

A stylized graphic of a computer monitor with a thick blue border and a smaller inner rectangle representing the screen. The monitor is centered on the page.

# INTRO TO GIT

IM ALINA!

A HANDS-ON INTRODUCTION TO THE VERSION  
CONTROL SYSTEM AND ITS USES

# CONTENTS

- 01. What is Git?
- 02. Git & GitHub
- 06. Configure access to GitHub repositories
- 10. Clone a repository
- 14. Create!
- 15. Stage and Share

# WHAT IS GIT?

Git is a popular version control system.

## GIT VS GITHUB?

You do not need GitHub to use git, but you cannot use GitHub without using git.

# SETTING UP

# GIT

# GITHUB

# ACCOUNT

**Download:** <https://git-scm.com/downloads>

**A setup  
reference** <https://adamtheautomator.com/git-bash/>

**Create/Login  
to GitHub  
account**

Have the Git bash and  
GitHub side by side.

# HOW DO I INTEGRATE GIT WITH GITHUB?

Step 1: Configure access  
to GitHub repositories

Step 2: Clone a  
repository

Step 3: Create!

Step 4: Stage and  
Share

## STEP 1:

# CONFIGURE ACCESS TO GITHUB REPOSITORIES

Generate SSH Keys

Associate key with  
GitHub

Configure your GitHub  
user in Git bash

# GENERATE SSH KEYS

```
ssh-keygen -t rsa -b 4096 -C  
"yourgithubemail@blank.edu"
```

```
cat ~/.ssh/id_rsa.pub
```

Add the generated SSH Key on  
GitHub in **Settings→SSH keys**

# ASSOCIATE KEY WITH GITHUB

- Public profile
- Account
- Appearance
- Accessibility
- Notifications

---

Access

- Billing and plans
- Emails
- Password and authentication
- Sessions
- SSH and GPG keys**
- Organizations
- Moderation

---

Code, planning, and automation

- Repositories
- Codespaces

## SSH keys / Add new

Title

Key type

Authentication Key ↕

Key

Begins with 'ssh-rsa', 'ecdsa-sha2-nistp256', 'ecdsa-sha2-nistp384', 'ecdsa-sha2-nistp521', 'ssh-ed25519', 'sk-ecdsa-sha2-nistp256@openssh.com', or 'sk-ssh-ed25519@openssh.com'

Add SSH key



# CONFIGURE YOUR GITHUB USER IN GIT BASH

```
git config --global user.email  
"yourgithubemail@blank.edu"
```

```
git config --global user.name  
"Your Github Username"
```

```
git config --global  
user.password "your ssh key"
```

```
git config --global  
user.password "1234321"
```

```
git config --global  
credential.helper store
```

```
git config --list
```

