Mark McFadden
https://m2web.github.io/
https://github.com/m2web

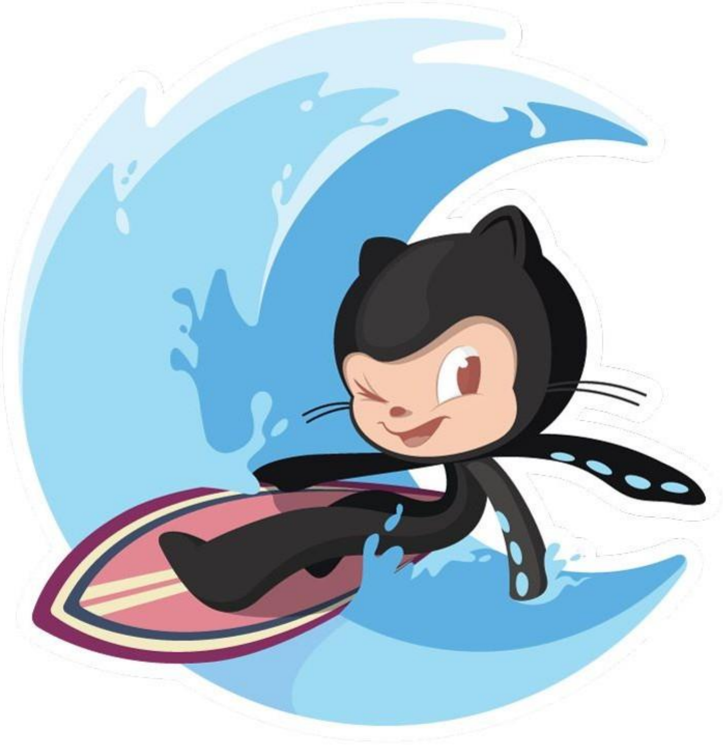
Introduction to
Git

Mark McFadden
https://m2web.github.io/
https://github.com/m2web

Presentation Flow

- 🐣 **My first git:** What is Git?

- 🕶️ **git is good:** Git Fundamentals 101



So, What is Git?

A Distributed **Version Control System.**

Any project which uses Git will have a **.git** folder which stores all the history of the project.

📖 **History:**

Know exactly which files changed, who made those changes, and when those changes occurred.

💾 **Backup:**

Ability to have different versions of the code in different places.

Git is Good!

→

The fundamentals of Git 101.

What is a repository?

A repository is a container that houses your project and its’ history. If your project folder contains the **“.git”** folder, then you are working with a repository!

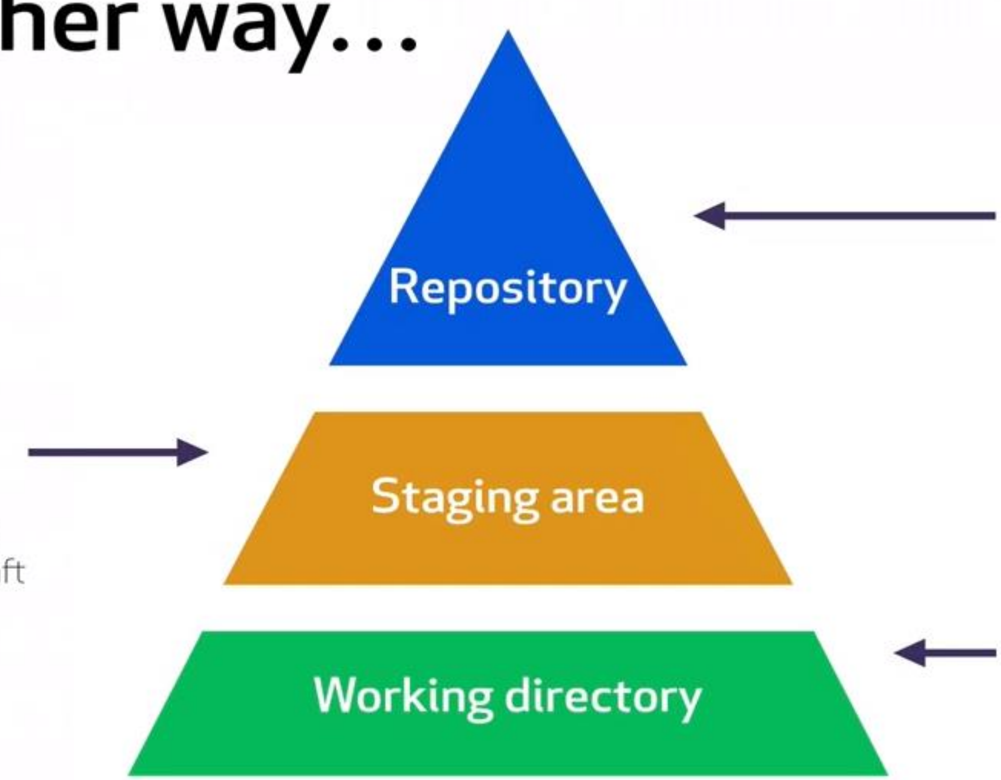
Git is like a desk

Working directory
where you write

Staging area
rough draft, in a
manila folder

Repository
final draft
in the filing cabinet

Put another way…



Repository

Staging area

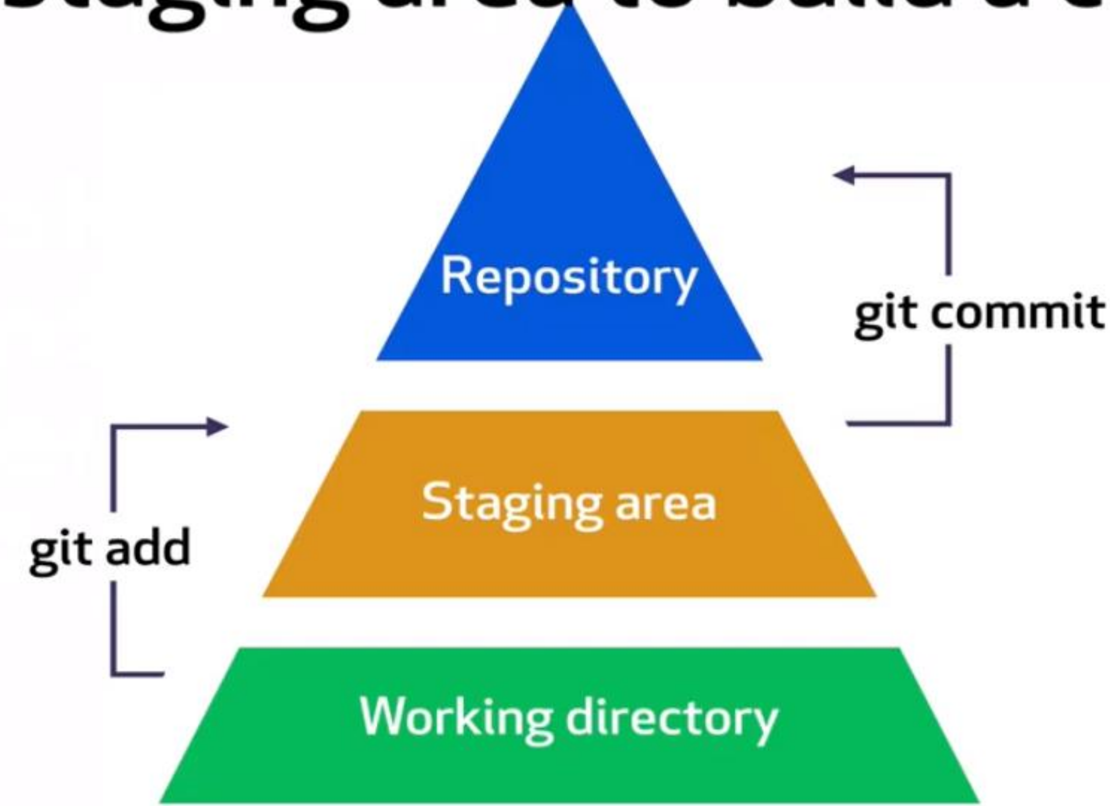
Working directory

Repository
Final draft,
permanent

Staging area
Your rough draft

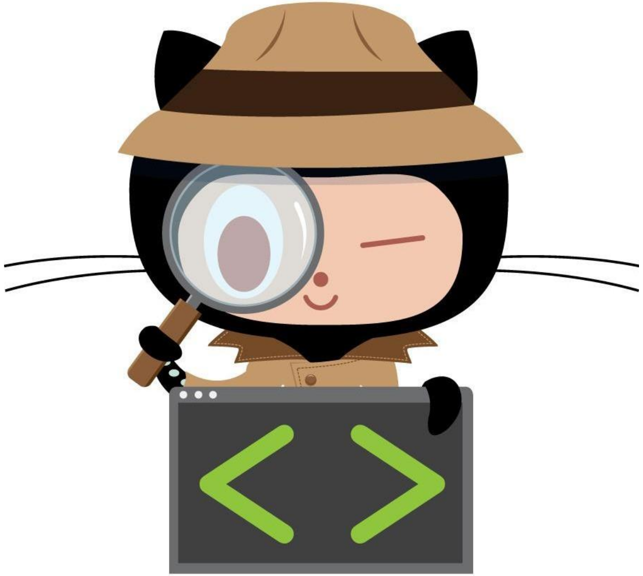
Working directory
Your workspace,
local on your machine

Use the staging area to build a commit



Before we start...

