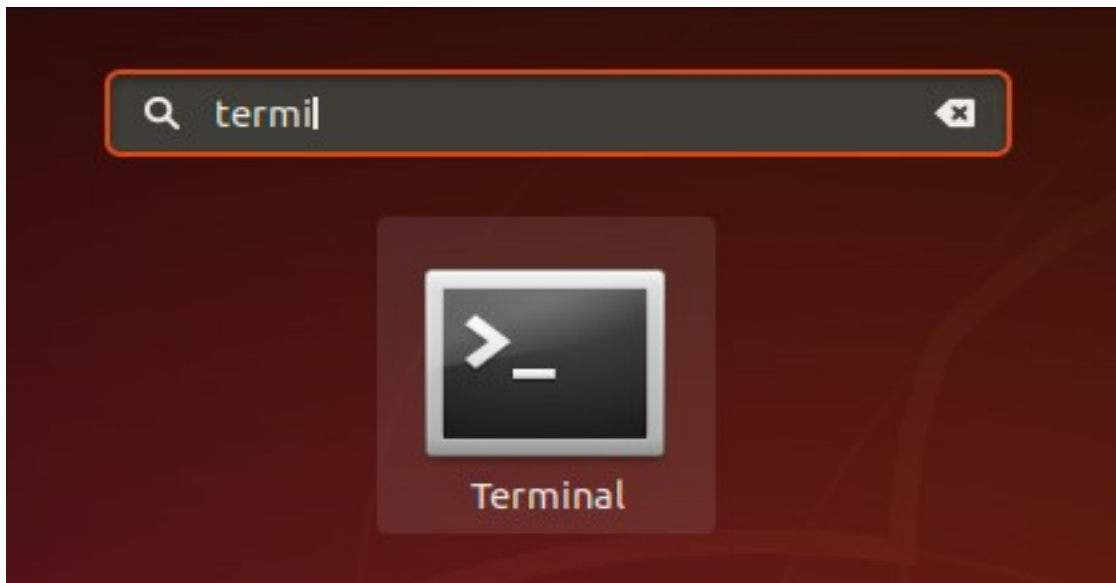


# Crear Repositorio para el uso de Proyectos con Git y GitHub

Abrir una terminal en el Sistema Operativo GNU/Linux presionando la tecla de “**Window**” de nuestro teclado y escribir “**terminal**”.



Una vez abierta la “**Terminal**” crear una carpeta con el comando “**mkdir**” donde se desea guardar los proyectos de programación y otorgarle un nombre.

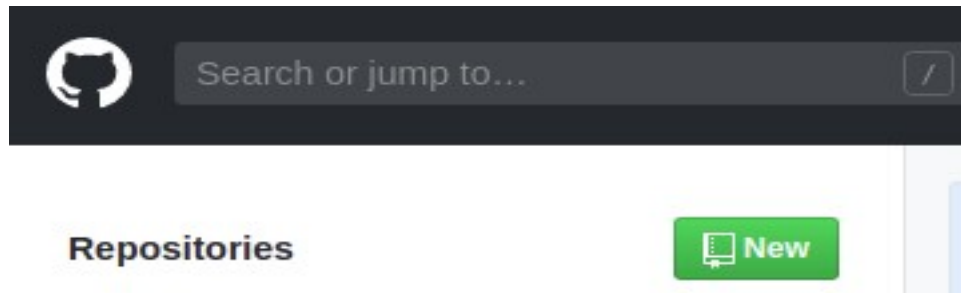
**Nota:** Se recomienda realizar un cambio de directorio antes de crear la carpeta hacia los directorios “**Escritorio**”, “**Descargas**” y/o “**Documentos**” para que la carpeta no quede dentro de la carpeta “**/home/user**” y esto pueda ocasionar que todos los archivos y/o carpetas que se encuentran en esta ruta se sincronicen.

```
ch Terminal Help
D:~$ cd Desktop/
D:~/Desktop$ mkdir Proyectos
```

Creada la carpeta que contendrá los proyectos de programación es necesario ingresar a “<https://github.com/>” y proporcionar el usuario y contraseña para acceder. Si no se cuenta todavía con una cuenta los pasos son muy sencillos, los cuales son proporcionar un correo electrónico, una contraseña y para finalizar abrir el correo de confirmación que llegará al correo proporcionado para el registro.

