

# Minitutorial - Crear un repositorio en Github

## Tabla de contenido

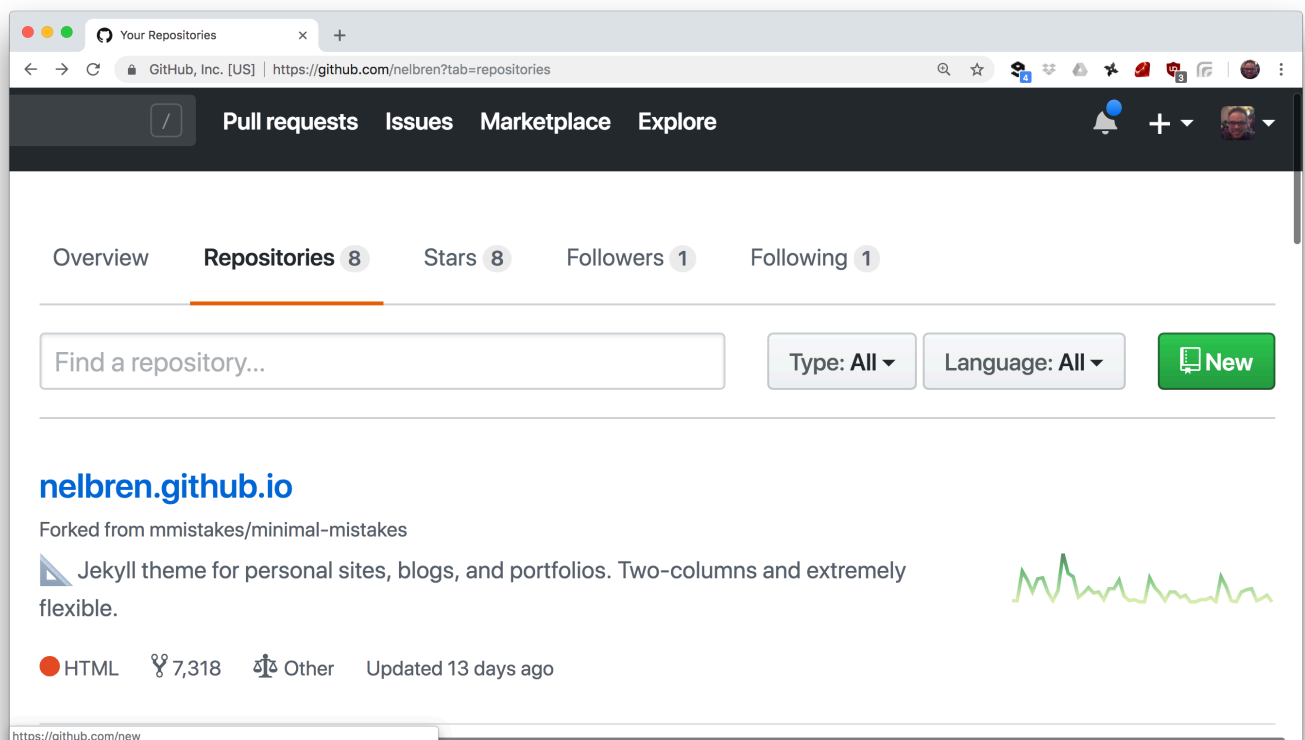
- [1. Requisitos previos](#)
- [2. Crear repositorio en Github](#)
- [3. Configurar llave pública y privada](#)
- [4. Subir archivos al repositorio](#)
- [5. Resultados de cambios](#)

## 1. Requisitos previos

- Cuenta en Github

## 2. Crear repositorio en Github

- Clickea en **New**



- Escribe el nombre del repositorio **mycodesin\_sololearn** y clickea en **Create repository**

Create a new repository

A repository contains all the files for your project, including the revision history.

Owner: **nelbren** / Repository name: **my\_codes\_in\_sololearn** ✓

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

Description (optional)

☒ **Public**  
Anyone can see this repository. You choose who can commit.

☐ **Private**  
You choose who can see and commit to this repository.