- ✓ Git Installed - https://git-scm.com/downloads
- ✓ Your favorite editor

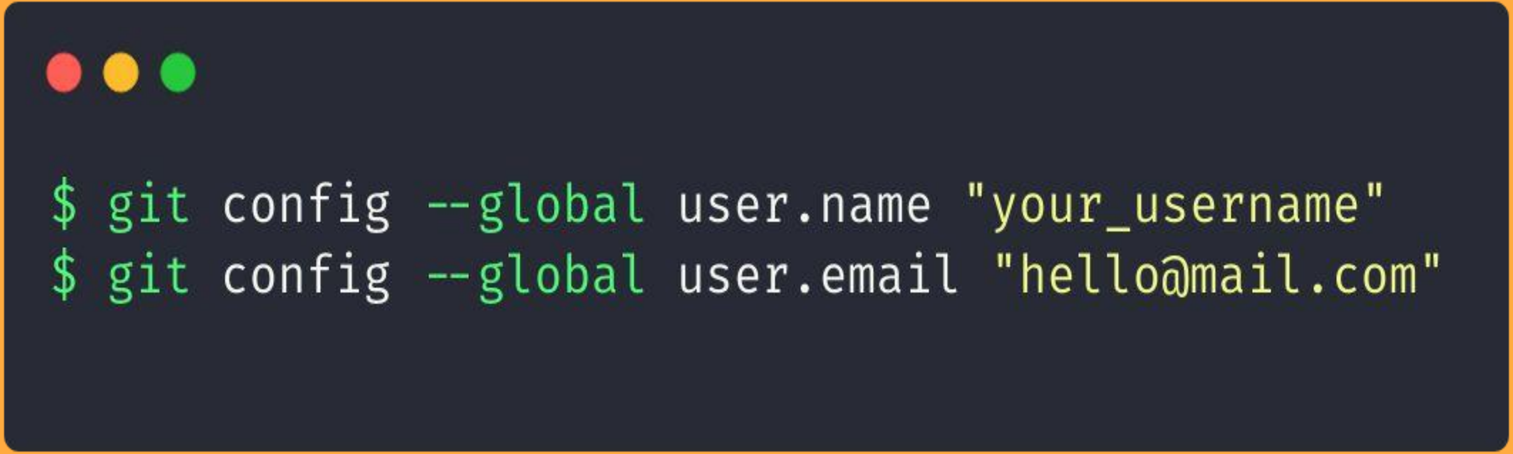


Let’s get started!

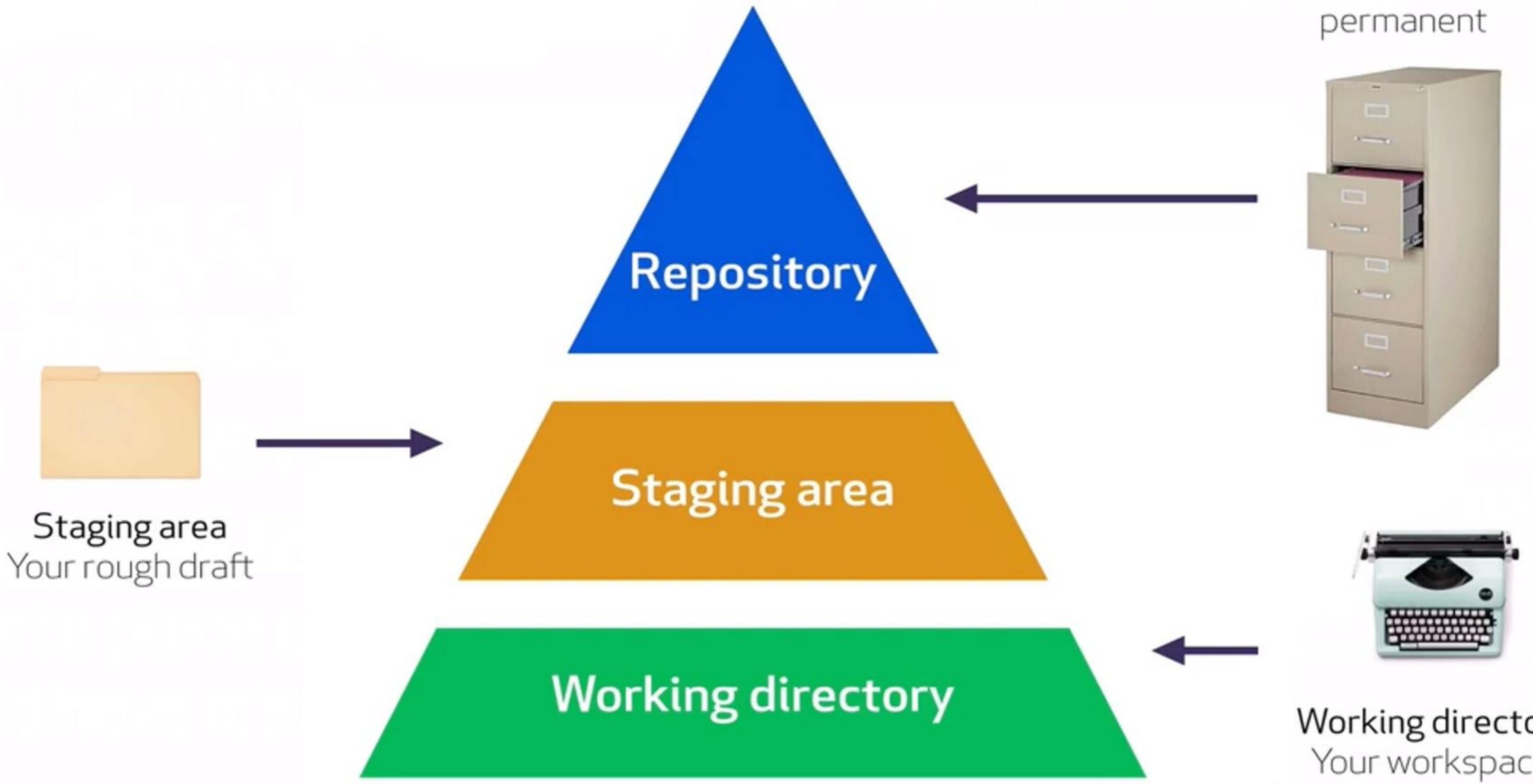
If you are on windows,open Git-Bash.

If you are on linux or Mac, fire up a terminal.

Let Git Know About You



```
$ git config --global user.name "your_username"
$ git config --global user.email "hello@mail.com"
```