Una vez dentro de la página “<https://github.com/>” crearemos un repositorio al cual le otorgaremos un nombre.

En la página principal presionaremos el boton “**New**” y crearemos un repositorio con el nombre “**Proyectos**”, Description “**Algun texto que sirva de utilidad**”, seleccionar “**Public**” si se desea que se colabore con este repositorio o “**Private**” si se desea utilizar para uso personal. (Siga los pasos de la imagen).





## 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 \*

 / Proyectos 

Great repository names are short and memorable. Need inspiration? How about **friendly-octo-parakeet**?

Description (optional)


Repositorio para los proyectos de programacion

- ☒  **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** ▼ 

Create repository

Nos aparecera una nueva ventana donde se nos indica que el repositorio ha sido creado. Ahora es necesario inicializar la carpeta donde se guardaran los proyectos con los comandos que se muestran a continuacion.

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

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

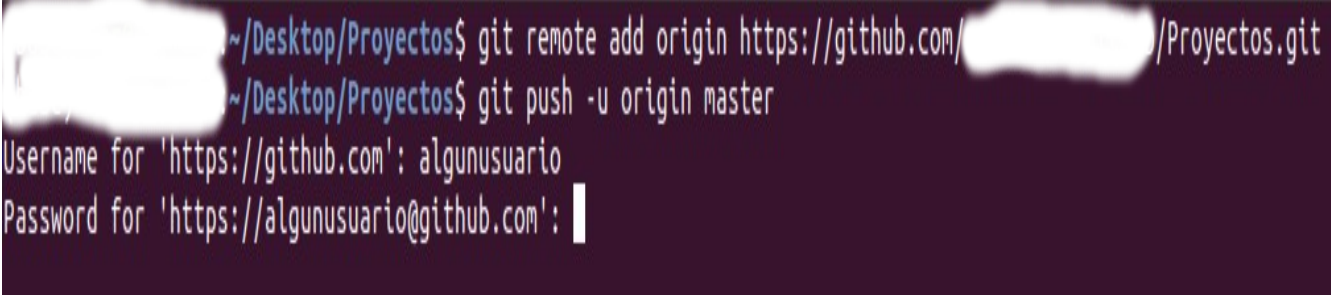
En la terminal nos colocados en la carpeta “**Proyectos**” y ejecutaremos previamente los siguientes comando.

```
~/Desktop/Proyectos$ git config --global user.name algunUsuario
~/Desktop/Proyectos$ git config --global user.email algunUsuario@gmail.com
```

Realizada esta acción ingresaremos los comandos mostrados con anterioridad.

```
[redacted]:~/Desktop/Proyectos$ echo "# Proyectos" >> README.md
[redacted]:~/Desktop/Proyectos$ git init
Initialized empty Git repository in /home/csr/Desktop/Proyectos/.git/
[redacted]:~/Desktop/Proyectos$ git add README.md
[redacted]:~/Desktop/Proyectos$ git commit -m "first commit"
[master (root-commit) 840dd6d] first commit
1 file changed, 1 insertion(+)
create mode 100644 README.md
```

Al ingresar el comando “**git push -u origin master**” se nos pedirá el usuario de “**GitHub**” y su contraseña.

