

APLICACIONES DE LA CRIPTOGRAFÍA. PRÁCTICAS

SSH

Poder utilizar el servicio de GITHUB a través de SSH.

Para esta práctica vamos a necesitar crear una cuenta en GITHUB (si ya tenéis una podéis utilizar esa)

Ideas claves:

- Instalar git en Ubuntu/Xubuntu o GIT
- Fork de un repositorio
- Generar las claves ssh si no existen, y agregarlas GITHUB.
- Hacer un cambio en un fichero, commit y subir el cambio a github a través de SSH.

Recursos:

- [Escenario Katacoda de SSH básico.](#)
- [Escenario Katacoda de SSH intermedio.](#)
- [Escenario Katacoda de GIT básico.](#)
- [Chuleta de los comandos GIT.](#)

Ejercicios

Ejercicio 1

Instala GIT en ubuntu/xubuntu/windows. Si ya está instalado, muestra los comandos que serían necesarios para la instalación.

Para instalar Git en Windows es necesario ir a su página web y descargarse un instalador, y ejecutarlo.

Por otra parte, en Linux se puede hacer mediante comandos en la consola. He encontrado una web muy interesante donde se explican los diferentes comandos y pasos a seguir para la instalación y configuración de tu nombre en Git, para los principales SO, la cual tiene un diseño muy pulcro y su contenido está muy claro.

<https://www.atlassian.com/es/git/tutorials/install-git>

En nuestro caso, hemos optado por descargar una consola llamada “cmder” la cual incluye también la instalación de Git en nuestro Windows

Enlace de descarga a cmder: <https://cmder.net/>

| | | |
|--|---|---|
| With help of the best | Total portability | Git and others |
| Think about cmder more as a software package than a separate app. All the magic is happening through ConEmu. With enhancements from Clink. | Carry it with you on a USB stick or in the Cloud, so your settings, aliases and history can go anywhere you go. You will not see that ugly Windows prompt ever again. | Oooh yes! If you decide to use the slightly bigger git-for-windows version, you will have all Unix commands ready in PATH so that you can <code>git init</code> or <code>cat</code> instantly on every machine. |

Download

Latest Version: v1.3.18

[GitHub Repository](#) (for issues/requests/source)

[Download Mini](#) ~9.8MB

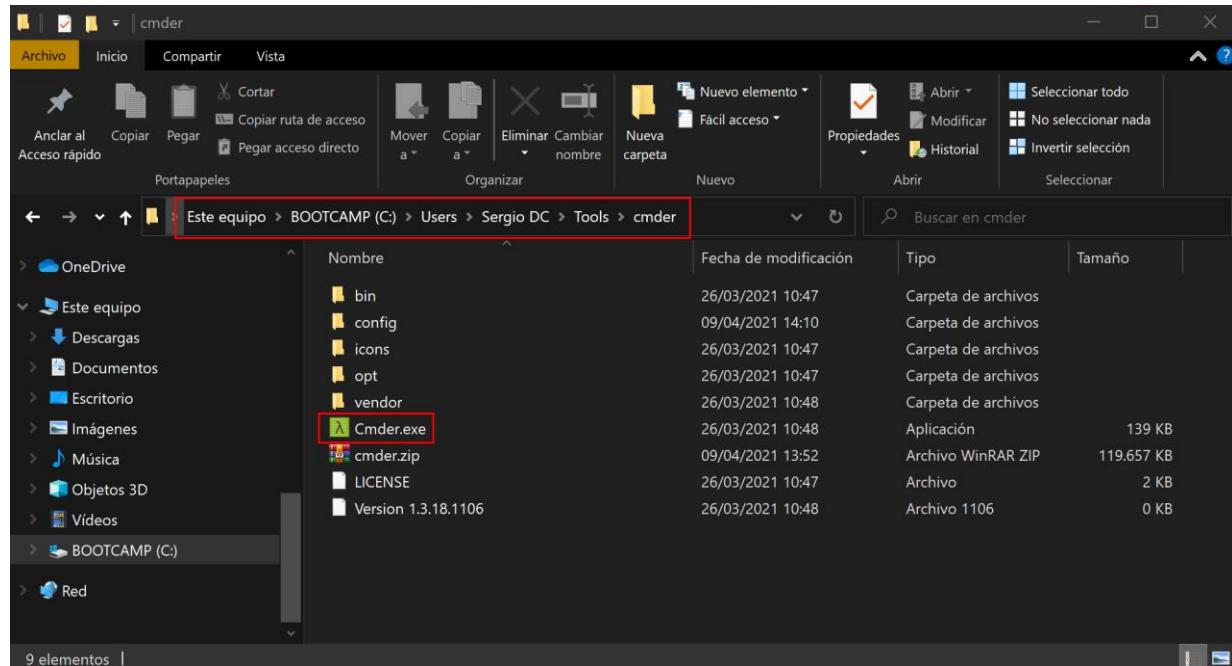
[Download Full](#) (with Git for Windows) 117MB or ~63MB 7z