Repository
Final draft,
permanent

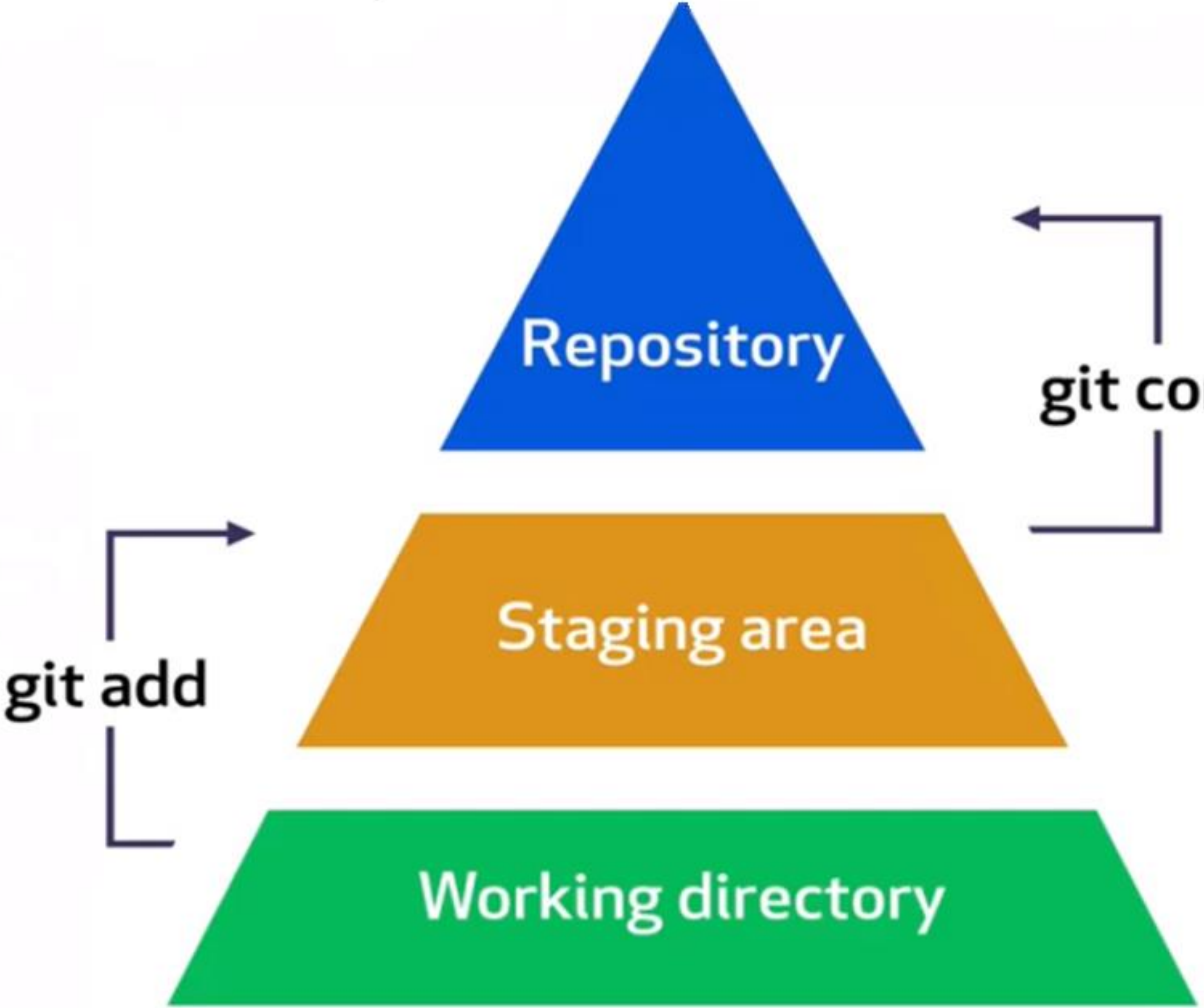
Repository

Staging area

Working directory

Staging area
Your rough draft

Working directory
Your workspace,
local on your machine


Repository
git commit
Staging area
git add
Working directory

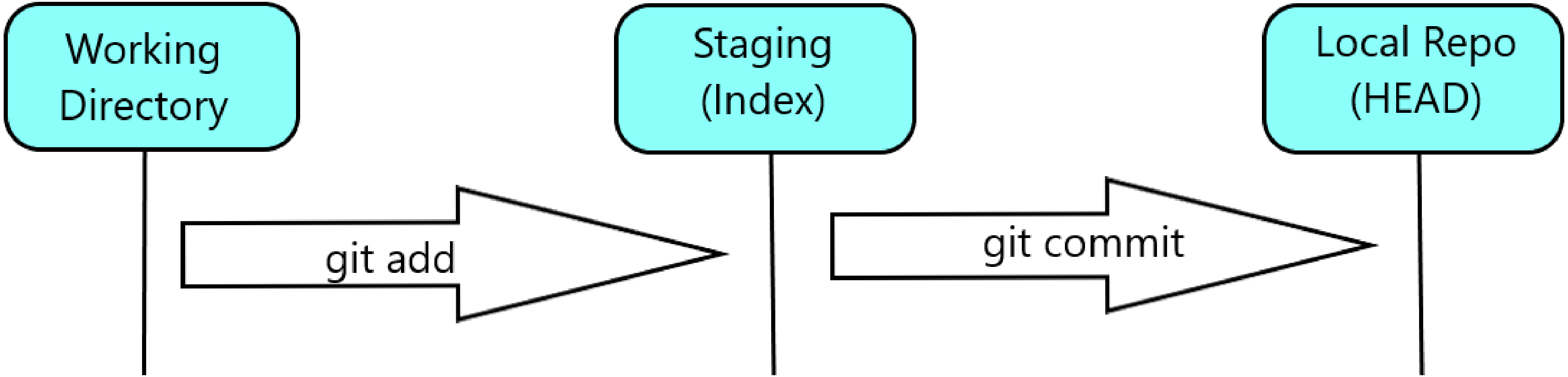
Git Flow Elements

Working Directory

Staging (Index)

Local Repo (HEAD)

Git Flow Elements

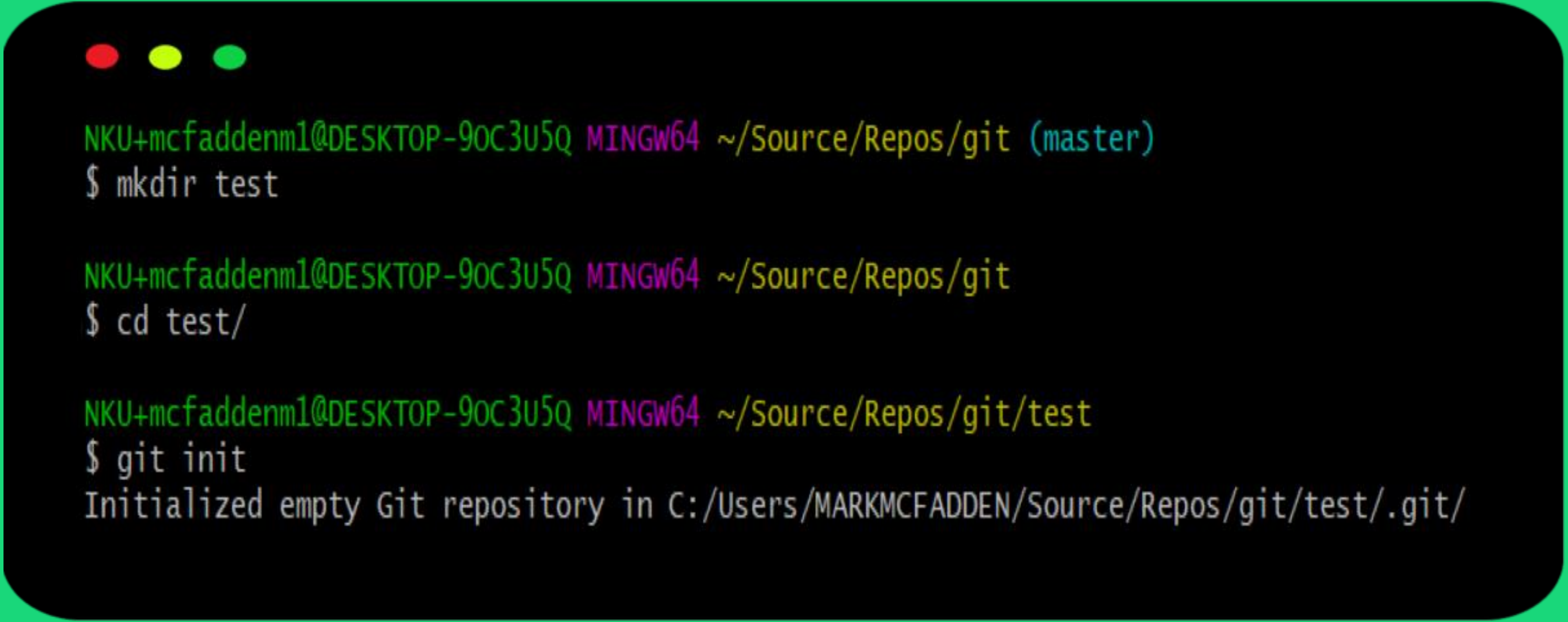


`git add` is a command used to add a file that is in the working directory to the staging area.

`git commit` is a command used to add all files that are staged to the local repository.

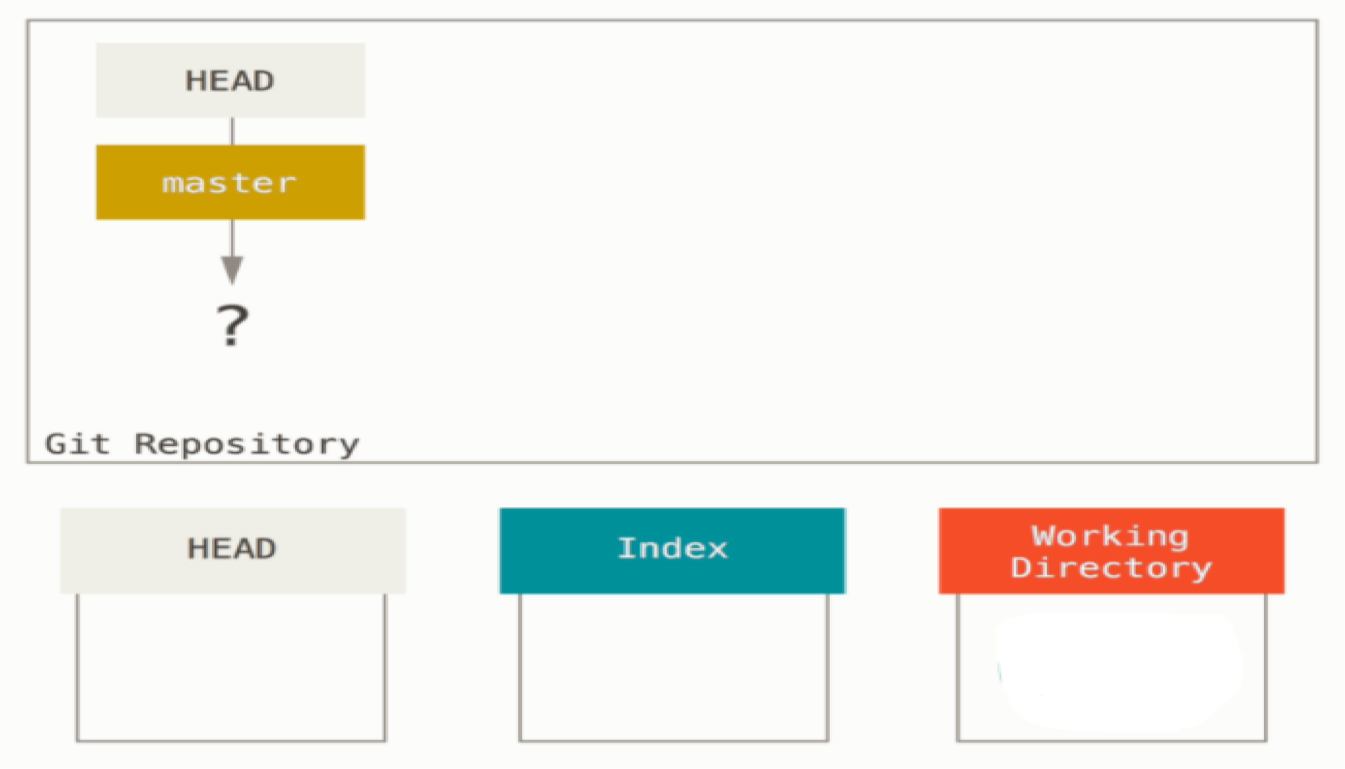
Your First Repo!