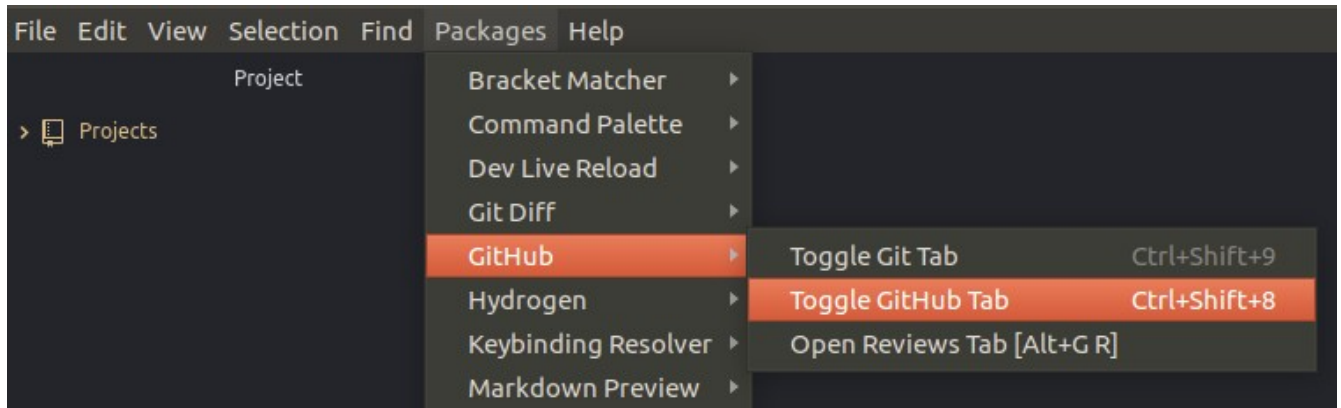
A terminal window with a dark purple background and light blue text. The prompt is ~/Desktop/Proyectos\$. The first command is git remote add origin https://github.com/[redacted]/Proyectos.git. The second command is git push -u origin master. This is followed by two prompts: Username for 'https://github.com': algunosusuario and Password for 'https://algunusuario@github.com': [redacted].

```
~/Desktop/Proyectos$ git remote add origin https://github.com/[redacted]/Proyectos.git
~/Desktop/Proyectos$ git push -u origin master
Username for 'https://github.com': algunosusuario
Password for 'https://algunusuario@github.com': [redacted]
```

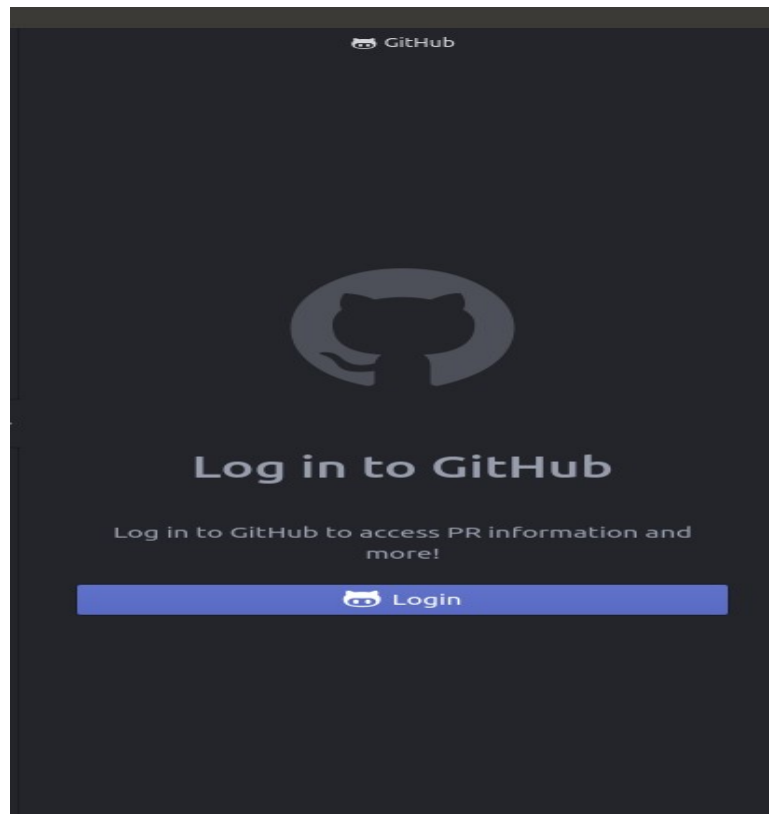
Con esto se ha inicializado y sincronizado la carpeta “Proyectos” con el repositorio en “GitHub.com”.

