TRABAJO PRACTICO 1

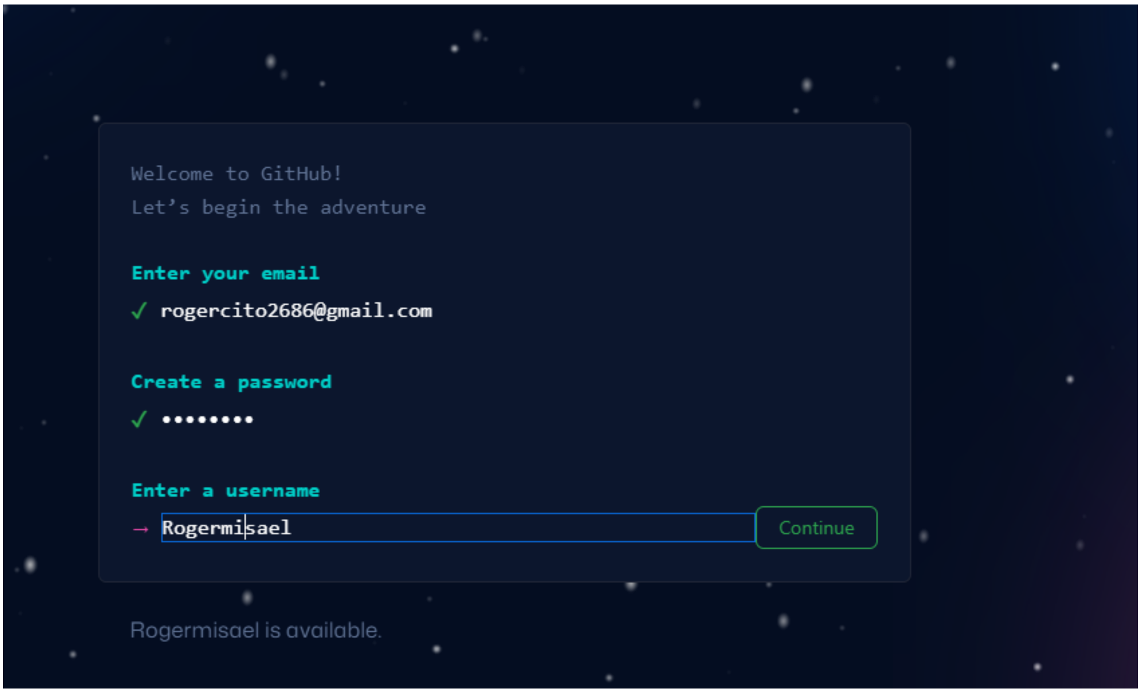
ACTIVIDAD 1: Crear un repositorio github y clonar

Alumno: Roger Misael Villarreal

Fecha: 10/03/2023

1- Descargo e Instalo Git en la Computadora

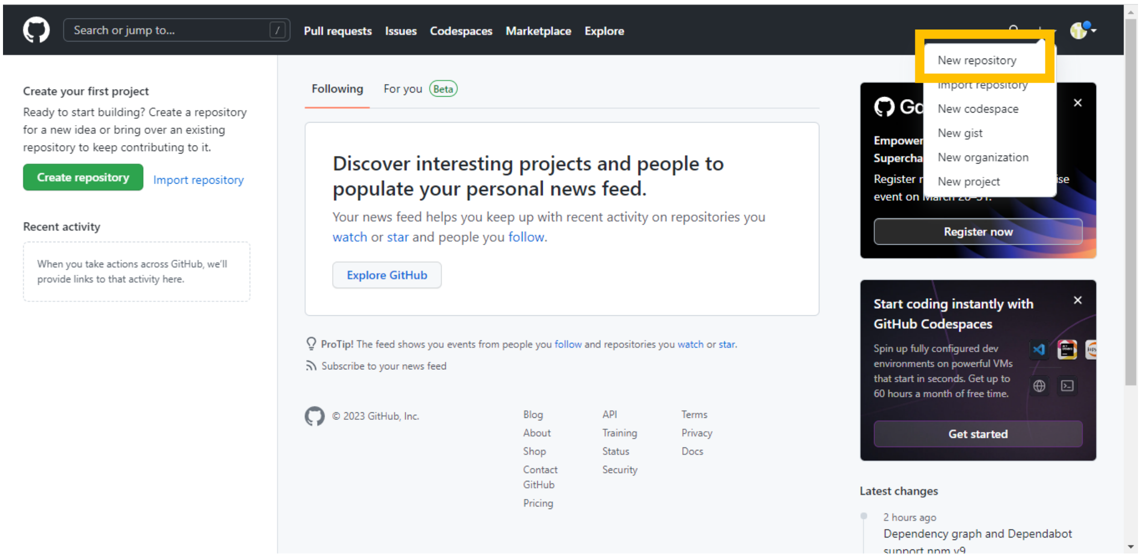
2- Creación de una cuenta



```
Welcome to GitHub!
Let's begin the adventure

Enter your email
✓ rogercito2686@gmail.com

Create a password
✓ ••••••••

Enter a username
→ Rogermisael
```