Initializing a new repository

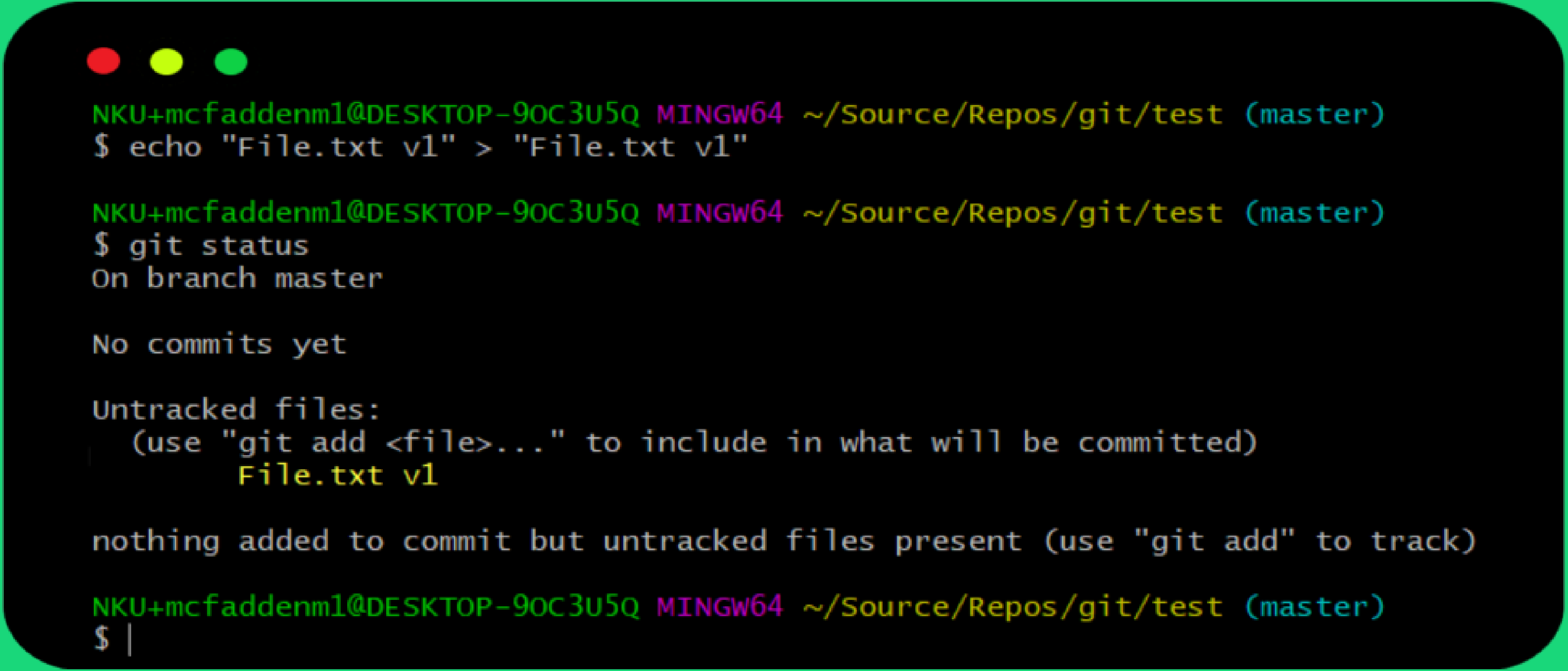


```
NKU+mcfaddenm1@DESKTOP-90C3U5Q MINGW64 ~/Source/Repos/git (master)
$ mkdir test

NKU+mcfaddenm1@DESKTOP-90C3U5Q MINGW64 ~/Source/Repos/git
$ cd test/

NKU+mcfaddenm1@DESKTOP-90C3U5Q MINGW64 ~/Source/Repos/git/test
$ git init
Initialized empty Git repository in C:/Users/MARKMCFADDEN/Source/Repos/git/test/.git/
```