# Iniciarlizar GitHub en Atom

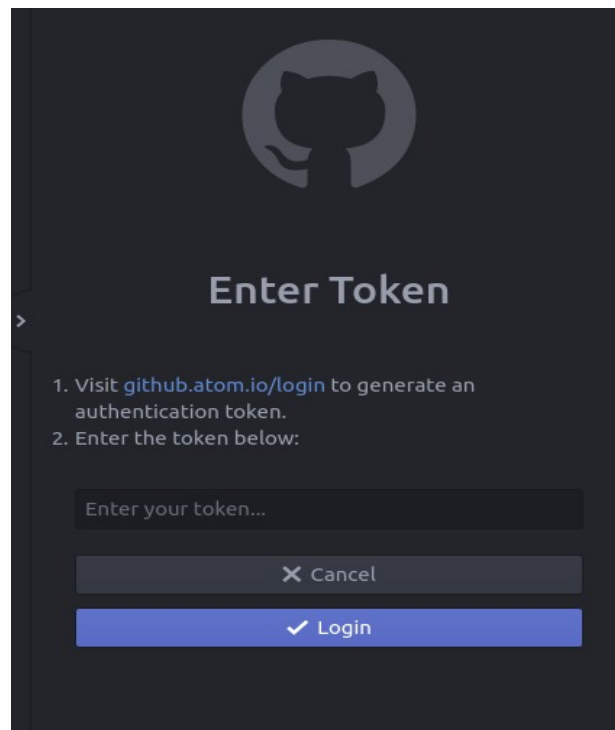
Ir al menu “**Packages**” y seleccionar “**GitHub**” → “**Toggle GitHub Tab**”



En la pestaña de “**GitHub**” presionar el botón “**Login**”.



En este punto es necesario ya haber ingresado a “<https://github.com/>” y estar logeado con su usuario y contraseña respectivamente, además de inicializada la carpeta que se desea con “**git**”. Se nos pedirá que visitemos “**github.atom.io/login**” para generar el token de autenticación.




The image shows a dark-themed dialog box titled "Enter Token" with the GitHub logo at the top. It contains two numbered instructions: "1. Visit github.atom.io/login to generate an authentication token." and "2. Enter the token below:". Below the instructions is a text input field with the placeholder "Enter your token...". At the bottom are two buttons: a grey "Cancel" button with a close icon and a blue "Login" button with a checkmark icon.

A continuación se nos redirigirá a la página donde se nos mostrará el token de autenticación que deberemos de ingresar en el campo “**Enter your token...**” de la imagen anterior.