Rogermisael is available.

3- Dentro de tu cuenta GitHub, presiona el icono "+" y selecciona la opción "New Repository"



Create your first project

Ready to start building? Create a repository for a new idea or bring over an existing repository to keep contributing to it.

Recent activity

When you take actions across GitHub, we'll provide links to that activity here.

Discover interesting projects and people to populate your personal news feed.

Your news feed helps you keep up with recent activity on repositories you watch or star and people you follow.

ProTip! The feed shows you events from people you follow and repositories you watch or star.

Subscribe to your news feed

© 2023 GitHub, Inc.

Empower… Supercha… Register … …se event on March 20–31.

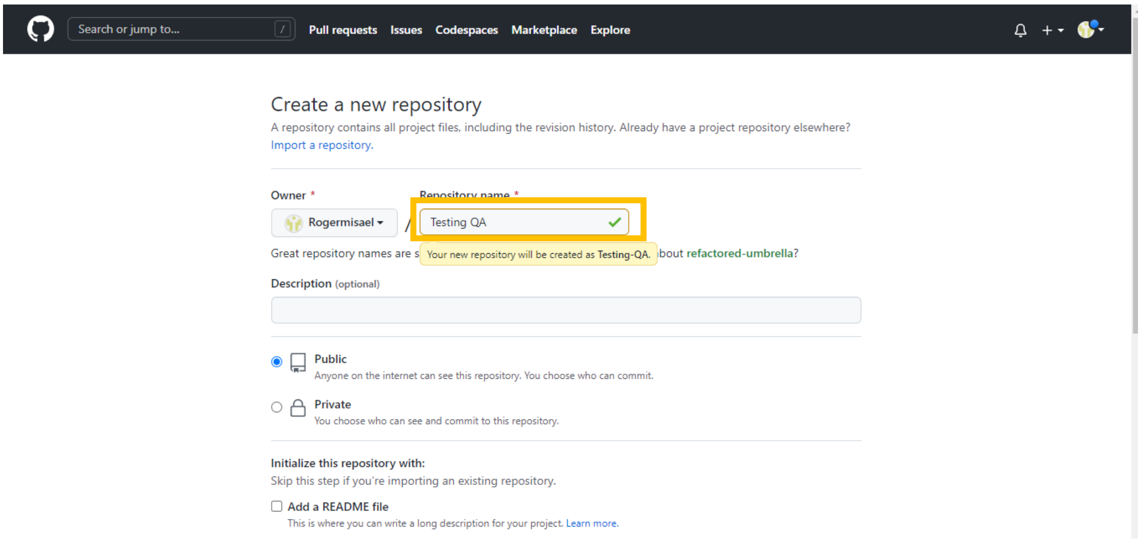
Start coding instantly with GitHub Codespaces

Spin up fully configured dev environments on powerful VMs that start in seconds. Get up to 60 hours a month of free time.

Latest changes

2 hours ago
Dependency graph and Dependabot support npm v9

4- Asigna un Nombre a tu repositorio; GitHub te recomienda que sea un nombre corto y fácil de recordar (quizá algo relacionado con el proyecto que vas a almacenar)



Create a new repository

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

Owner * Rogermisael / Repository name * Testing QA ✓

Great repository names are s Your new repository will be created as Testing-QA. bout refactored-umbrella?