Git Repo & Flow Elements State After Initialization

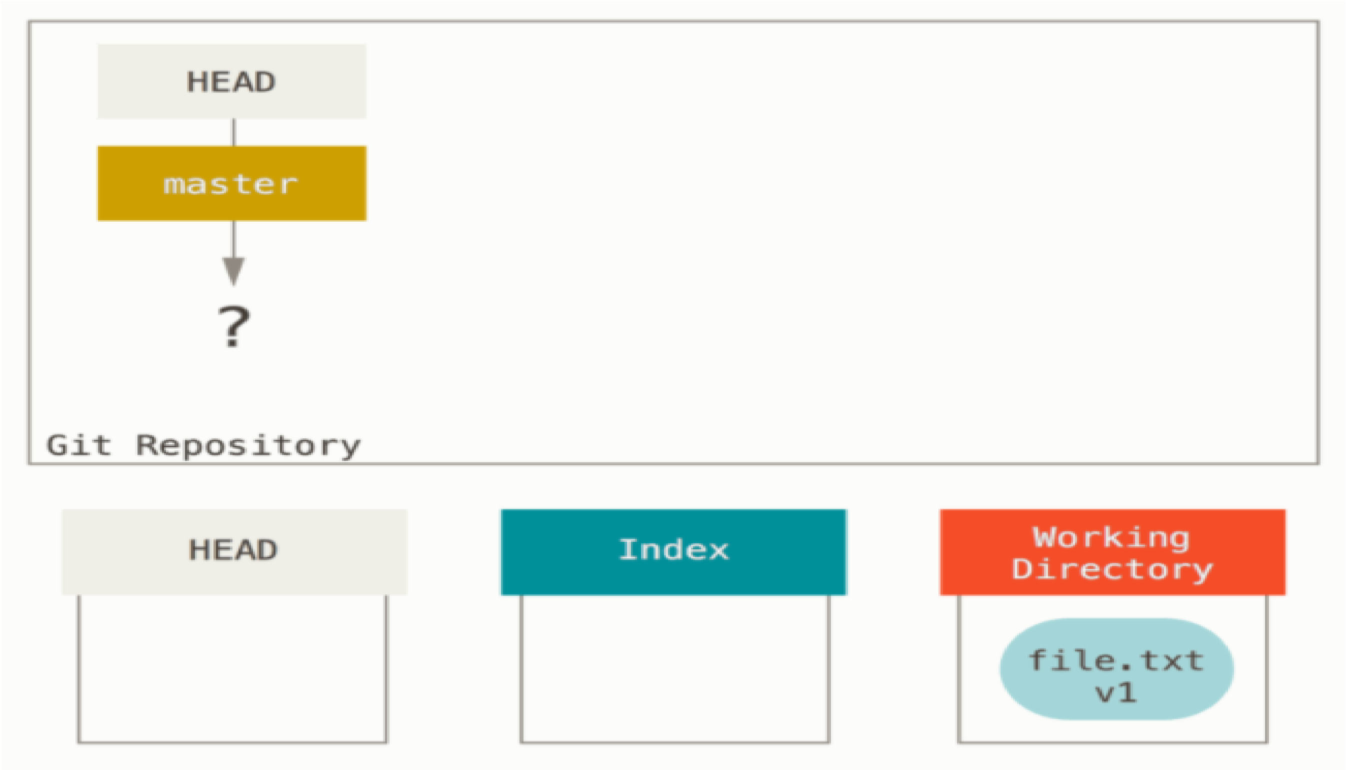


Adding a File to your Working Directory and Checking Status

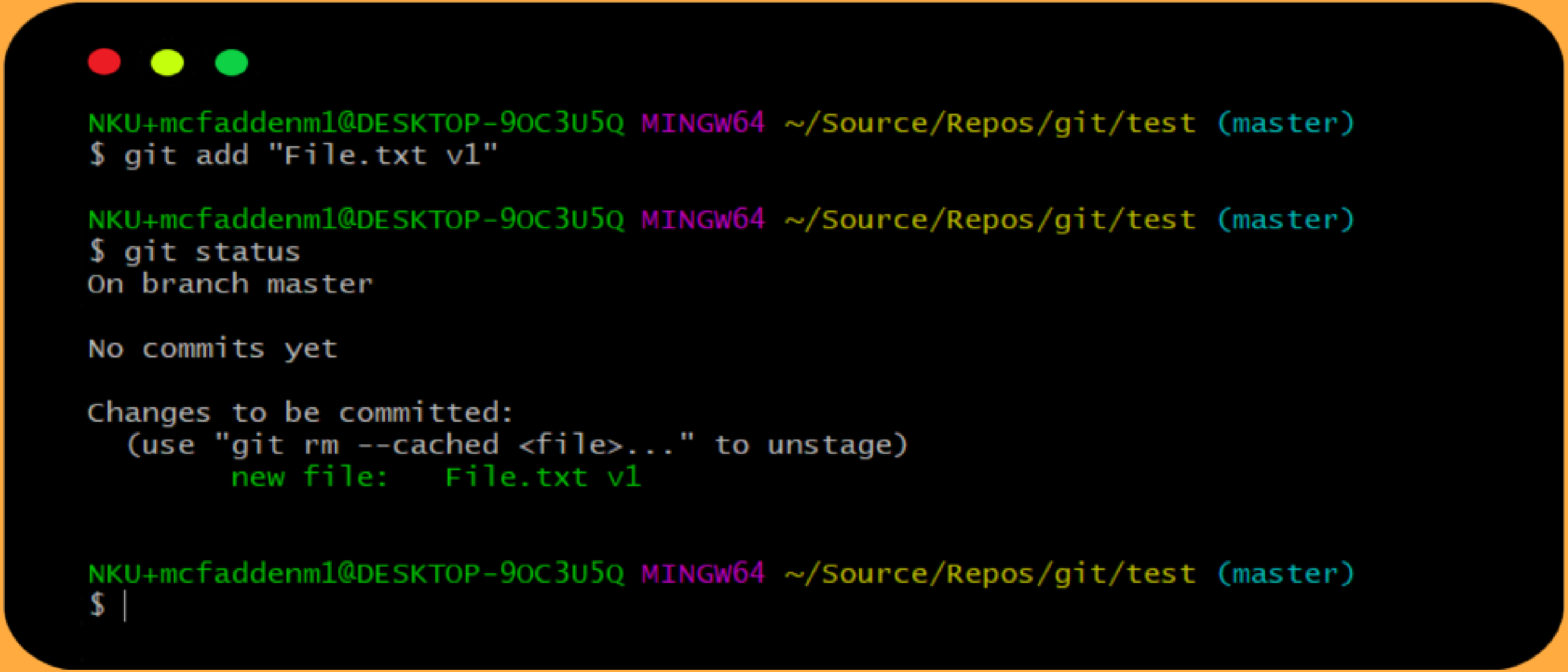


```
NKU+mcfaddenm1@DESKTOP-9OC3U5Q MINGW64 ~/Source/Repos/git/test (master)
$ echo "File.txt v1" > "File.txt v1"

NKU+mcfaddenm1@DESKTOP-9OC3U5Q MINGW64 ~/Source/Repos/git/test (master)
$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        File.txt v1

nothing added to commit but untracked files present (use "git add" to track)

NKU+mcfaddenm1@DESKTOP-9OC3U5Q MINGW64 ~/Source/Repos/git/test (master)
$ |
```

Git Repo & Flow Elements Now

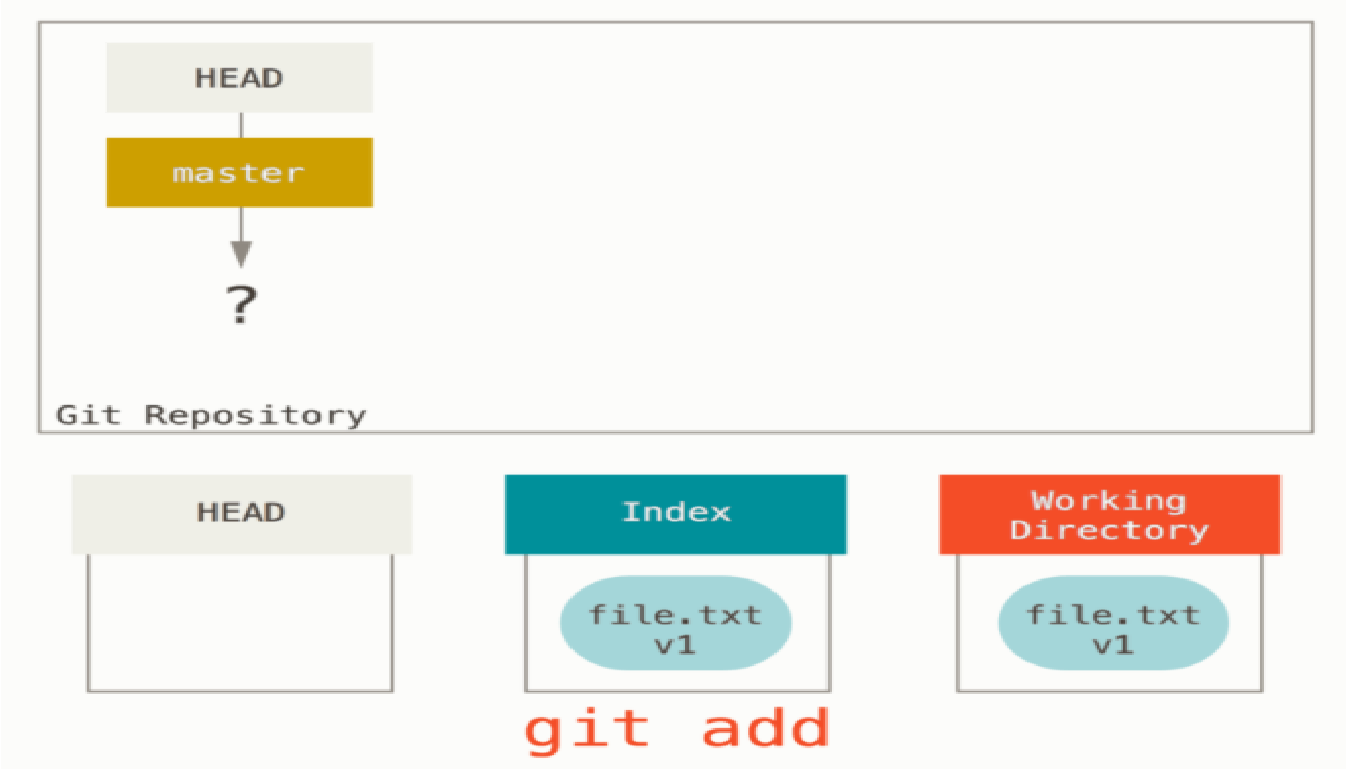


Adding your file to Stage (Index) and Checking Status.



```
NKU+mcfaddenm1@DESKTOP-9OC3U5Q MINGW64 ~/Source/Repos/git/test (master)
$ git add "File.txt v1"

NKU+mcfaddenm1@DESKTOP-9OC3U5Q MINGW64 ~/Source/Repos/git/test (master)
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   File.txt v1


NKU+mcfaddenm1@DESKTOP-9OC3U5Q MINGW64 ~/Source/Repos/git/test (master)
$ |
```