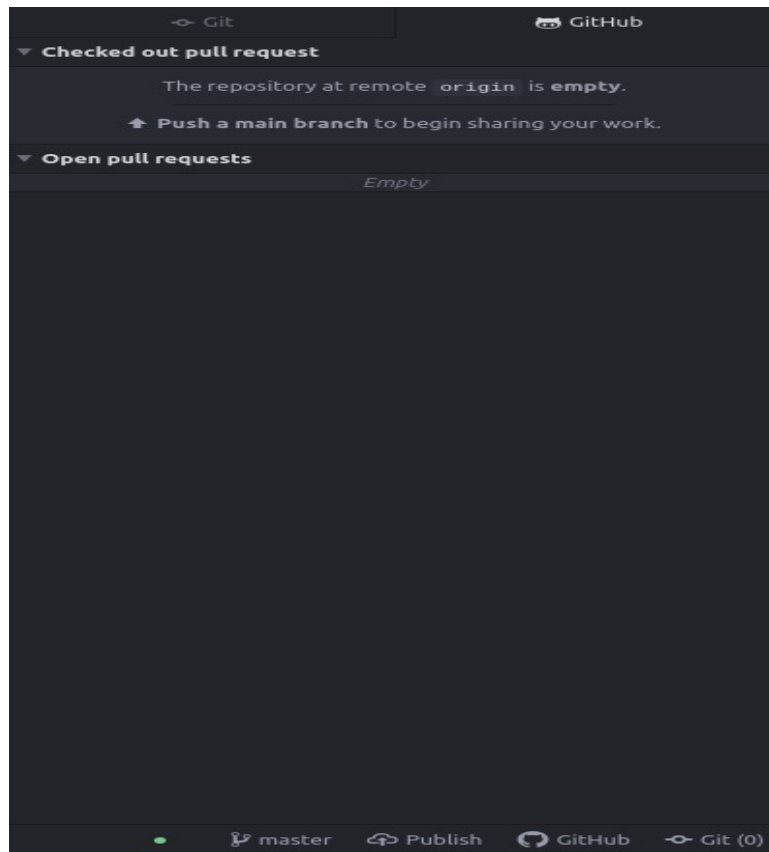


#### Your GitHub token

 Copy token

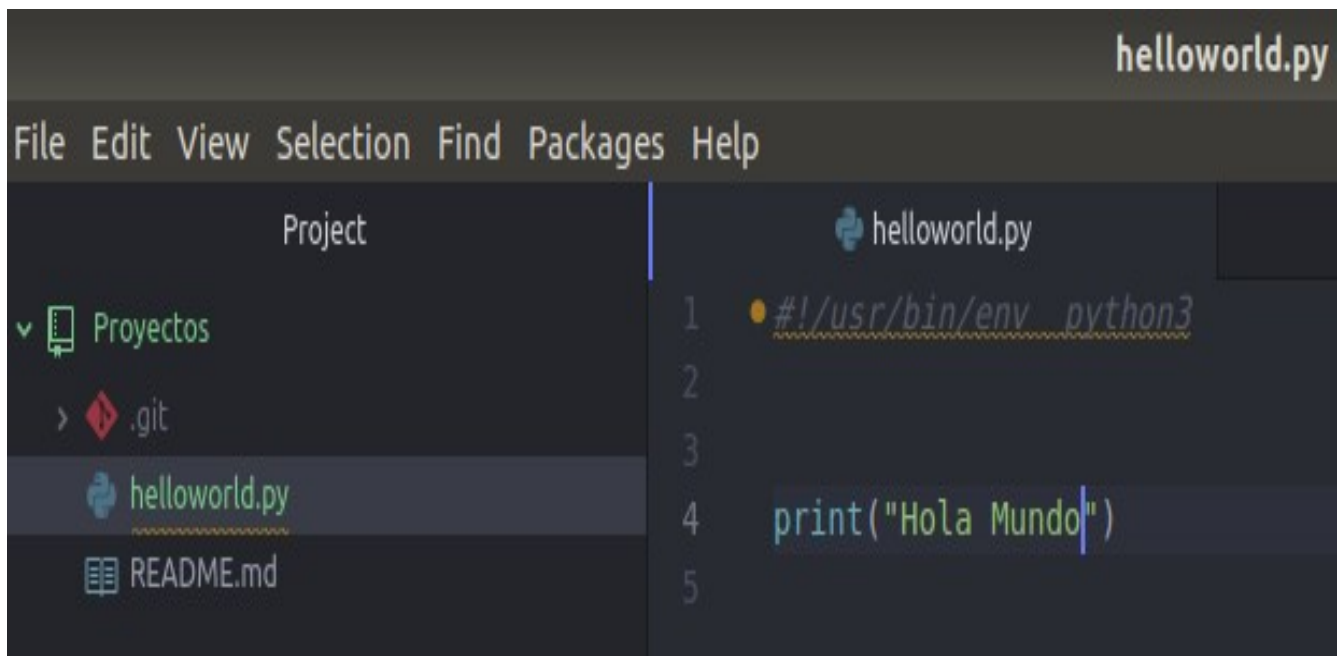
To finish signing in, copy your token and paste it into the sign-in form in Atom.

Una vez ingresado el token de autenticación el panel de GitHub aparecerá de la siguiente manera.