Description (optional)

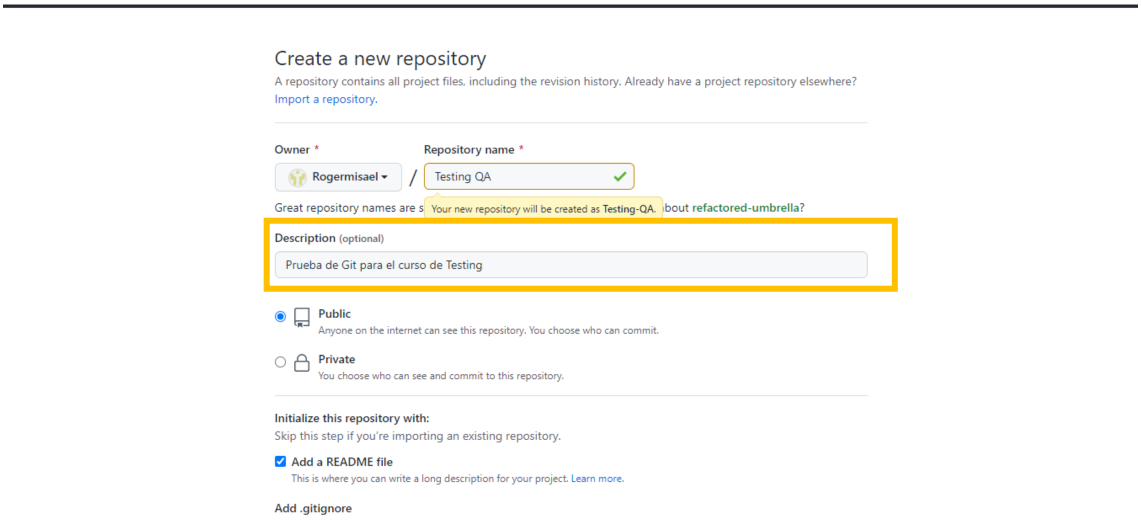
● Public
Anyone on the internet can see this repository. You choose who can commit.

○ 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.

5- Agrega una Descripción Corta y marca la opción "Inicializar este repositorio con un README", y haz clic en el botón "Crear Repositorio".



Create a new repository

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

Owner * Rogermisael / Repository name * Testing QA ✓

Great repository names are s Your new repository will be created as Testing-QA. bout refactored-umbrella?

Description (optional)
Prueba de Git para el curso de Testing

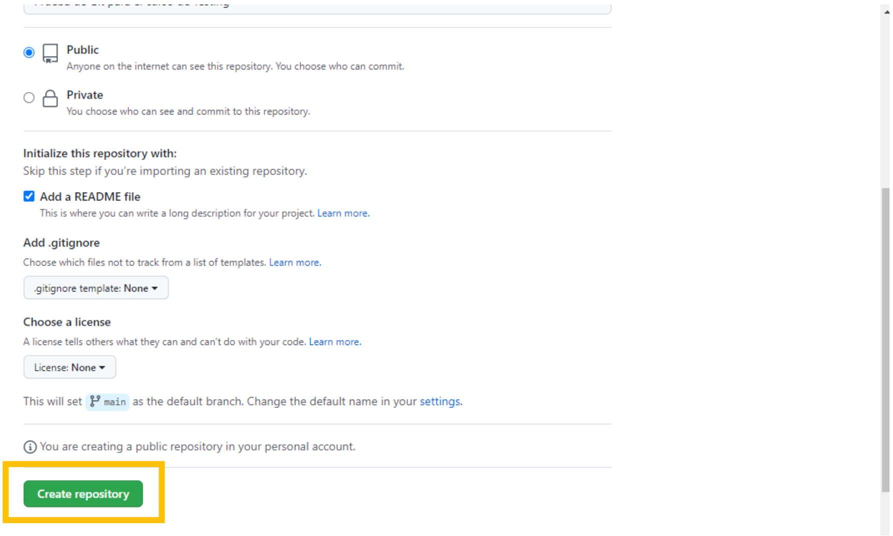
● Public
Anyone on the internet can see this repository. You choose who can commit.

