIC REPOSITORY' (This repository is public and visible to anyone. Manage) and 'DIRECT ACCESS' (1 has access to this repository. 1 invitation). Below this is the 'Manage access' section, which includes a 'Select all' checkbox, a search bar 'Find a collaborator...', and a list of collaborators. One collaborator, 'ahms86', is listed with the status 'Awaiting ahms86's response' and a 'Pending Invite' label. A green button 'Invite a collaborator' is visible above the collaborator list. At the bottom, there are 'Previous' and 'Next' buttons.

ahmedsza / mygitrepo

Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Actions Projects 0 Wiki Security 0 Insights Settings

Options

Manage access

Branches

Webhooks

Notifications

Integrations

Deploy keys

Secrets

Actions

Moderation

Interaction limits

Who has access

PUBLIC REPOSITORY

This repository is public and visible to anyone.

[Manage](#)

DIRECT ACCESS

1 has access to this repository. 1 [invitation](#).

Manage access

☒ Select all Type ▾

Find a collaborator...

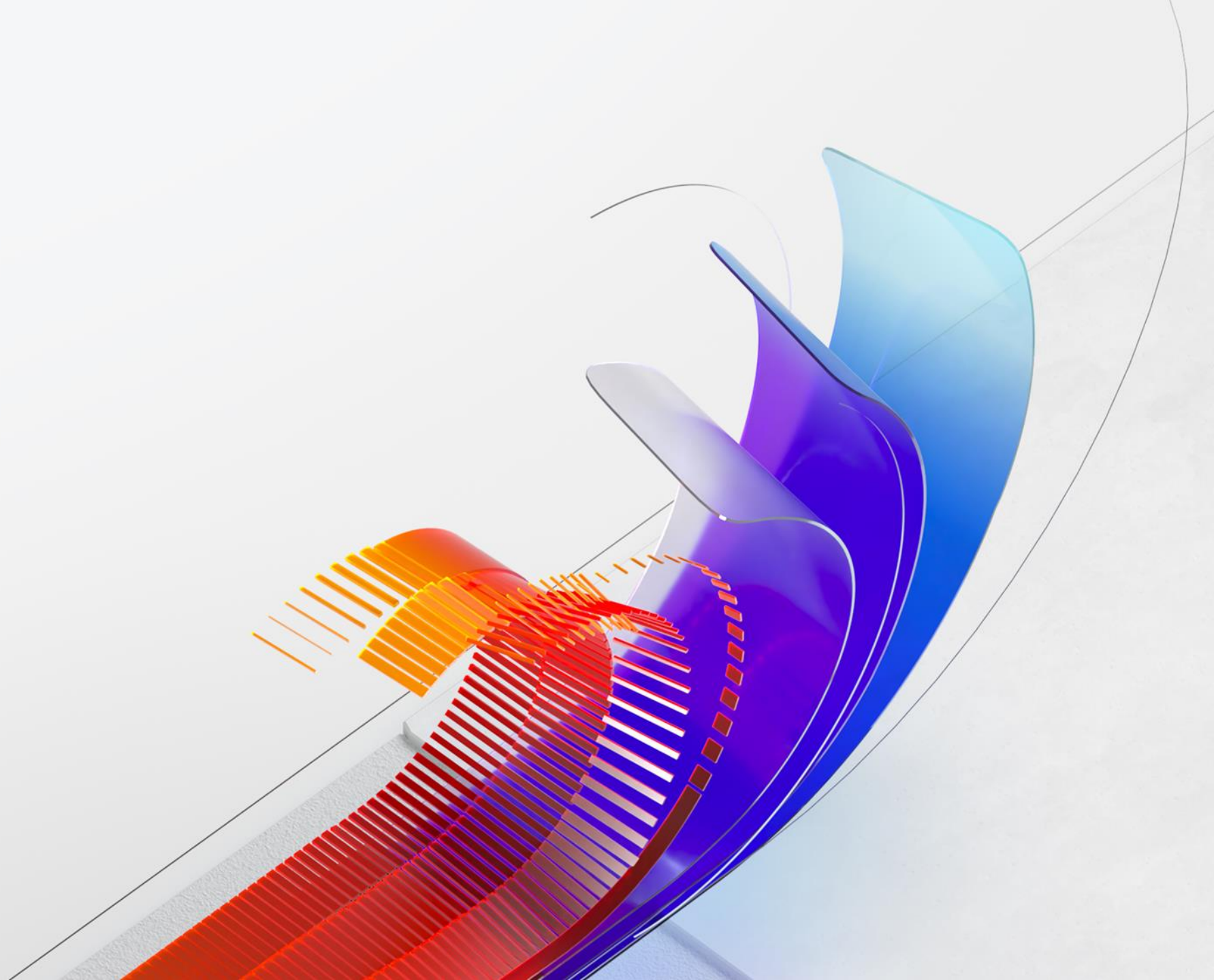
<input checked="" type="checkbox"/>	ahms86 Awaiting ahms86's response	Pending Invite	
-------------------------------------	---	----------------	--

[Previous](#) [Next](#)

Click settings

Invite someone. You need their github id

Pull request



Options
Manage access
Branches
Webhooks
Notifications
Integrations
Deploy keys
Secrets
Actions
Moderation
Interaction limits

Default branch

The default branch is considered the “base” branch in your repository, against which all pull requests and code commits are automatically made, unless you specify a different branch.