Installation

1. Unzip
2. (optional) Place your own executable files into the `bin` folder to be injected into your PATH.
3. Run **Cmder** (`Cmder.exe`)

Keyboard shortcuts

La descarga nos brinda un archivo .zip, el cual hay que descomprimirlo en un subdirectorio previamente creado por nosotros (Tools -> cmder) en nuestro disco (C:), y tras ello, ejecutar el archivo `Cmder.exe`.



Ejercicio 2

Crearos un usuario en GITHUB (si ya lo tenéis utilizar ese), y vamos a crear un fork de este [repositorio](#) sobre seguridad informática.

¿Cómo realizar el fork? Seguir la [ayuda del propio github](#).

This screenshot shows the GitHub repository page for 'mattaereal / seginfoFAQ'. At the top right, there is a red box highlighting the 'Fork' button, which has a value of '25' next to it. The repository description is 'FAQ del mundo de la seguridad informática en español.' and it includes tags like 'hacking', 'seguridad', and 'introducción'.

This screenshot shows the GitHub user profile for 'Royal6969'. In the 'Repositories' tab, there is a red box highlighting the repository 'seginFAQ', which was forked from 'mattaereal/seginFAQ'. The repository description is 'FAQ del mundo de la seguridad informática en español.' and it was last updated on 27 Jan.

This screenshot shows the GitHub repository page for 'Royal6969 / seginfoFAQ'. At the top left, there is a red box highlighting the repository name. The repository details show it was forked from 'mattaereal/seginFAQ'. The repository description is 'FAQ del mundo de la seguridad informática en español.' and it includes a 'Settings' tab.

Ejercicio 3

Una vez creado el Fork, el repositorio ya es vuestro y por tanto vais a poder modificarlo a vuestro gusto. Para ello lo que vamos a hacer ahora, es clonarlo en local, en vuestro ordenador, a través de la consola.

Pero para poder acceder a GITHUB desde la consola, lo podemos hacer utilizando HTTP, donde nos pedirá nuestro usuario/pass, o a través de SSH.

Vosotros lo vais a hacer a través de SSH.

Para ello necesitaremos crear una clave SSH, para ello seguir esta [ayuda](#).

Una vez creada la clave SSH, tenemos que [agregar la clave pública](#) de esta clave a nuestra cuenta de GITHUB.

Una vez hecho esto, ya podéis probar si funciona. Para ello podéis hacerlo [de esta forma](#).

Una vez todo configurado, procedemos a realizar el [clonado de nuestro repositorio](#).