○ 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



● 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

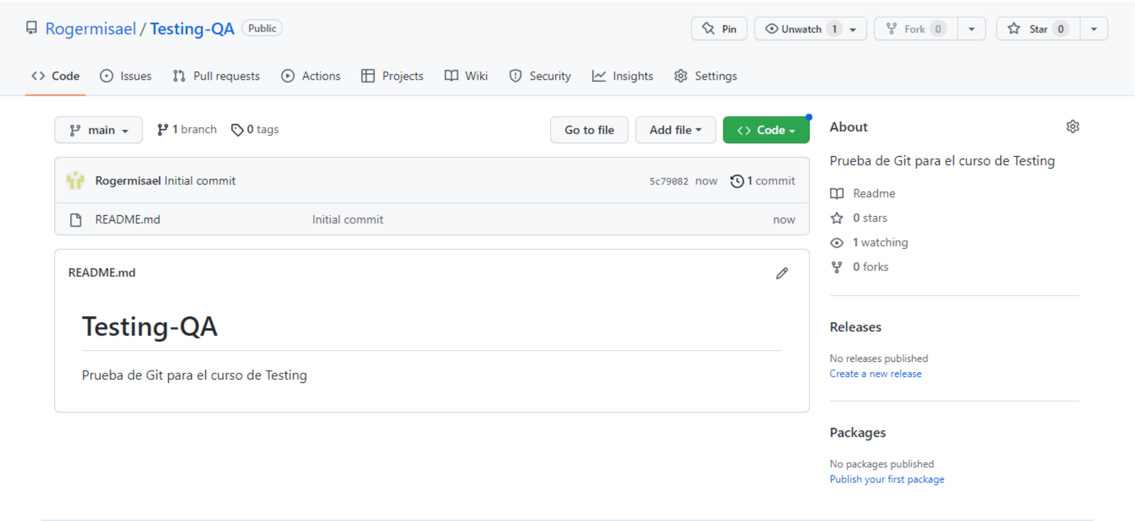
Choose a license
A license tells others what they can and can't do with your code. Learn more.

License: None

This will set main as the default branch. Change the default name in your settings.

You are creating a public repository in your personal account.

6- Así se ve tu REPOSITORIO:



Rogermisael / Testing-QA Public

1 branch 0 tags

Rogermisael Initial commit 5c79082 now 1 commit

README.md Initial commit now

README.md

Testing-QA

Prueba de Git para el curso de Testing

About

Prueba de Git para el curso de Testing

Readme
0 stars
1 watching
0 forks

Releases

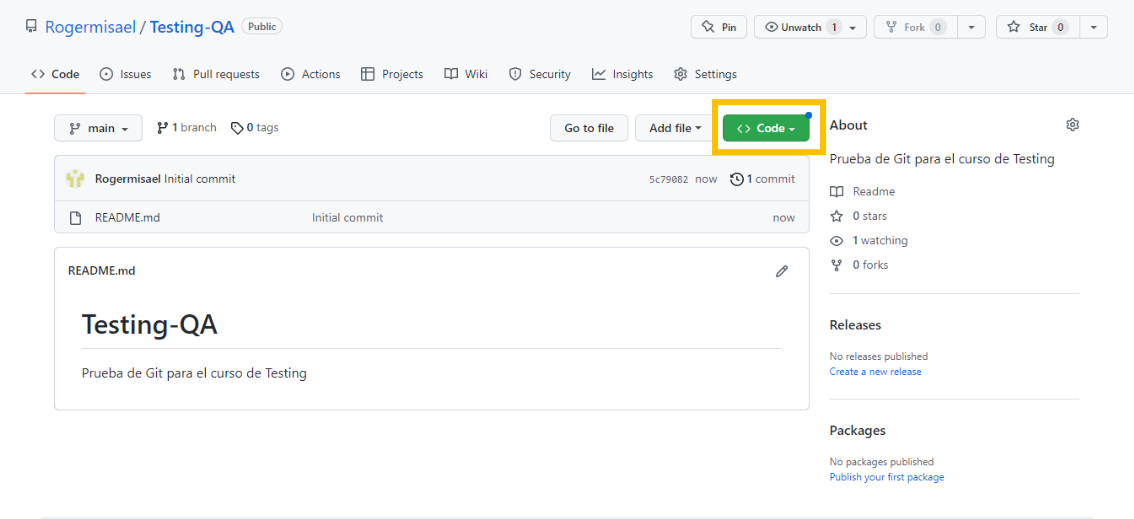
No releases published
Create a new release