☐ **Initialize this repository with a README**  
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: **None** | Add a license: **None** ⓘ

**Create repository**

- Seleccionar el metodo de conexión a github **https** o **ssh**

nelbren / my\_codes\_in\_sololearn

Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

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

Set up in Desktop or **HTTPS** **SSH** `git@github.com:nelbren/my_codes_in_sololearn.git`

We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

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

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

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

```
git remote add origin git@github.com:nelbren/my_codes_in_sololearn.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](#)

**NOTA:** En el caso de un proyecto propio, aconsejo usar **ssh** con el uso de llaves pública y privada.

Al usar https, al obtener cambios del repositorio no se requiere contraseña. Solo es requerida cuando se envien cambios al repositorio.

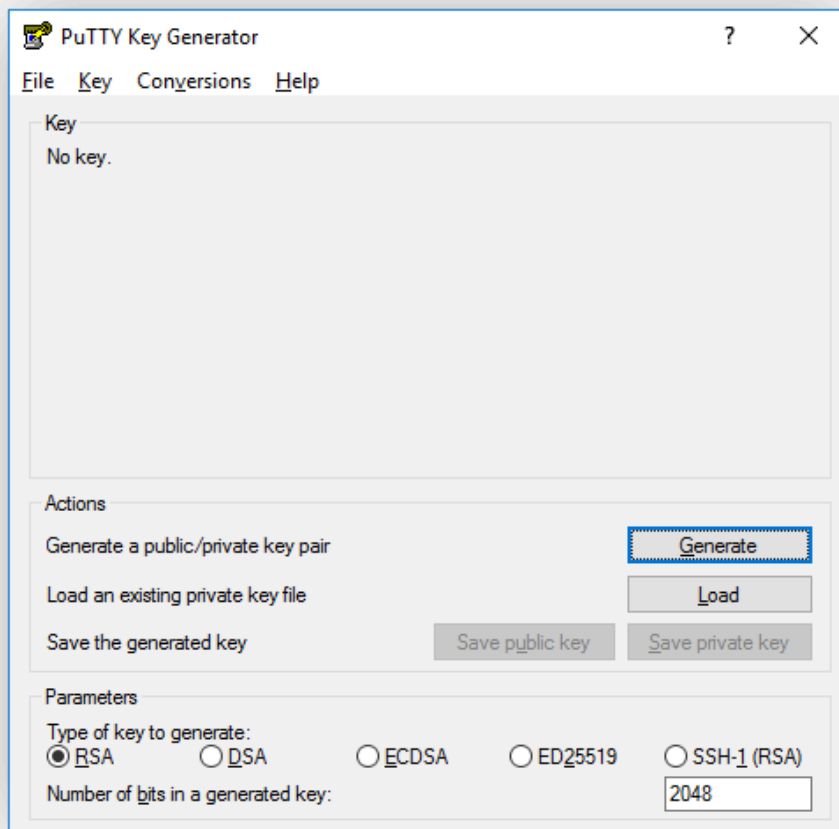
### 3. Configurar llave pública y privada

- Crear llaves usando linea de comando (Linux o MAC)

```
ssh-keygen -t dsa
```

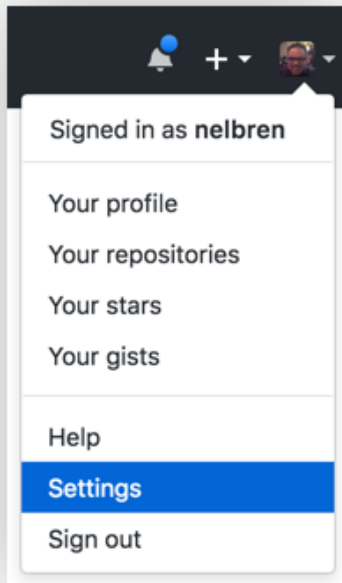
Bash

- Crear llaves usando PuTTYgen (Windows)

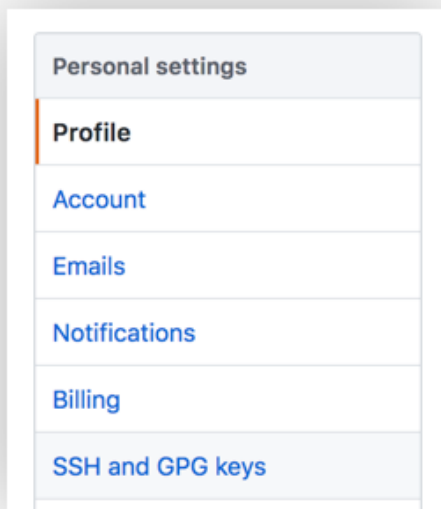


Ver instrucciones en [PUTTYGEN - KEY GENERATOR FOR PUTTY ON WINDOWS](#)

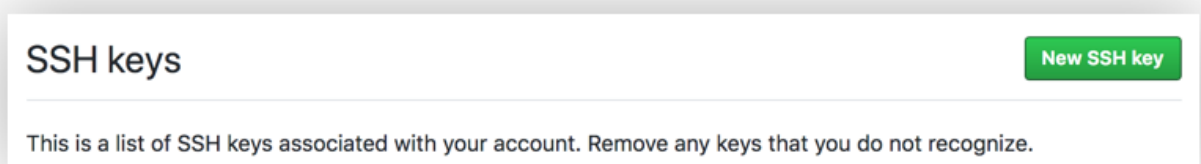
- Asignar llave publica a Github
  - Clickea en **Settings**




- Cliquez en **SSH and GPG keys**



- Cliquez en **New SSH key**



- Copiar la **llave pública** generada al portapapeles