Empezamos creando un subdirectorio (oculto) dentro de la carpeta de nuestro usuario del disco (C):.

```
C:\Users\Sergio DC
λ ls -a
./
../
.argouml/
.config/
.dia/
.eclipse/
.p2/
.packettracer
.ssh/ .ssh
.tooling/
.VirtualBox/
.vscode/
'3D Objects'
.AppData/
'Cisco Packet Tracer 8.0'
'Configuración local'@
.Contacts/
.Cookies@
'Datos de programa'@
/Desktop/
/Documents/
/Downloads/
/eclipse/
/eclipse-workspace/
'Entorno de red'@
/Favorites/
/Impresoras@
λ cmd.exe
```

```
C:\Users\Sergio DC
λ cd .ssh
C:\Users\Sergio DC\.ssh
λ cmd.exe
```

```
C:\Users\Sergio DC\.ssh
λ ssh-keygen -t rsa -b 4096 -C "sergiodiazcampos@gmail.com"
Generating public/private rsa key pair.
Enter file in which to save the key (C:\Users\Sergio DC/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in C:\Users\Sergio DC/.ssh/id_rsa.
Your public key has been saved in C:\Users\Sergio DC/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:lrcAONqfYr9eYaSHgg6xEIxBW2VuPgm8aSV1J0VHdLA sergiodiazcampos@gmail.com
The key's randomart image is:
+---[RSA 4096]---+
|++o..+ ooo=.. |
| o+ +.. o . o |
| o. +o+. . E |
| .o +o..= . |
| o o+o+o S . |
| o. o.= + . |
| . o o . . |
| . o . . |
| .+.
+---[SHA256]---+
C:\Users\Sergio DC\.ssh
λ ls -a
./ .. id_rsa id_rsa.pub
```

C:\Users\Sergio DC\.ssh

C:\Users\Sergio DC\.ssh

```
λ cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQADY1L/tYofPYQWH+kwrTYaG4Cy2oZ3w1e8TQ1zUcNbM5A4j4eKrHEHJnSFkToZXwKRYA5f1dSuASz/3bawg
0CXyFVwVWCkI49XOHbNK4Za8xobhz7RaeM7RFYScVnuKixLBYfPgrpFV+xDDnFgUZjkPw4tNYUYNRUkrffrIKEpd5FBv/mQKqpWpA/GZj9/fw3aZzRoq1d
H78Gi6+arepr5veSpukFB8HPuJpCtCKTroffFnLdfNoSrkJdeSv7S9+yTHtrDazzZOMlgp8dvKKitemUqlVgGPL9XH3CYRduucb1cSwm6ooXfYH0f9UZom
ltEZ6aBp+szA8V3G+D3MhsQWlvWrCFYL7jtdX4y+oWlpqrqZN9y6E9np3AkCX3xNMZ3XBAQVELg1H9kjjccbnZ7VSppBn70oa6o3i7wFlJHiE2JMHTuKzVgSZ
Cc+Hpp/mckHuWy2avCTgra+kf1svZXDQJ/9921nmFoTpGdmJRKf6lquzDCnTzfN1npjnbpV5uMKzFW01or7U3vmAFA12MpvybK1Q0CvlRMcsuzIRf32/T9t
WoLBoIySgatge4An2U/Aml+a/IxCnzmvExed7N0AdNcIVz8zVaCtTf2QICG6SWr/ZGKwsqZUkzTEm/yH3C270PU50JymVm4ub0dPLybdbYeKaPZMGFbky3yD
XQ== sergiodiazcampos@gmail.com
```

C:\Users\Sergio DC\.ssh

The screenshot shows the GitHub account settings for user Royal6969. The left sidebar has a red box around the 'SSH and GPG keys' link. The main area shows the 'SSH keys' section with a green 'New SSH key' button at the top right, which is also highlighted with a red box.

The screenshot shows the 'SSH keys / Add new' page. The 'Title' field contains 'pc-1dam' and the 'Key' field contains a long SSH RSA key. The 'Add SSH key' button at the bottom is highlighted with a red box.

The screenshot shows the 'SSH keys' section after adding the key. It lists one key: 'pc-1dam' (SHA256:1rcAONqfYr9eYaShgg6xEbxBw2VuPgm8aSV10V, Added on 9 Apr 2021, Never used — Read/write). The 'Delete' button next to it is highlighted with a red box. The 'New SSH key' button is also visible.

Probar tu conexión SSH

Después de haber configurado tu clave SSH y haberla agregado a tu GitHub cuenta, puedes probar tu conexión.