Packages

No packages published
Publish your first package

7- CLONAR REPOSITORIO

- Dentro de tu repositorio identifica el botón verde “CODE” y da clic



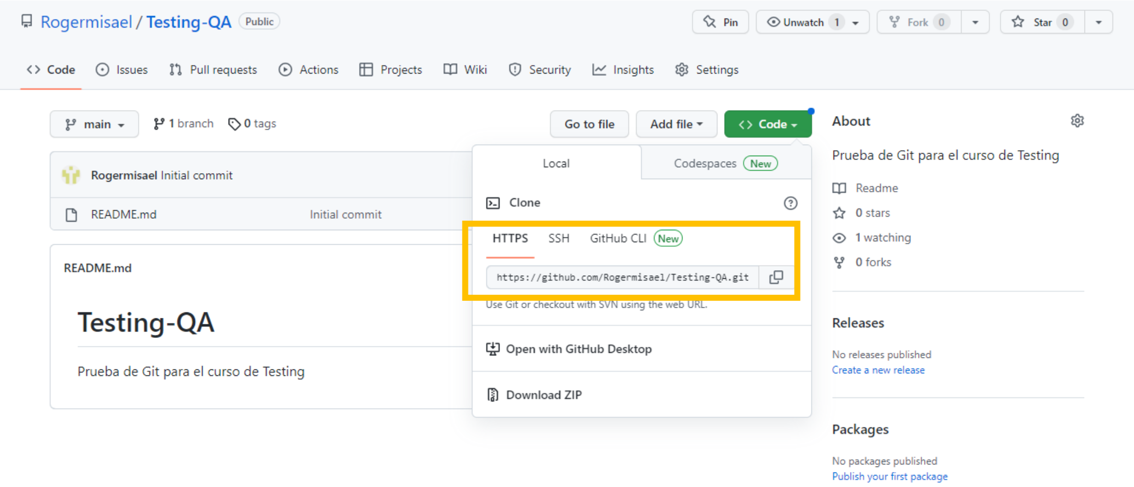
Prueba de Git para el curso de Testing

README.md

Testing-QA

Prueba de Git para el curso de Testing

- Copia la dirección HTTPS del apartado LOCAL



```
https://github.com/Rogermisael/Testing-QA.git
```

Use Git or checkout with SVN using the web URL.