```
2. nelbren@MBP01-5: ~ (zsh)
nelbren@MBP01-5 ~$ cat .ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQACAj1HTT1oeE3DeKpWgSAAEuANQHZJbmjrgv1F3xugEana
mR214bYlKx...
nelbren@MBP01-5 ~$
```

- Pegar la **llave pública** del portapapeles y asignarle un **título**, clicar **Add SSH key**

## SSH keys / Add new


Title

Key

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQACAj1HTT1oeE3DeKpWgSAAEuANQHZJbmjrgv1F3xugEana
mR214bYlKx...
nelbren@MBP01-5 ~$
```

Add SSH key

- Confirmar la asignación con la **contraseña** de la cuenta



Confirm password to continue

Password [Forgot password?](#)

Confirm password

## 4. Subir archivos al repositorio

---

- Por primera vez - [v0.0.1](#)

```
git init
git add .
git commit -m "Mi primera vez!"
git remote add origin git@github.com:nelbren/my_codes_in_sololearn.git
git push -u origin master
```

Bash

```

3. nelbren@npro-vps-01: ~/my_codes_in_sololearn (ssh)
[nelbren@npro-vps-01:~/my_codes_in_sololearn]$ ls
heart.rb          shift_cipher.c.bin  shift_cipher.php    shift_cipher.rb
shift_cipher.c    shift_cipher.c_compile.bash  shift_cipher.py    string_rotations.rb
[nelbren@npro-vps-01:~/my_codes_in_sololearn]$ git init
Initialized empty Git repository in /home/nelbren/my_codes_in_sololearn/.git/
[nelbren@npro-vps-01:~/my_codes_in_sololearn]$ git add .
[nelbren@npro-vps-01:~/my_codes_in_sololearn]$ git commit -m "Mi primera vez!"
[master (root-commit) 3bc139e] Mi primera vez!
 8 files changed, 126 insertions(+)
 create mode 100755 heart.rb
 create mode 100644 shift_cipher.c
 create mode 100755 shift_cipher.c.bin
 create mode 100755 shift_cipher.c_compile.bash
 create mode 100755 shift_cipher.php
 create mode 100755 shift_cipher.py
 create mode 100755 shift_cipher.rb
 create mode 100755 string_rotations.rb
[nelbren@npro-vps-01:~/my_codes_in_sololearn]$ git remote add origin git@github.com:nelbren/
my_codes_in_sololearn.git
[nelbren@npro-vps-01:~/my_codes_in_sololearn]$ git push -u origin master
Counting objects: 10, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (10/10), done.
Writing objects: 100% (10/10), 4.68 KiB | 0 bytes/s, done.
Total 10 (delta 0), reused 0 (delta 0)
remote:
remote: Create a pull request for 'master' on GitHub by visiting:
remote:      https://github.com/nelbren/my_codes_in_sololearn/pull/new/master
remote:
To git@github.com:nelbren/my_codes_in_sololearn.git
 * [new branch]      master -> master
Branch master set up to track remote branch master from origin.
[nelbren@npro-vps-01:~/my_codes_in_sololearn]$