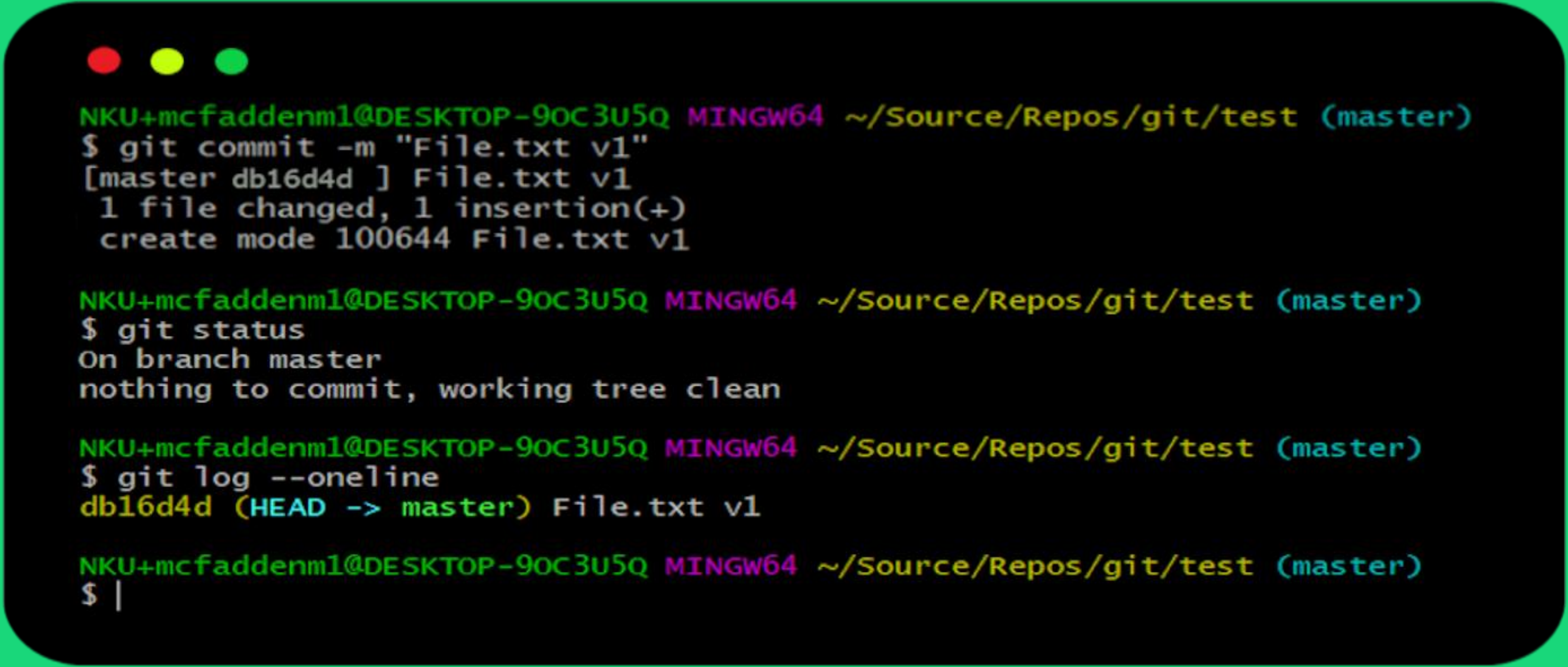
Git Repo & Flow Elements Now



Commits

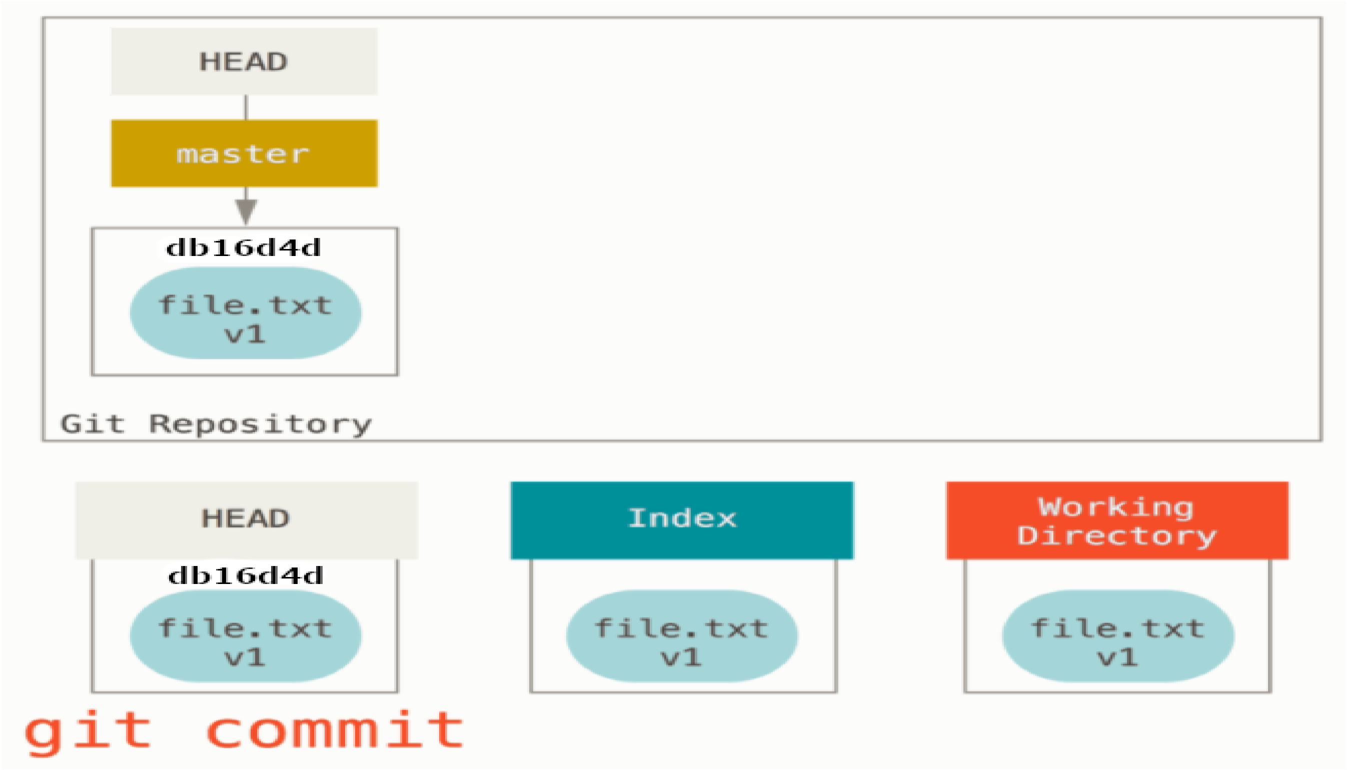
Checkpoints/Snapshot of the state of your repository (project) at a particular time.

Committing your file, Checking Status, and Viewing the Repo History.

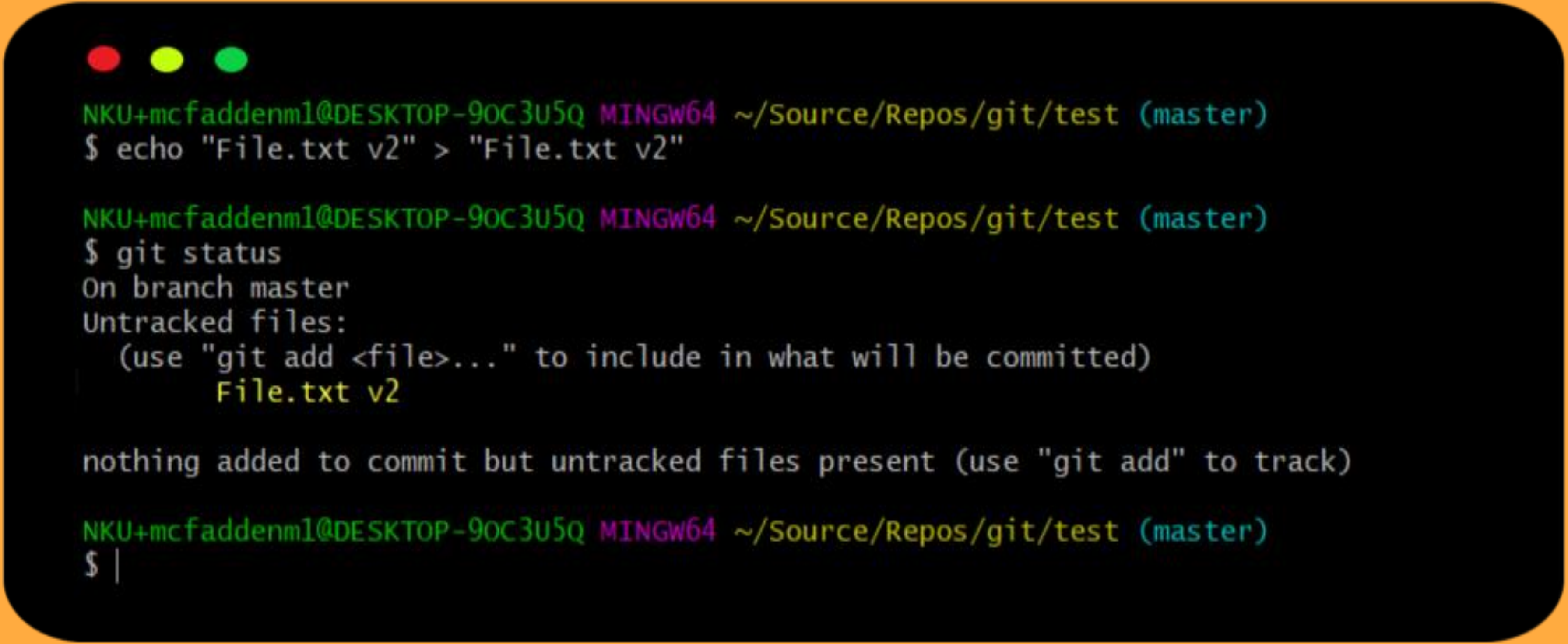


```
NKU+mcfaddenm1@DESKTOP-9OC3U5Q MINGW64 ~/Source/Repos/git/test (master)
$ git commit -m "File.txt v1"
[master db16d4d ] File.txt v1
 1 file changed, 1 insertion(+)
 create mode 100644 File.txt v1

NKU+mcfaddenm1@DESKTOP-9OC3U5Q MINGW64 ~/Source/Repos/git/test (master)
$ git status
On branch master
nothing to commit, working tree clean

NKU+mcfaddenm1@DESKTOP-9OC3U5Q MINGW64 ~/Source/Repos/git/test (master)
$ git log --oneline
db16d4d (HEAD -> master) File.txt v1

NKU+mcfaddenm1@DESKTOP-9OC3U5Q MINGW64 ~/Source/Repos/git/test (master)
$ |
```