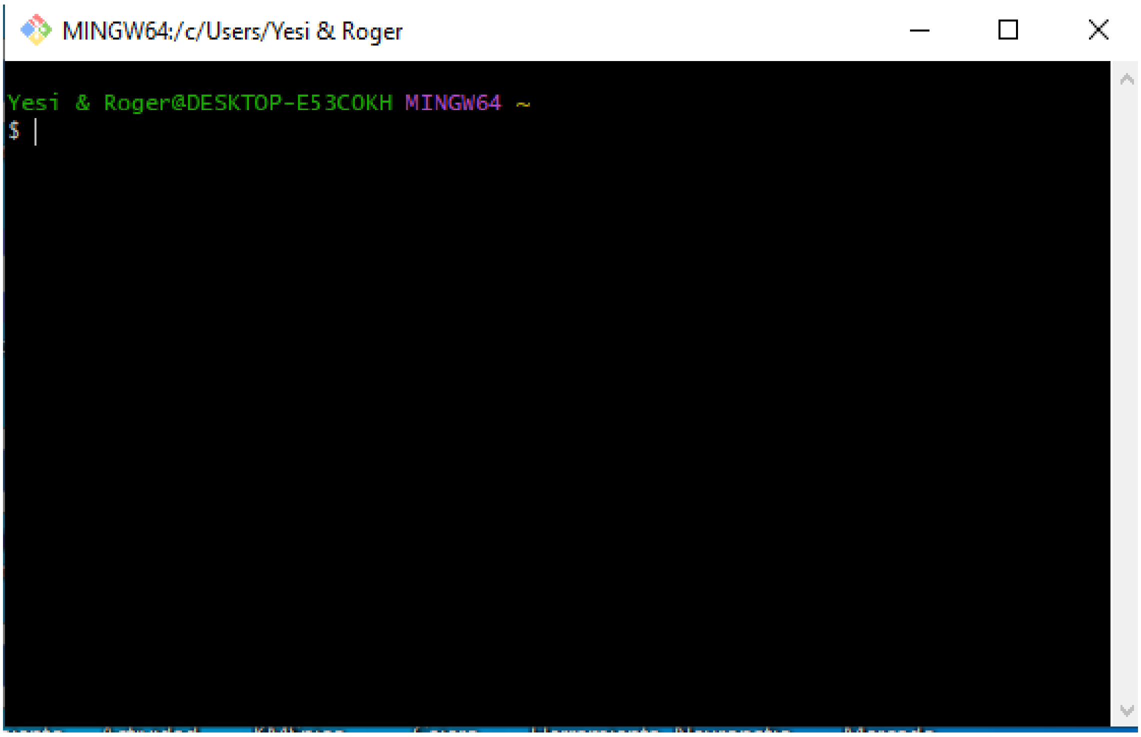
Prueba de Git para el curso de Testing

README.md

Testing-QA

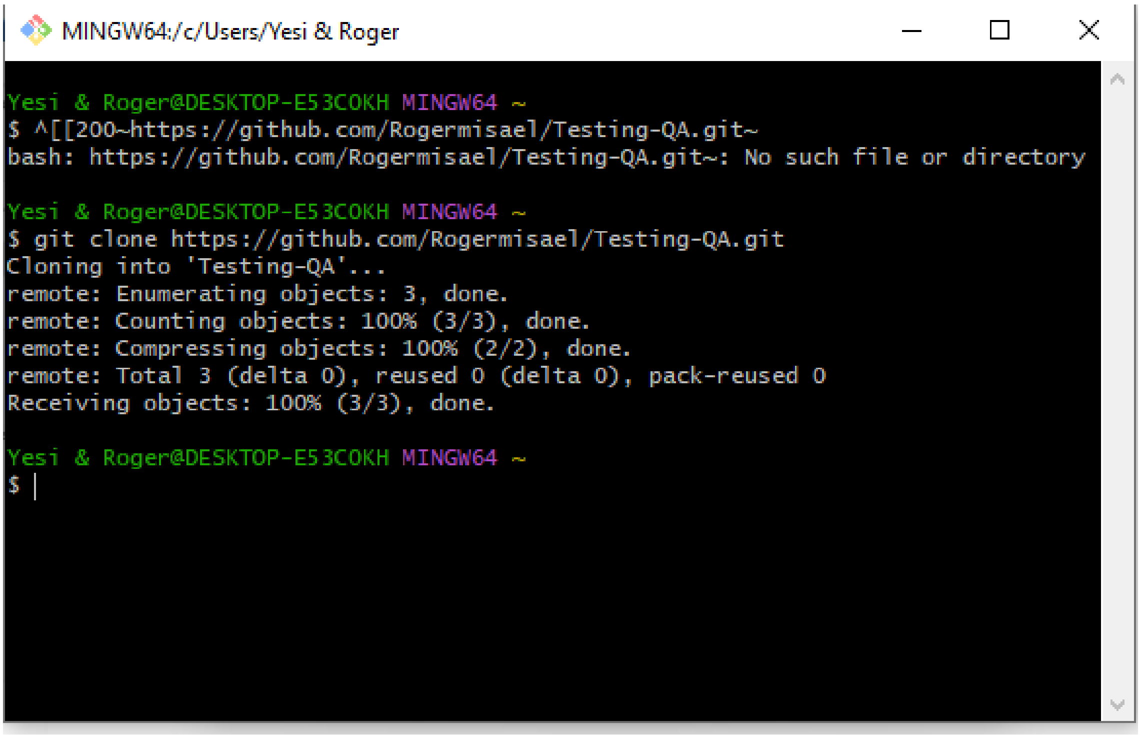
Prueba de Git para el curso de Testing