```

- Por segunda vez - [v0.0.2](#)

```

git status
echo shift_cipher.c.bin > .gitignore
git add .
git commit -m "Update to v0.0.2"
git push

```

Bash

```

3. nelbren@npro-vps-01: ~/my_codes_in_sololearn (ssh)
[nelbren@npro-vps-01:~/my_codes_in_sololearn]$ grep 0.0.2 *
heart.rb:# heart.ruby v0.0.2
shift_cipher.c:/* shift_cipher_v0.0.2.c
shift_cipher.rb:# shift_cipher_v0.0.2.rb
string_rotations.rb:# string_rotations_v0.0.2.rb
[nelbren@npro-vps-01:~/my_codes_in_sololearn]$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   heart.rb
        modified:   shift_cipher.c
        modified:   shift_cipher.rb
        modified:   string_rotations.rb

no changes added to commit (use "git add" and/or "git commit -a")
[nelbren@npro-vps-01:~/my_codes_in_sololearn]$ echo shift_cipher.c.bin > .gitignore
[nelbren@npro-vps-01:~/my_codes_in_sololearn]$ git add .
[nelbren@npro-vps-01:~/my_codes_in_sololearn]$ git commit -m "Update to v0.0.2"
[master d32be70] Update to v0.0.2
 5 files changed, 28 insertions(+), 51 deletions(-)
 create mode 100644 .gitignore
 rewrite shift_cipher.c (62%)
[nelbren@npro-vps-01:~/my_codes_in_sololearn]$ git push
Counting objects: 7, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (6/6), done.
Writing objects: 100% (7/7), 1.02 KiB | 0 bytes/s, done.
Total 7 (delta 3), reused 0 (delta 0)
remote: Resolving deltas: 100% (3/3), completed with 3 local objects.
To git@github.com:nelbren/my_codes_in_sololearn.git
 3bc139e..d32be70 master -> master
[nelbren@npro-vps-01:~/my_codes_in_sololearn]$

```

- Por tercera vez - [v0.0.3](#)

```

rm shift_cipher.c.bin
git status
git add .
git commit -m "Update to v0.0.3"
git push

```

Bash



```

3. nelbren@npro-vps-01: ~/my_codes_in_sololearn (ssh)
[nelbren@npro-vps-01:~/my_codes_in_sololearn]$ rm shift_cipher.c.bin
[nelbren@npro-vps-01:~/my_codes_in_sololearn]$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   heart.rb
        deleted:    shift_cipher.c.bin
        modified:   shift_cipher.rb
        modified:   string_rotations.rb

no changes added to commit (use "git add" and/or "git commit -a")
[nelbren@npro-vps-01:~/my_codes_in_sololearn]$ git add .
[nelbren@npro-vps-01:~/my_codes_in_sololearn]$ grep 0.0.3 *
heart.rb:# heart.ruby v0.0.3
shift_cipher.rb:# shift_cipher_v0.0.3.rb
string_rotations.rb:# string_rotations_v0.0.3.rb
[nelbren@npro-vps-01:~/my_codes_in_sololearn]$ git commit -m "Update to v0.0.3"
[master 80414f9] Update to v0.0.3
 4 files changed, 10 insertions(+), 14 deletions(-)
 delete mode 100755 shift_cipher.c.bin
[nelbren@npro-vps-01:~/my_codes_in_sololearn]$ git push
Counting objects: 5, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 611 bytes | 0 bytes/s, done.
Total 5 (delta 3), reused 0 (delta 0)
remote: Resolving deltas: 100% (3/3), completed with 3 local objects.
To git@github.com:nelbren/my_codes_in_sololearn.git
 d32be70..80414f9  master -> master
[nelbren@npro-vps-01:~/my_codes_in_sololearn]$
[nelbren@npro-vps-01:~/my_codes_in_sololearn]$
[nelbren@npro-vps-01:~/my_codes_in_sololearn]$

```

## 5. Resultados de cambios

- [Github](#) (control de versiones) versus [Sololearn](#) (ambiente de ejecución)