The default branch is set to master. To change this setting, [add another branch](#).

Branch protection rules

Define branch protection rules to disable force pushing, prevent branches from being deleted, and optionally require status checks before merging. New to branch protection rules? [Learn more](#).

master

Currently applies to 1 branch

EditDelete

Branch protection rule

Branch name pattern

master

Applies to 1 branch

master

Protect matching branches

☒ **Require pull request reviews before merging**

When enabled, all commits must be made to a non-protected branch and submitted via a pull request with the required number of approving reviews and no changes requested before it can be merged into a branch that matches this rule.

Required approving reviews: 1

☐ **Dismiss stale pull request approvals when new commits are pushed**

New reviewable commits pushed to a matching branch will dismiss pull request review approvals.

☐ **Require review from Code Owners**

Require an approved review in pull requests including files with a designated code owner.

☐ **Require status checks to pass before merging**

Choose which [status checks](#) must pass before branches can be merged into a branch that matches this rule. When enabled, commits must first be pushed to another branch, then merged or pushed directly to a branch that matches this rule after status checks have passed.

Add a file

mygitrepo /

< > Edit new file

Preview

Spaces 2 No wrap

1



Commit new file

Create new file

Add an optional extended description...

- ☐ You can't commit to `master` because it is a [protected branch](#).
- ☒ Create a new branch for this commit and start a pull request. [Learn more about pull requests.](#)

Commit new file

Cancel



Open a pull request

[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#)

Open a pull request

The change you just made was written to a new branch named `ahms86-patch-1`. Create a pull request below to propose these changes.

base: `master` ← compare: `ahms86-patch-1` ✓ **Able to merge.** These branches can be automatically merged.

Create ahms86file.txt

Write Preview

H B I

≡ <> 🔗

≡ ≡ ☑ @ ↻ ↶

Leave a comment

Attach files by dragging & dropping, selecting or pasting them.

Create pull request

Reviewers

No reviews—at least 1 approving review is required.

Assignees

No one—assign yourself

Labels

None yet

Projects

None yet

Milestone

No milestone

Linked issues

Use [Closing keywords](#) in the description to automatically close issues

Remember, contributions to this repository should follow our [GitHub Community Guidelines](#).

Create ahms86file.txt #1 [Edit](#) [Open with](#)

[Open](#) ahms86 wants to merge 1 commit into `master` from `ahms86-patch-1`

Conversation

Commits

Checks

Files changed

+1 -0

ahms86 commented now

No description provided.

Collaborator

Create ahms86file.txt

Verified

913bfe9

Add more commits by pushing to the `ahms86-patch-1` branch on [ahmedsza/mygitrepo](#).

Review required

At least 1 approving review is required by reviewers with write access. [Learn more](#).

Merging is blocked

Merging can be performed automatically with 1 approving review.

Merge pull request

You can also [open this in GitHub Desktop](#) or view [command line instructions](#).

Write Preview

H B I

≡ <> 🔗

≡ ≡ ☑ @ ↻ ↶

Leave a comment

Attach files by dragging & dropping, selecting or pasting them.

Close pull request

Comment

Reviewers

No reviews—at least 1 approving review is required.

Still in progress? Convert to draft

Assignees

No one—assign yourself

Labels

None yet

Projects

None yet

Milestone

No milestone

Linked issues

Successfully merging this pull request may close these issues.

None yet

Notifications

Customize

Unsubscribe

You're receiving notifications because you authored the thread.

Review the pull request

ahmedsza / mygitrepo

Unwatch 1 Star 0 Fork 0

<> Code Issues 0 Pull requests 1 Actions Projects 0 Wiki Security 0 Insights Settings

Label issues and pull requests for new contributors

Now, GitHub will help potential first-time contributors [discover issues](#) labeled with `good first issue`

[Dismiss](#)

Filters is:pr is:open

Labels 9 Milestones 0

1 Open 0 Closed

Author Label Projects Milestones Reviews Assignee

Create ahms86file.txt
#1 opened 1 minute ago by ahms86 • Review required

ProTip! Exclude everything labeled bug with `-label:bug`.