8- Abre la terminal GIT BASH que instalaste con git



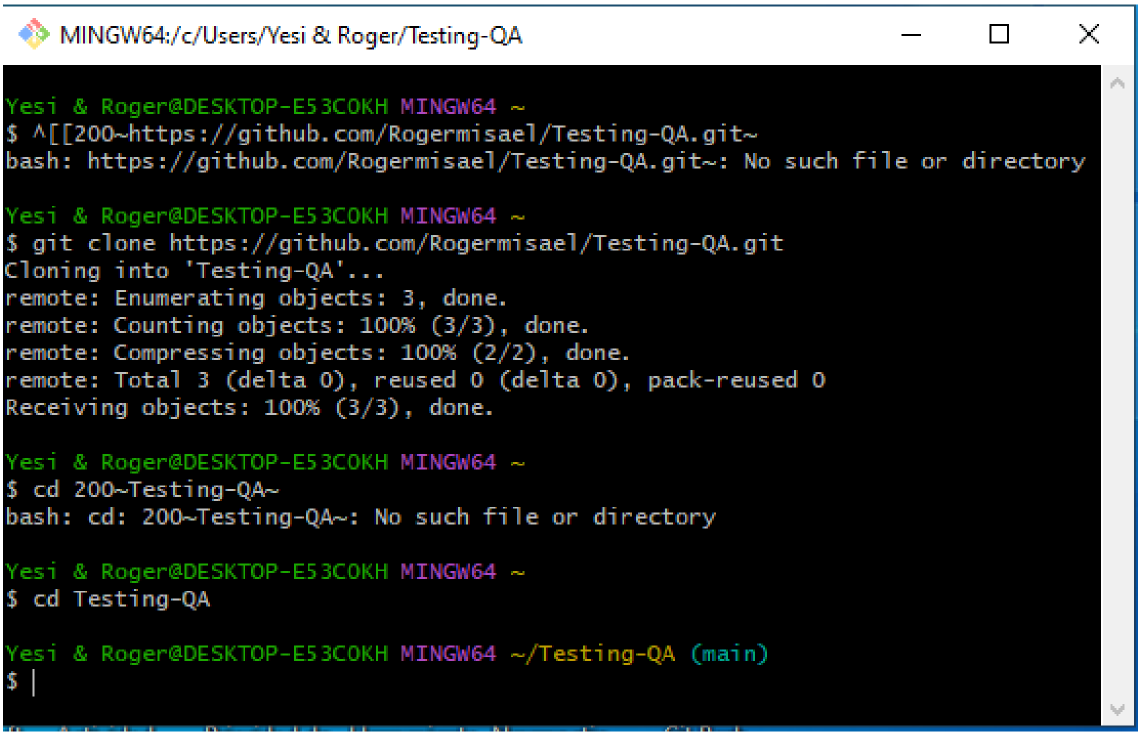
```
Yesi & Roger@DESKTOP-E53COKH MINGW64 ~
$ |
```

9- Escribe dentro de la terminal: git clone ***ingresa tu http*** y presiona enter