Mac Windows Linux

Antes de probar tu conexión SSH, debes haber hecho lo siguiente:

- Comprobado tus claves SSH existentes
- Generado una clave SSH nueva
- Agregado una clave SSH nueva a tu cuenta de GitHub

Cuando pruebas tu conexión, tendrás que autenticar esta acción utilizando tu contraseña, que es la contraseña de clave SSH que ya creaste. Para obtener más información acerca de trabajar con contraseñas de clave SSH, consulta "Trabajar con contraseñas de clave SSH".

- 1 Abre la Git Bash.
- 2 Ingresa lo siguiente:

```
$ ssh -T git@github.com
# Attempts to ssh to GitHub
```

Puedes ver una advertencia como la siguiente:

```
C:\Users\Sergio DC\.ssh
λ ssh -T git@github.com
ssh: connect to host github.com port 22: Connection timed out

C:\Users\Sergio DC\.ssh
λ
cmd.exe
```

En este momento, tratamos de hacer la conexión mediante wifi, y las restricciones del mismo wifi nos lo impiden, con lo que tendremos que volver a probar usando una conexión LAN, con el cable del RJ45.

```
C:\Users\Sergio DC\.ssh
λ ----[SHA256]-----
C:\Users\Sergio DC\.ssh
λ ls -a
./ ..\ id_rsa id_rsa.pub

C:\Users\Sergio DC\.ssh
λ cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQADYLL/tYofPYQWH+kwrTYaG4Cy2oZ3w1e8TQ1zUcNbM5A4j4eKrHEHJnSFkToZxwKRYA5f1dSuASz/3bawg
0CXyhFvWvCkI49XHDN4K4zaxobhz7RaeM7RFYSVcVuNkix1BYfPgrpFV+xDdnFgUZjkPw4tNYUYNRUkrffR1KEpd5FBv/mQkqpWpA/GZj9/fw3aZzRoq1d
H78Gi6+arepr5veSpukFB8HPuJLpCtCKTroffN1ldfNo5rkfdeSv7S9+yTdrDazZOM1gp8dvKKitemUqlVyGVPGL9XH3CYRduucb1cSwm6ooXfyH0f9UZom
ltEZ6aBp+szA8V3G+D3MhsQwvWrCFYL7jtdX4y+oQwlporqZN9y6E9np3AkCX3xNMZ3XBAQVElg1H9kjccbnZ7SppBn70oa6o3i7wFlJHiE2JMHtUkZvgS2
Cc+Hpp+mckHuWy2avCTgr+aFlsvZXDOJ/9921nmFotpGDMJRKf61lquzDCnTzf1npjnbpVsUmkzfW01or7U3vnAFa12MpygYbK1Q0cvIRMsuzIRf32/T9t
WolBoiySgatge4An2U/Aml+a/IxCnzmVExed7N0AdNcIVz8zVaCtf2QICG6Swr/ZGKwsqZUkzTEm/yH3C270Pu50JymVm4ub0dPlYbdbYeKaPZMGFbk3yD
XQ== sergiodiazcampos@gmail.com

C:\Users\Sergio DC\.ssh
λ ssh -T git@github.com
ssh: connect to host github.com port 22: Connection timed out

C:\Users\Sergio DC\.ssh
λ ssh -T git@github.com
The authenticity of host 'github.com (140.82.121.4)' can't be established.
RSA key fingerprint is SHA256:nThbg6kXUpJWGl7E1IGOCspRomTxCARLviKw6E5SY8.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'github.com,140.82.121.4' (RSA) to the list of known hosts.
Hi Royal6969! You've successfully authenticated, but GitHub does not provide shell access.

C:\Users\Sergio DC\.ssh
λ
cmd.exe
```

```
C:\Users\Sergio DC\Tools\cmder
λ ls -al ~/.ssh
total 29
drwxr-xr-x 1 Sergio DC 197121    0 abr.  9 14:38 ./
drwxr-xr-x 1 Sergio DC 197121    0 abr.  9 14:21 ../
-rw-r--r-- 1 Sergio DC 197121 3243 abr.  9 14:25 id_rsa
-rw-r--r-- 1 Sergio DC 197121  753 abr.  9 14:25 id_rsa.pub
-rw-r--r-- 1 Sergio DC 197121   406 abr.  9 14:38 known_hosts

C:\Users\Sergio DC\Tools\cmder
λ 
```

```
C:\Users\Sergio DC\Tools\cmder
λ ssh -T git@github.com
Hi Royal6969! You've successfully authenticated, but GitHub does not provide shell access.

C:\Users\Sergio DC\Tools\cmder
λ 
```