0 / 1 files viewed



Review changes

Write Preview

H B I

Leave a comment

Attach files by dragging & dropping, selecting or pasting them.



☒ **Comment**

Submit general feedback without explicit approval.

☐ **Approve**

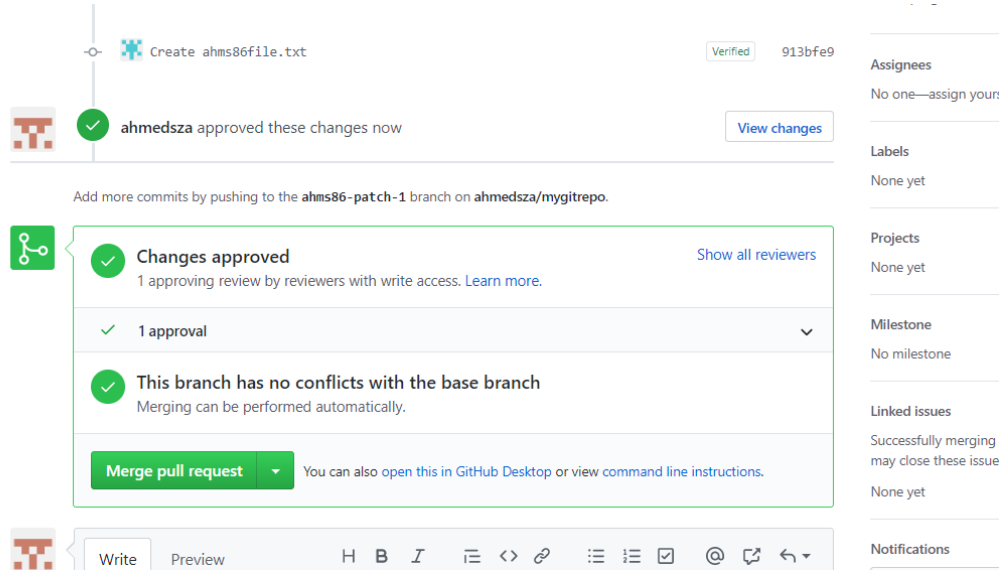
Submit feedback and approve merging these changes.

☐ **Request changes**

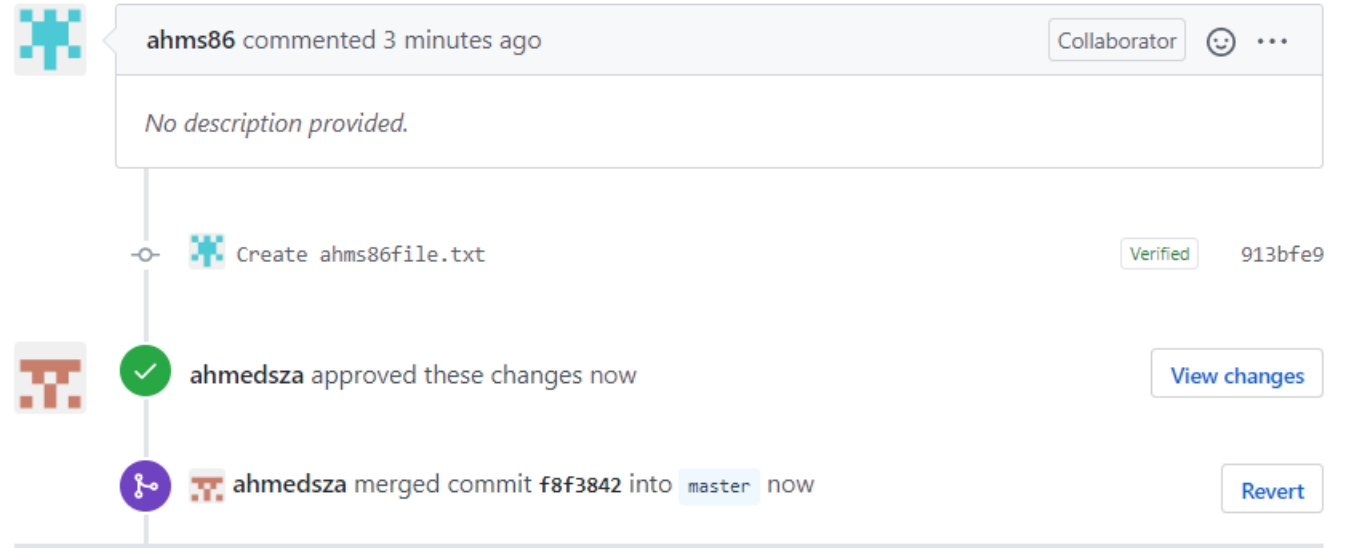
Submit feedback that must be addressed before merging.