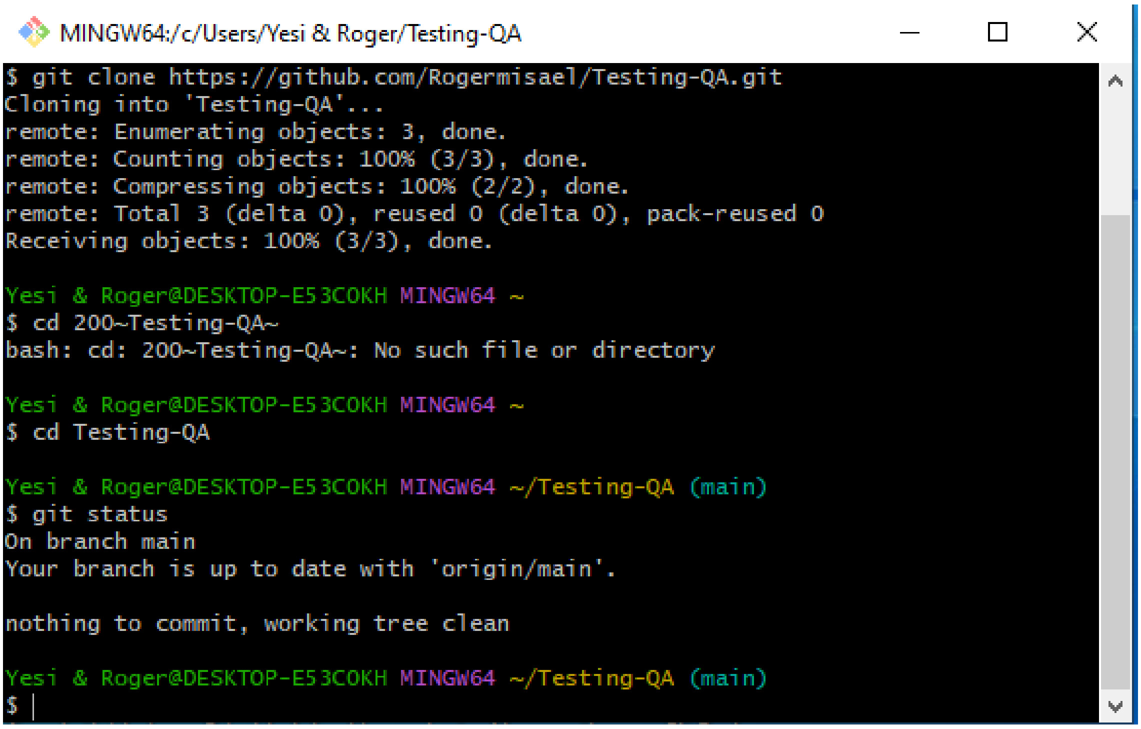
```
Yesi & Roger@DESKTOP-E53COKH MINGW64 ~
$ ^[[200~https://github.com/Rogermisael/Testing-QA.git~
bash: https://github.com/Rogermisael/Testing-QA.git~: No such file or directory

Yesi & Roger@DESKTOP-E53COKH MINGW64 ~
$ git clone https://github.com/Rogermisael/Testing-QA.git
Cloning into 'Testing-QA'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.

Yesi & Roger@DESKTOP-E53COKH MINGW64 ~
$ |
```

10- Utiliza el comando cd ***nombre del repositorio*** para navegar sobre el.



```
Yesi & Roger@DESKTOP-E53COKH MINGW64 ~
$ ^[[200~https://github.com/Rogermisael/Testing-QA.git~
bash: https://github.com/Rogermisael/Testing-QA.git~: No such file or directory

Yesi & Roger@DESKTOP-E53COKH MINGW64 ~
$ git clone https://github.com/Rogermisael/Testing-QA.git
Cloning into 'Testing-QA'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.

Yesi & Roger@DESKTOP-E53COKH MINGW64 ~
$ cd 200~Testing-QA~
bash: cd: 200~Testing-QA~: No such file or directory

Yesi & Roger@DESKTOP-E53COKH MINGW64 ~
$ cd Testing-QA

Yesi & Roger@DESKTOP-E53COKH MINGW64 ~/Testing-QA (main)
$ |
```

11- Vamos a usar el comando git status



```
$ git clone https://github.com/Rogermisael/Testing-QA.git
Cloning into 'Testing-QA'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.

Yesi & Roger@DESKTOP-E53COKH MINGW64 ~
$ cd 200~Testing-QA~
bash: cd: 200~Testing-QA~: No such file or directory

Yesi & Roger@DESKTOP-E53COKH MINGW64 ~
$ cd Testing-QA

Yesi & Roger@DESKTOP-E53COKH MINGW64 ~/Testing-QA (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean

Yesi & Roger@DESKTOP-E53COKH MINGW64 ~/Testing-QA (main)
$ |
```