# STEP 2: CLONE A REPOSITORY

<https://github.com/Alinamoo/IntroToGit>

```
git clone
```

Create new Repository on  
GitHub

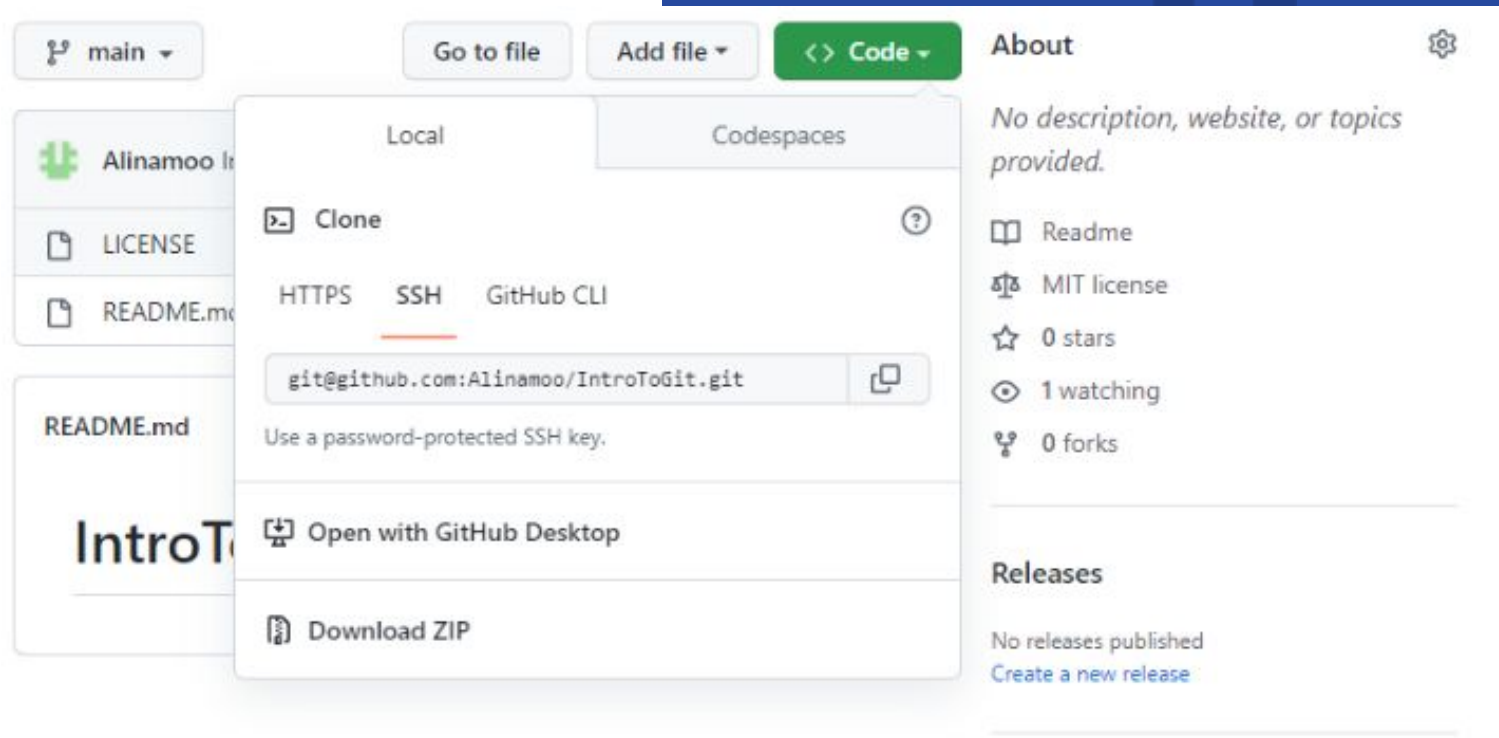
Configure a git URL as  
alias to GitHub repo

# GIT CLONE

<https://github.com/Alinamoo/IntroToGit>

```
git clone
```

```
git@github.com:Alinamoo/IntroToGit.git
```



# CREATE NEW REPOSITORY ON GITHUB

## Create a new repository

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

[Import a repository.](#)

Owner \*

Repository name \*



Alinamoo ▾



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

Description (optional)



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:

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



Add a README file

This is where you can write a long description for your project. [Learn more.](#)

### Add .gitignore

Choose which files not to track from a list of templates. [Learn more.](#)

.gitignore template: None ▾

### Choose a license

A license tells others what they can and can't do with your code. [Learn more.](#)

License: None ▾



You are creating a public repository in your personal account.

Create repository

# CONFIGURE A GIT URL AS ALIAS TO GITHUB REPO

```
git remote remove origin
```

```
git remote add origin
```

```
git@github.com
```

```
yourusername/yourreponame.git
```

```
git push -u origin main
```

```
$ git push -u origin main
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 12 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (7/7), 1.54 KiB | 1.54 MiB/s, done.
Total 7 (delta 0), reused 4 (delta 0), pack-reused 0
To github.com:Alinamoo/IntroToGitpart2adding.git
 * [new branch]      main -> main
Branch 'main' set up to track remote branch 'main' from 'origin'.
```

Lets Explore the linux  
terminal!

Create your file, get  
creative!

STEP 3:  
CREATE!



## STEP 4:

### STAGE AND SHARE

```
git add "file.py"
```

```
git commit -m "commit  
msg"
```

```
git push origin main
```

WAY TO GO! THAT WAS A LOT OF  
INFORMATION!







Linus Torvalds

So what this builds up to is that in the end we're all here to have fun. We might as well sit down and relax, and enjoy the ride.”

— Linus Torvalds, *Just for Fun: The Story of an Accidental Revolutionary*



**THANK YOU!**