mattaereal / seginfoFAQ

Code Issues 2 Pull requests 2 Actions Projects Wiki Security Insights

master 1 branch 0 tags Go to file Add file Code

mattaereal Added ekoparty's yt channel 4b91a4f on 14 Sep 2020 35 commits

| | | |
|--------------------------------------|-----------------------------|---------------|
| Cybersecurity Domains v2 Spanish.png | Cybersecurity Domain v2 | 13 months ago |
| Glosario.md | Adding Glosario.md | 11 months ago |
| README.md | Added ekoparty's yt channel | 7 months ago |

Go to file Add file Code

Clone ?

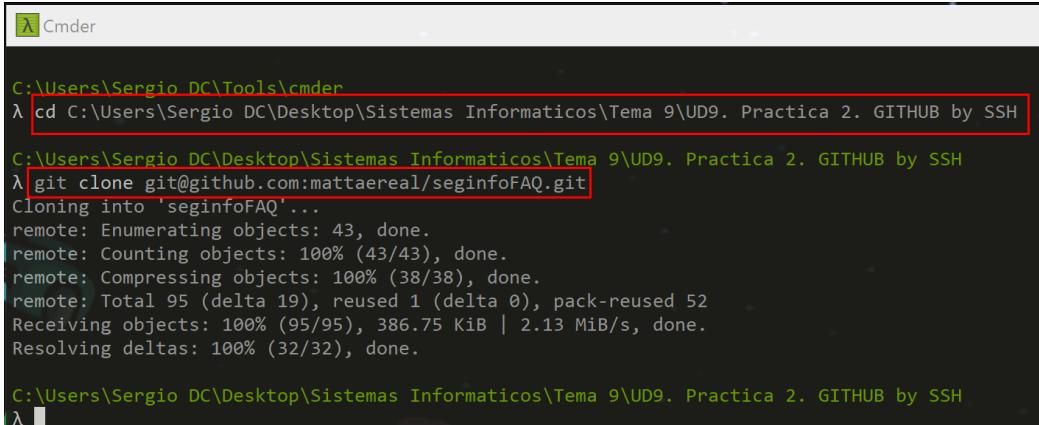
HTTPS **SSH** GitHub CLI New

git@github.com:mattaereal/seginfoFAQ.git

Use a password-protected SSH key.