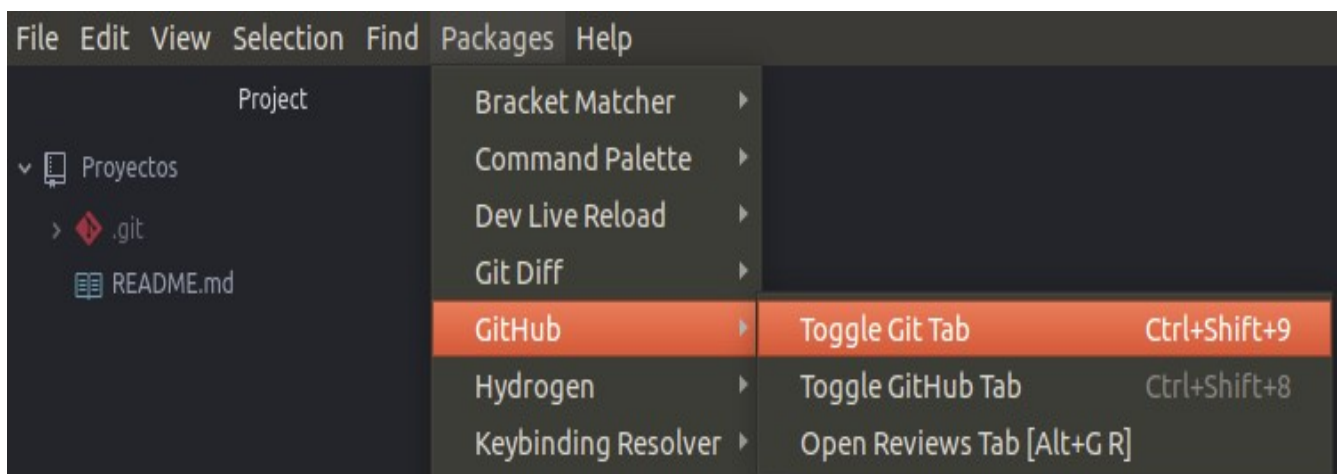


# Subir archivos al repositorio GitHub en Atom

Para subir archivos a nuestro repositorio de “**GitHub**” con **Atom** es necesario tener carpetas y/o archivos creados en la carpeta donde se encuentran los proyectos que deseamos enviar al repositorio remoto.

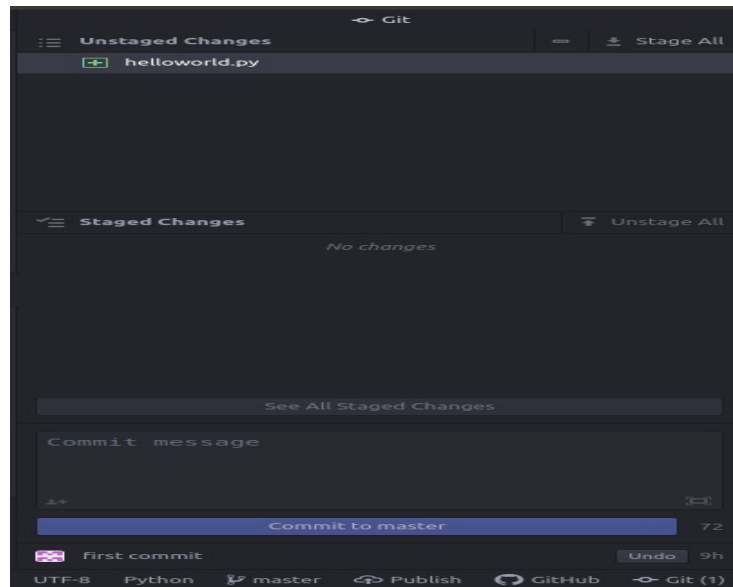


Ir al menu “**Packages**” y seleccionar “**GitHub**” → “**Toggle GitHub Tab**”