Git Repo & Flow Elements Now

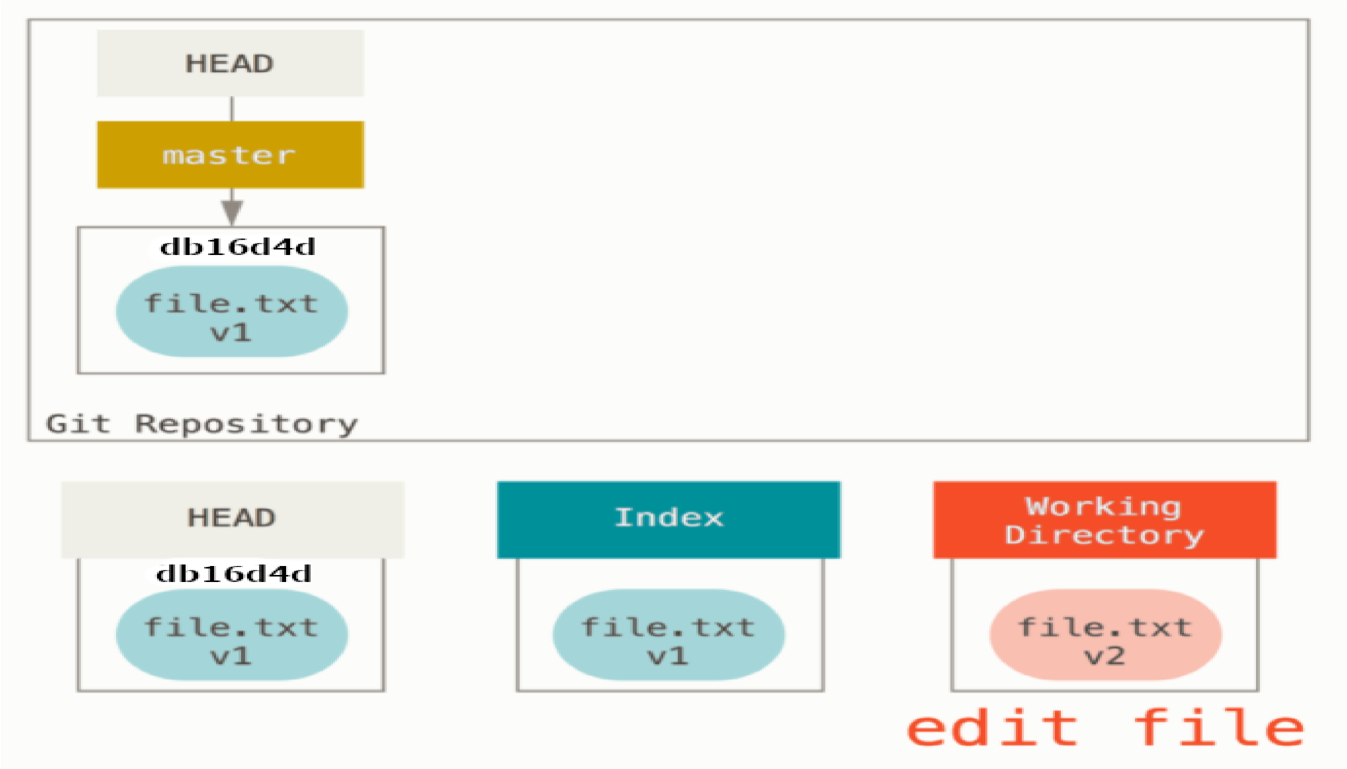


Adding a New File to your Working Directory and Checking Status.

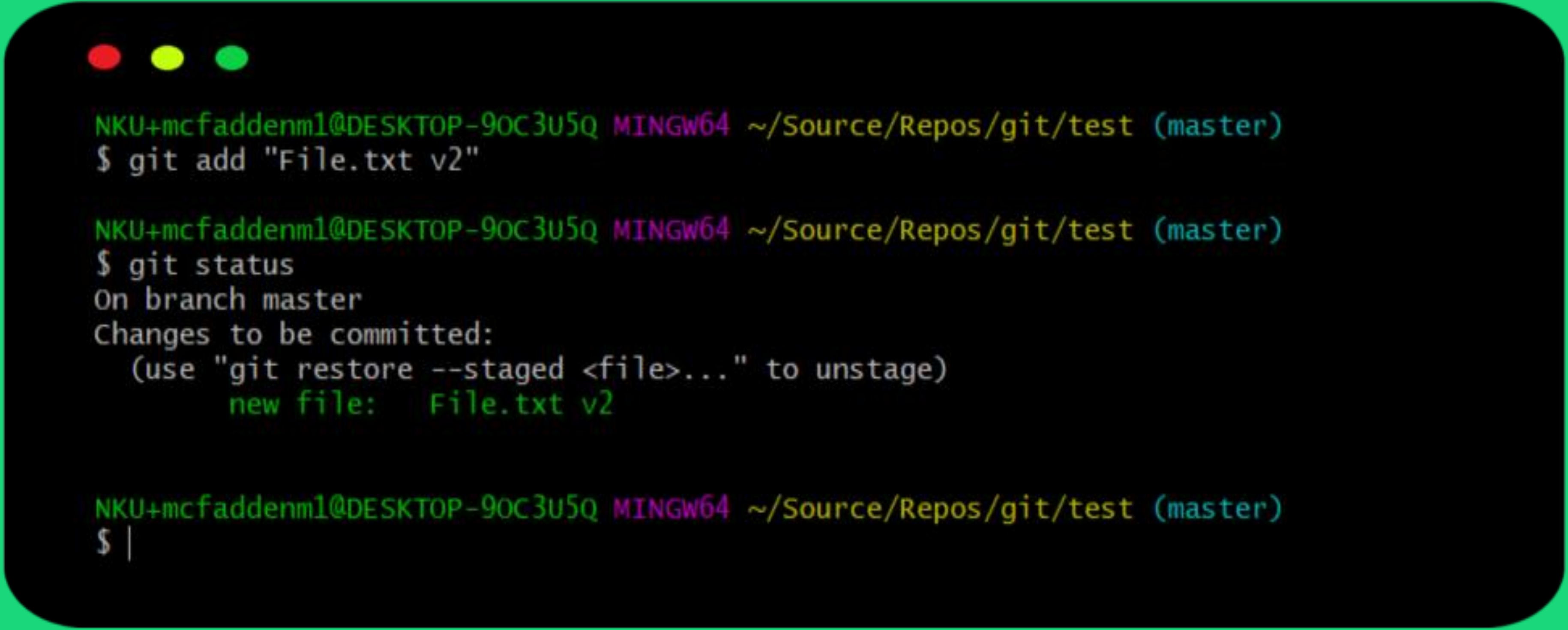


```
NKU+mcfaddenm1@DESKTOP-90C3U5Q MINGW64 ~/Source/Repos/git/test (master)
$ echo "File.txt v2" > "File.txt v2"

NKU+mcfaddenm1@DESKTOP-90C3U5Q MINGW64 ~/Source/Repos/git/test (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        File.txt v2

nothing added to commit but untracked files present (use "git add" to track)

NKU+mcfaddenm1@DESKTOP-90C3U5Q MINGW64 ~/Source/Repos/git/test (master)
$ |
```

Git Repo & Flow Elements Now

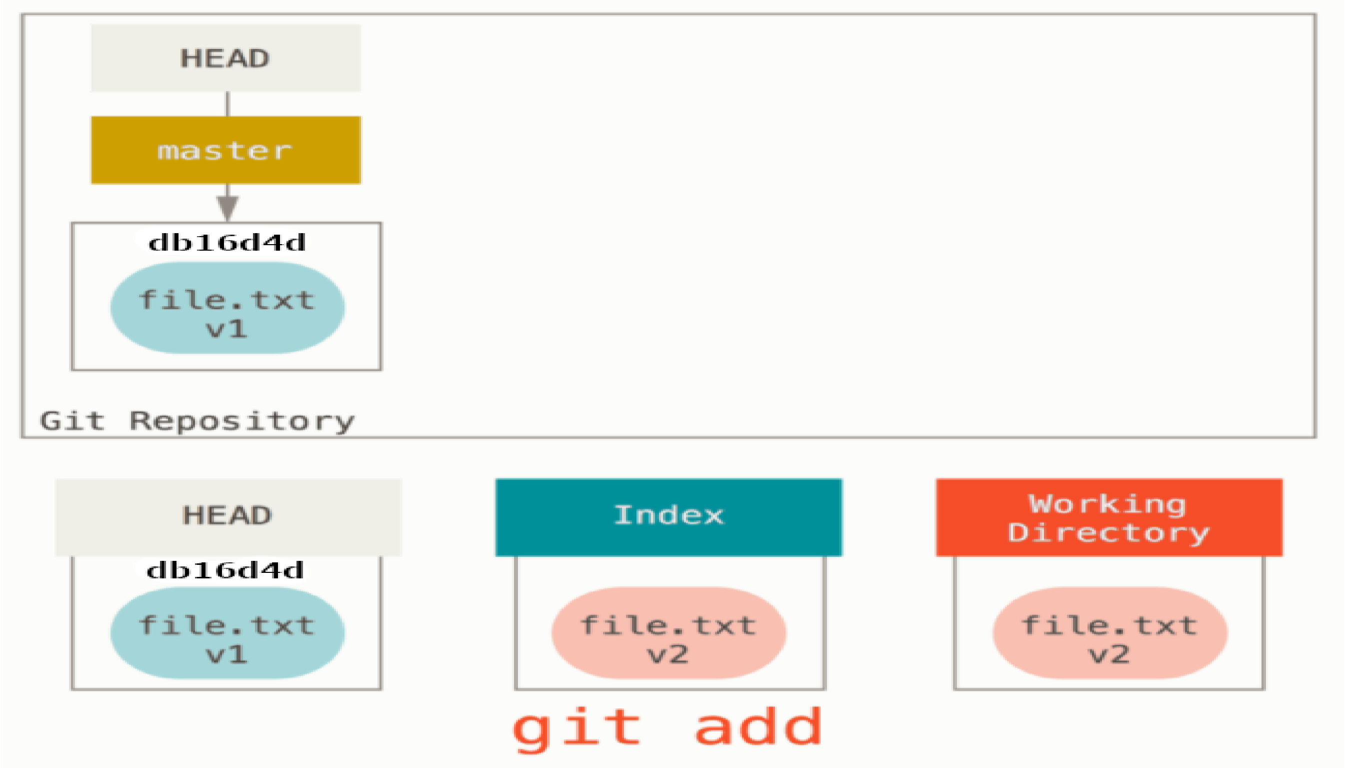


Adding your New File to Stage (Index) and Checking Status.