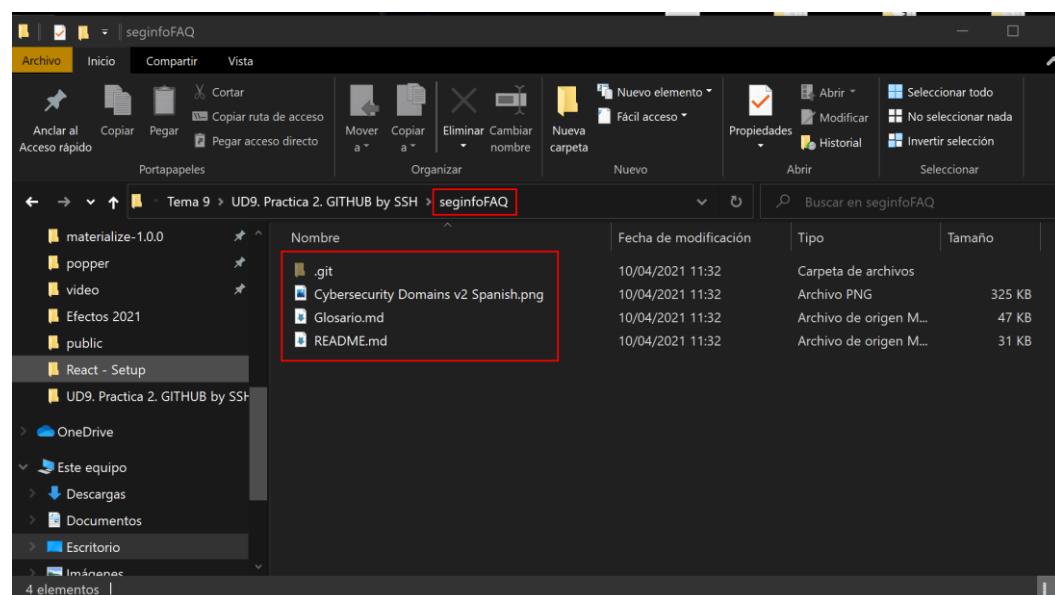
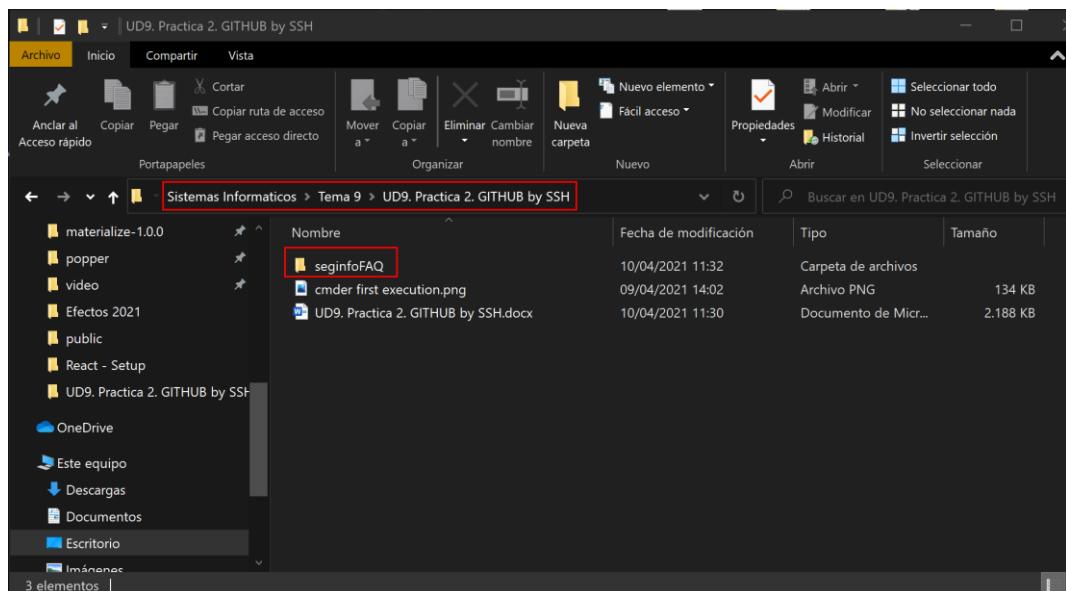
Open with GitHub Desktop

Download ZIP



```
C:\Users\Sergio DC\Tools\cmder
λ cd C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH
λ git clone git@github.com:mattaereal/seginformationFAQ.git
Cloning into 'seginformationFAQ'...
remote: Enumerating objects: 43, done.
remote: Counting objects: 100% (43/43), done.
remote: Compressing objects: 100% (38/38), done.
remote: Total 95 (delta 19), reused 1 (delta 0), pack-reused 52
Receiving objects: 100% (95/95), 386.75 KiB | 2.13 MiB/s, done.
Resolving deltas: 100% (32/32), done.

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH
λ
```



Ejercicio 4

Una vez clonado, vamos a realizar alguna modificación al mismo, simplemente modificando el fichero README.md, disponible en la raíz del mismo.

Este fichero tiene el formato [markdown](#), que es un formato de texto muy utilizado por los desarrolladores de código, para poder dar formato al texto, utilizando simplemente caracteres, lo que permite ser texto plano y no depender de un formato propietario.