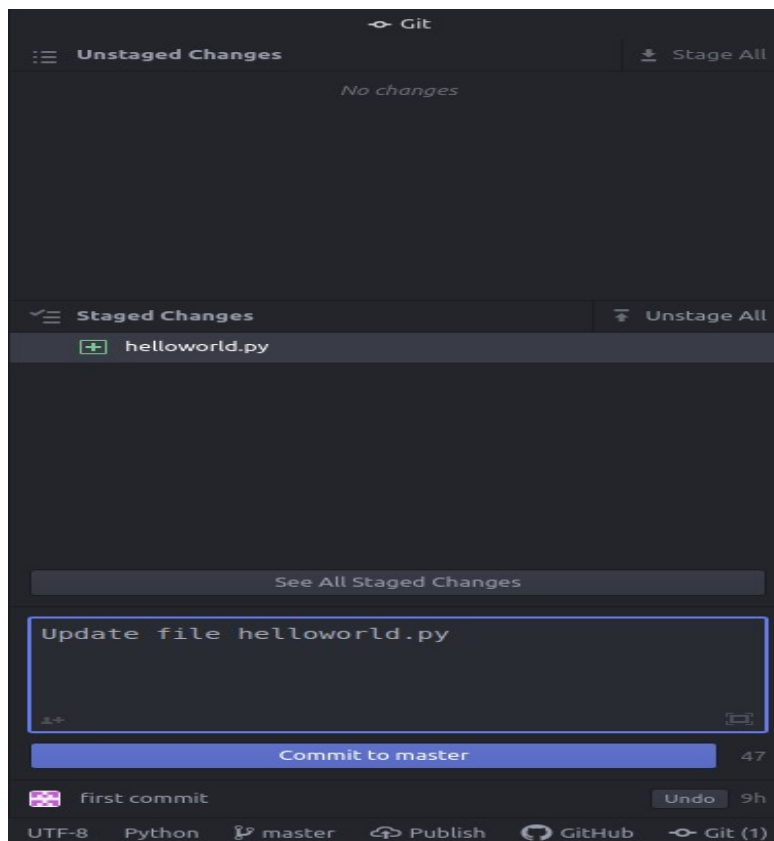




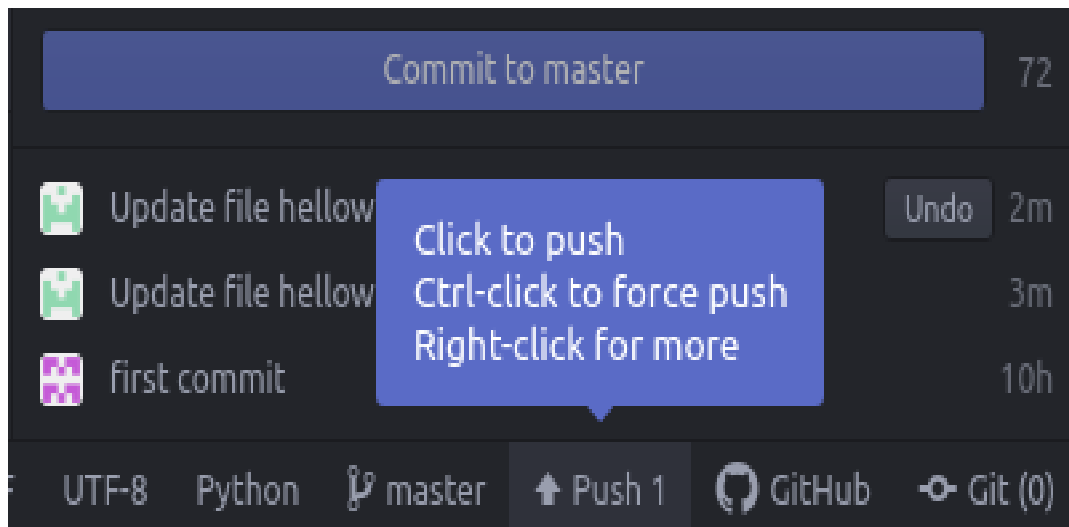
Se nos abra el Panel “**Git**” mostrandonos los archivo y/o archivos que no se han enviado al repositorio remoto.



A continuación, presionaremos el boton “**Stage All**” y el archivo y/o archivos serán movidos a “**Stage Changes**”, después escribiremos su respectivo “**Commit**” para identificar los cambios hechos y presionaremos el botón azul “**Commit to master**”.



Después de presionar el botón “**Commit to master**” podemos observar que el botón “**Publish**” ha cambiado a “**Push**” en donde se debe de presionar para enviar el archivo a nuestro repositorio remoto de “**GitHub**”.



Con esto se puede observar en la página de “<https://github.com/>” en la sección correspondiente al repositorio que el archivo y/o archivos ya se pueden observar en el repositorio.