Submit review

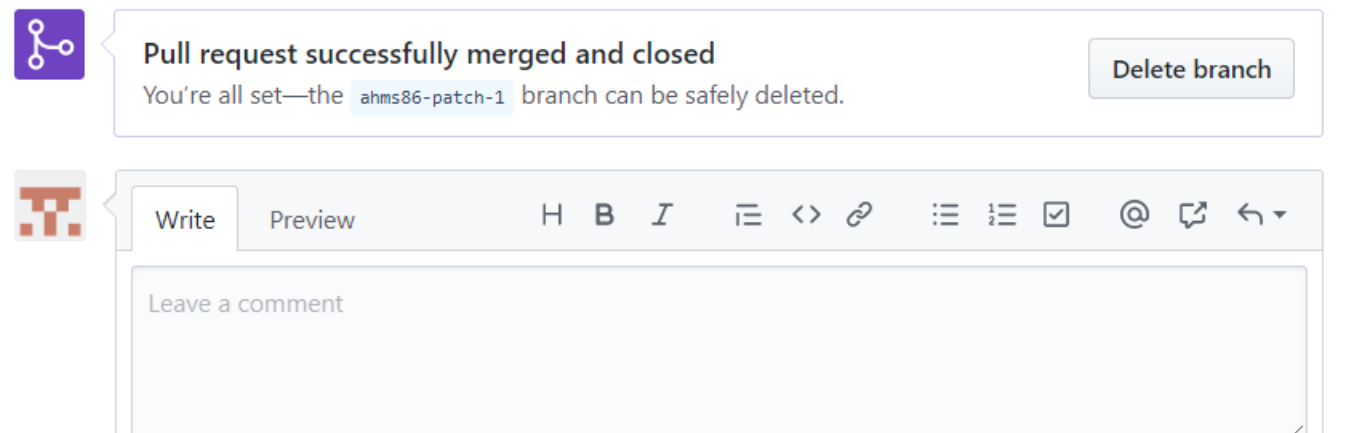
Merge it



This screenshot shows a GitHub pull request interface. At the top, a commit titled "Create ahms86file.txt" is shown with a "Verified" badge and the hash "913bfe9". Below the commit, a review by "ahmedsza" is displayed, indicating that the changes have been approved. A "View changes" button is visible next to the review. On the right side, there are sections for "Assignees" (No one—assign your...), "Labels" (None yet), "Projects" (None yet), "Milestone" (No milestone), "Linked issues" (Successfully merging may close these issue), and "Notifications". At the bottom, there is a "Merge pull request" button and a "Write" button.



This screenshot shows a GitHub pull request interface. At the top, a commit titled "Create ahms86file.txt" is shown with a "Verified" badge and the hash "913bfe9". Below the commit, a review by "ahmedsza" is displayed, indicating that the changes have been approved. A "View changes" button is visible next to the review. On the right side, there are sections for "Assignees" (No one—assign your...), "Labels" (None yet), "Projects" (None yet), "Milestone" (No milestone), "Linked issues" (Successfully merging may close these issue), and "Notifications". At the bottom, there is a "Merge pull request" button and a "Write" button.




This screenshot shows a GitHub pull request interface. At the top, a commit titled "Create ahms86file.txt" is shown with a "Verified" badge and the hash "913bfe9". Below the commit, a review by "ahmedsza" is displayed, indicating that the changes have been approved. A "View changes" button is visible next to the review. On the right side, there are sections for "Assignees" (No one—assign your...), "Labels" (None yet), "Projects" (None yet), "Milestone" (No milestone), "Linked issues" (Successfully merging may close these issue), and "Notifications". At the bottom, there is a "Merge pull request" button and a "Write" button.

Branch: master ▾

Commits on Jun 2, 2020

Merge pull request #1 from ahmedsza/ahms86-patch-1 ...

 ahmedsza committed 3 minutes ago


Verified



f8f3842



Create ahms86file.txt

 ahms86 committed 6 minutes ago

Verified




913bfe9



Commits on Jun 1, 2020

Create filebyahms86.txt

 ahms86 committed yesterday


Verified



af3e18d



Update f1.txt

 ahmedsza committed yesterday

Verified



df02db0



Create anewfile.txt

 ahmedsza committed yesterday

Verified




ae92998



Commits on May 28, 2020

f6 and f7


 ahmeds committed 5 days ago



45306f1



added f5


 ahmeds committed 5 days ago



b8e496e



Create f4.txt

 ahmedsza committed 5 days ago


Verified



f86507b



added f3 file in new branch

 ahmeds committed 5 days ago



de2fed9



f2



478a71a