Una vez modificado, vamos a confirmar el cambio en nuestro repositorio local. A confirmar o agregar un cambio a un fichero o varios, en GIT se llama crear un commit.

Para ello, tenemos que seleccionar el fichero modificado y después realizar el commit.

Una vez realizado el commit en local, toca subirlo hacia nuestro repositorio remoto (github).

Para realizar estos pasos seguir [esta guía](#).

```

 README.md 
  Archivo Editar Selección Ver Ir Ejecutar Terminal Ayuda README.md - Visual Studio Code
  README.md ...
  1 *Nota: Actualmente se encuentra en construcción.*
  2
  3 *Nota: Estoy añadiendo esta linea, y por tanto, modificando este archivo.
  4
  5 ## Tabla de contenidos
  6 1. [Acerca del FAQ](#acerca)
  7 2. [Disclaimer](#disclaimer)
  8 3. [¿Quiénes somos?](#quienes)
  9 4. [FAQ](#faq)

```

```

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginformation
A git commit -m "Add existing file"
Author identity unknown

*** Please tell me who you are.

Run

git config --global user.email "you@example.com"
git config --global user.name "Your Name"

to set your account's default identity.
Omit --global to set the identity only in this repository.

fatal: unable to auto-detect email address (got 'Sergio DC@SergioDiaz.(none)')

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginformation
A git config --global user.email "sergiodiazcampos@gmail.com"
git config --global user.name "Royal6969"

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginformation
A git commit -m "Add existing file"
[master bec2d11] Add existing file
 1 file changed, 2 insertions(+)

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginformation
A

```

```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ git pull
warning: Pulling without specifying how to reconcile divergent branches is
discouraged. You can squelch this message by running one of the following
commands sometime before your next pull:

git config pull.rebase false # merge (the default strategy)
git config pull.rebase true # rebase
git config pull.ff only # fast-forward only

You can replace "git config" with "git config --global" to set a default
preference for all repositories. You can also pass --rebase, --no-rebase,
or --ff-only on the command line to override the configured default per
invocation.

Already up to date.

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ git checkout -b Prueba Branch Royal6969/seginfoFAQ
Switched to a new branch 'Prueba_Branch_Royal6969/seginfoFAQ'

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal69
69/seginfoFAQ)
λ git push origin Prueba_Branch_Royal6969/seginfoFAQ
ERROR: Permission to mattaereal/seginfoFAQ.git denied to Royal6969.
fatal: Could not read from remote repository.

Please make sure you have the correct access rights
and the repository exists.

cmd.exe
```

```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal69
69/seginfoFAQ)
λ git pull
warning: Pulling without specifying how to reconcile divergent branches is
discouraged. You can squelch this message by running one of the following
commands sometime before your next pull:

git config pull.rebase false # merge (the default strategy)
git config pull.rebase true # rebase
git config pull.ff only # fast-forward only

You can replace "git config" with "git config --global" to set a default
preference for all repositories. You can also pass --rebase, --no-rebase,
or --ff-only on the command line to override the configured default per
invocation.

There is no tracking information for the current branch.
Please specify which branch you want to merge with.
See git-pull(1) for details.

git pull <remote> <branch>

If you wish to set tracking information for this branch you can do so with:

git branch --set-upstream-to=origin/<branch> Prueba_Branch_Royal6969/seginfoFAQ

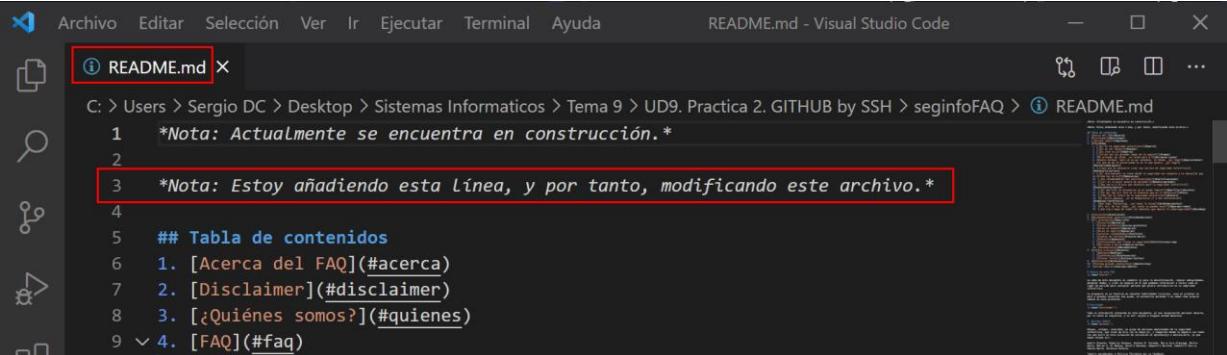
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal69
69/seginfoFAQ)
λ git push origin Prueba_Branch_Royal6969/seginfoFAQ
ERROR: Permission to mattaereal/seginfoFAQ.git denied to Royal6969.
fatal: Could not read from remote repository.

Please make sure you have the correct access rights
and the repository exists.

cmd.exe
```