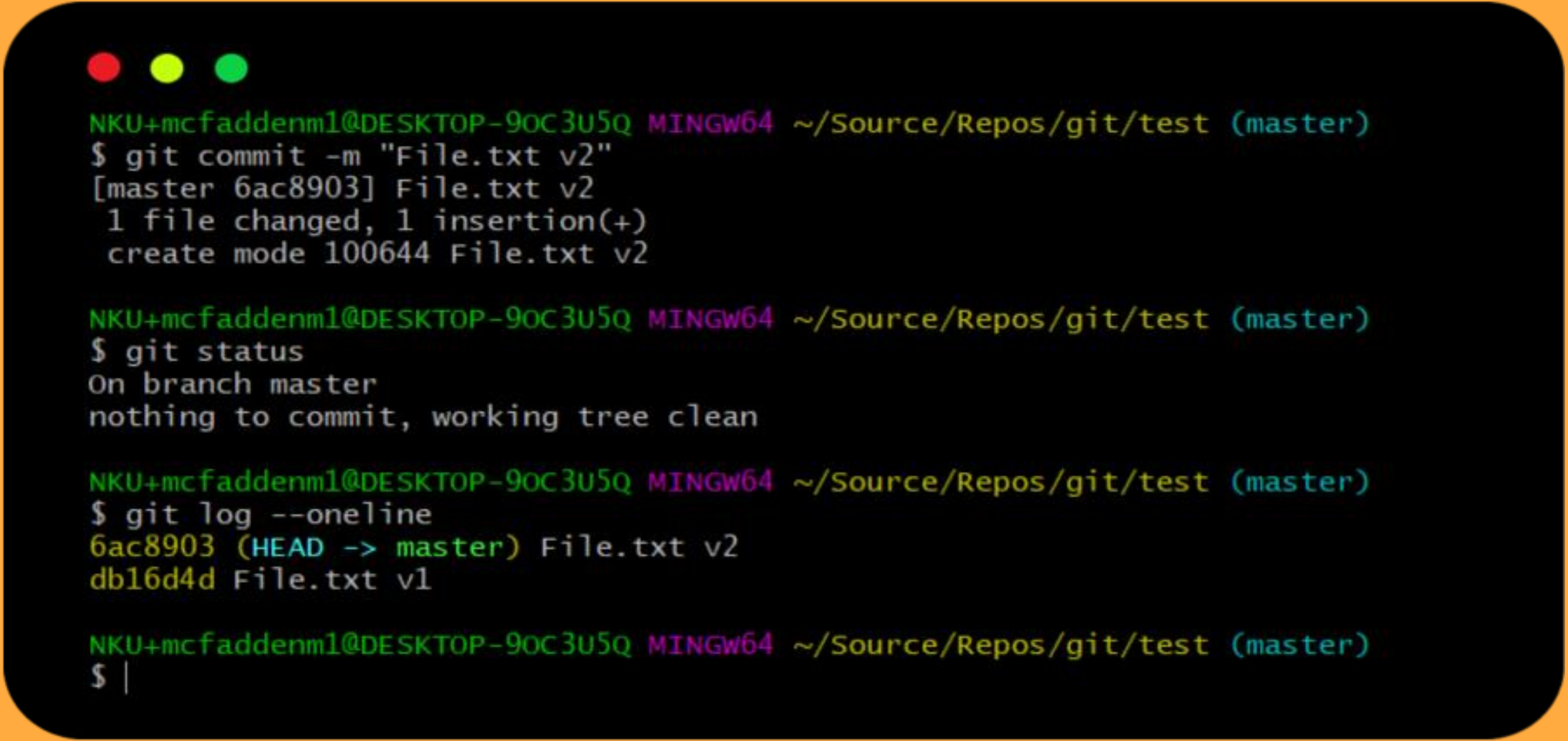


```
NKU+mcfaddenm1@DESKTOP-9OC3U5Q MINGW64 ~/Source/Repos/git/test (master)
$ git add "File.txt v2"

NKU+mcfaddenm1@DESKTOP-9OC3U5Q MINGW64 ~/Source/Repos/git/test (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   File.txt v2


NKU+mcfaddenm1@DESKTOP-9OC3U5Q MINGW64 ~/Source/Repos/git/test (master)
$ |
```

Git Repo & Flow Elements Now

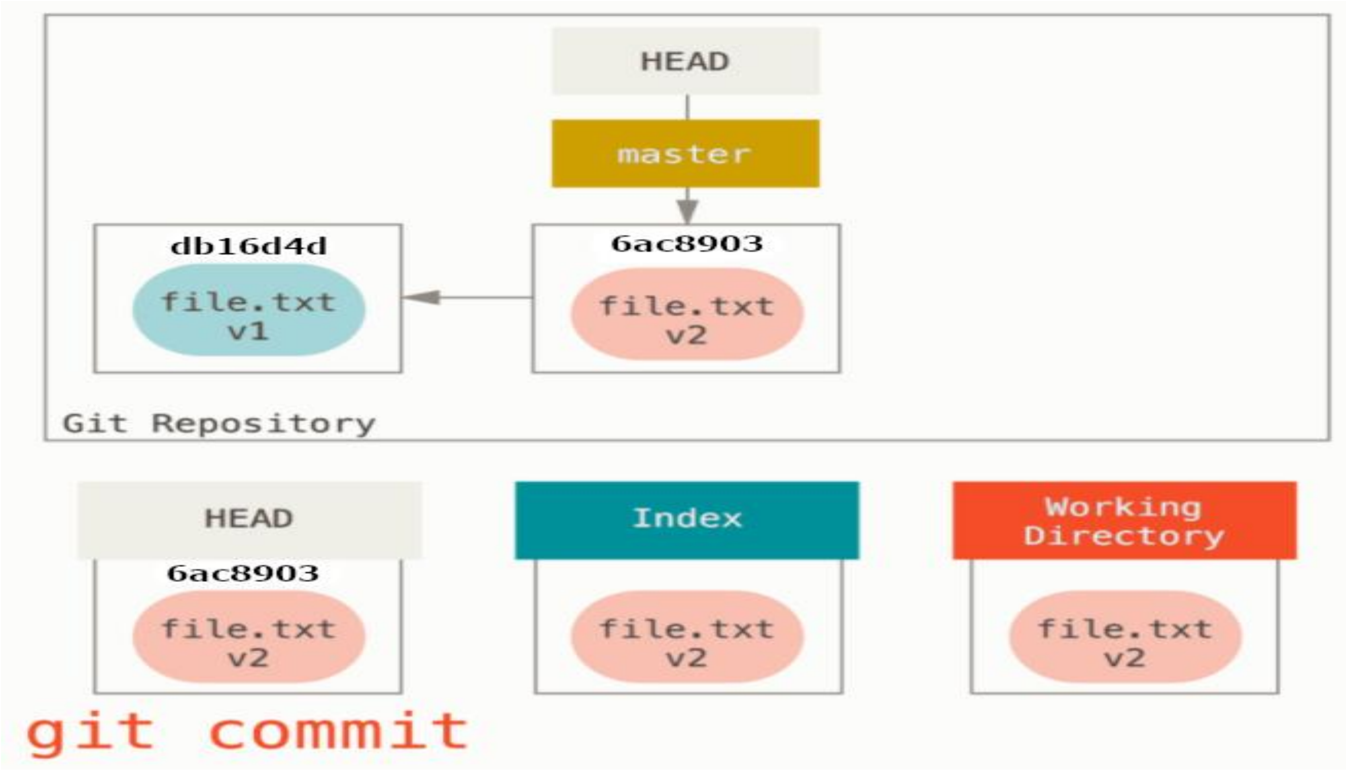


Committing your New File, Checking Status, and Viewing the Repo History.



```
NKU+mcfaddenm1@DESKTOP-90C3U5Q MINGW64 ~/Source/Repos/git/test (master)
$ git commit -m "File.txt v2"
[master 6ac8903] File.txt v2
 1 file changed, 1 insertion(+)
 create mode 100644 File.txt v2

NKU+mcfaddenm1@DESKTOP-90C3U5Q MINGW64 ~/Source/Repos/git/test (master)
$ git status
On branch master
nothing to commit, working tree clean

NKU+mcfaddenm1@DESKTOP-90C3U5Q MINGW64 ~/Source/Repos/git/test (master)
$ git log --oneline
6ac8903 (HEAD -> master) File.txt v2
db16d4d File.txt v1

NKU+mcfaddenm1@DESKTOP-90C3U5Q MINGW64 ~/Source/Repos/git/test (master)
$ |
```

Git Repo & Flow Elements Now



Time for some handson!

https://gitme.js.org