Versión	Github	Sololearn	Lenguaje	Descripción
<a href="#">0.0.1</a>	<a href="#">heart.rb</a>	<a href="#">heart.rb_v0.0.1</a>	Ruby	DisplayAHeart-FirstTry!
	<a href="#">shift_cipher.c</a>	<a href="#">shift_cipher_v0.0.1.c</a>	C	UseShiftCipherIn28lines
	<a href="#">shift_cipher.c_compile.bash</a>	-	Bash	Bash4CompileCProgram
	<a href="#">shift_cipher.php</a>	<a href="#">shift_cipher_v0.0.1.php</a>	PHP	UseShiftCipherIn13lines
	<a href="#">shiftl_cipher.py</a>	<a href="#">shift_cipher_v0.0.1.py</a>	Python	UseShiftCipherIn10lines
	<a href="#">shift_cipher.rb</a>	<a href="#">shift_cipher_v0.0.1.rb</a>	Ruby	UseShiftCipherIn20lines
	<a href="#">string_rotations.rb</a>	<a href="#">string_rotations_v0.0.1.rb</a>	Ruby	UsingSubstring
<a href="#">0.0.2</a> <a href="#">diff:0.0.1</a>	<a href="#">heart.rb</a>	<a href="#">heart.rb_v0.0.2</a>	Ruby	DisplayAHeart-UseShortenedIf!
	<a href="#">shift_cipher.c</a>	<a href="#">shift_cipher_v0.0.2.c</a>	C	UseTernaryOperatorIn15lines
	<a href="#">shift_cipher.rb</a>	<a href="#">shift_cipher_v0.0.2.rb</a>	Ruby	UseTernaryOperatorIn13lines
	<a href="#">string_rotations.rb</a>	<a href="#">string_rotations_v0.0.2.rb</a>	Ruby	UsingRotate
<a href="#">0.0.3</a> <a href="#">diff:0.0.2</a>	<a href="#">heart.rb</a>	<a href="#">heart.rb_v0.0.3</a>	Ruby	DisplayAHeart-UseShortBlocks1
	<a href="#">shift_cipher.rb</a>	<a href="#">shift_cipher_v0.0.3.rb</a>	Ruby	UseShorthandOperIn11lines
	<a href="#">string_rotations.rb</a>	<a href="#">string_rotations_v0.0.3.rb</a>	Ruby	UsingRegex
<a href="#">0.0.4</a> <a href="#">diff:0.0.3</a>	<a href="#">heart.rb</a>	<a href="#">heart.rb_v0.0.4</a>	Ruby	DisplayAHeart-UseShortBlocks2
	<a href="#">shift_cipher.rb</a>	<a href="#">shift_cipher_v0.0.4.rb</a>	Ruby	UseBracketSyntaxIn7lines
	<a href="#">string_rotations.rb</a>	<a href="#">string_rotations_v0.0.4.rb</a>	Ruby	UsingSubstringAndOnlyOneAssing
<a href="#">0.0.5</a> <a href="#">diff:0.0.4</a>	<a href="#">heart.rb</a>	<a href="#">heart.rb_v0.0.5</a>	Ruby	DisplayAHeart-UseCase2Array
	<a href="#">string_rotations.rb</a>	<a href="#">string_rotations_v0.0.5.rb</a>	Ruby	UsingSubstringAndOnlyOneAssingInside
<a href="#">0.0.6</a> <a href="#">diff:0.0.5</a>	<a href="#">heart.rb</a>	<a href="#">heart.rb_v0.0.6</a>	Ruby	DisplayAHeart-UseMultipleAssign
	<a href="#">string_rotations.rb</a>	<a href="#">string_rotations_v0.0.6.rb</a>	Ruby	UsingAnCustomFunction
<a href="#">0.0.7</a> <a href="#">diff:0.0.6</a>	<a href="#">heart.rb</a>	<a href="#">heart.rb_v0.0.7</a>	Ruby	DisplayAHeart-GlobalVarsToLocal
<a href="#">0.0.8</a> <a href="#">diff:0.0.7</a>	<a href="#">heart.rb</a>	<a href="#">heart.rb_v0.0.8</a>	Ruby	DisplayAHeart-Array2NewAssign"
<a href="#">0.0.9</a> <a href="#">diff:0.0.8</a>	<a href="#">heart.rb</a>	<a href="#">heart.rb_v0.0.9</a>	Ruby	DisplayAHeart-ReplaceIfForTwolf