Aquí me quedo atascado, porque parece que no tengo el permiso para hacer esta acción, así que investigaremos por otra parte y volveremos hacer el proceso correctamente.

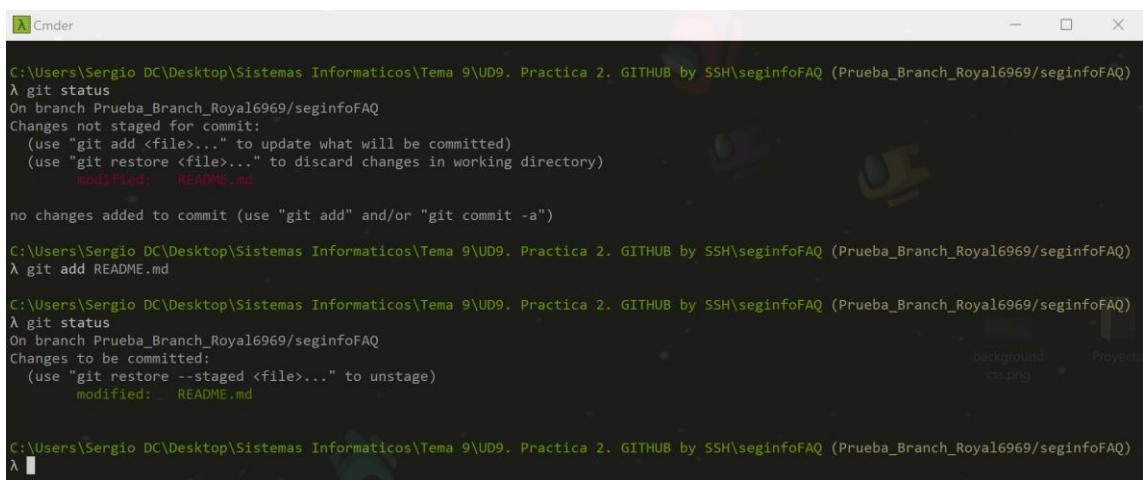
A partir de aquí, voy a crear los commits y un branch bajo la guía y siguiendo los pasos de un youtuber al que admiro mucho, “The Net Ninja” en un mini-curso que tiene sobre esto, en la lista de su canal “Git & GitHub Tutorial for beginners”.



```

1 *Nota: Actualmente se encuentra en construcción.*
2
3 *Nota: Estoy añadiendo esta Línea, y por tanto, modificando este archivo.*
4
5 ## Tabla de contenidos
6 1. [Acerca del FAQ](#acerca)
7 2. [Disclaimer](#disclaimer)
8 3. [¿Quiénes somos?](#quienes)
9 4. [FAQ](#faq)

```



```

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969\seginfoFAQ)
λ git status
On branch Prueba_Branch_Royal6969\seginfoFAQ
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified: README.md

no changes added to commit (use "git add" and/or "git commit -a")

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969\seginfoFAQ)
λ git add README.md

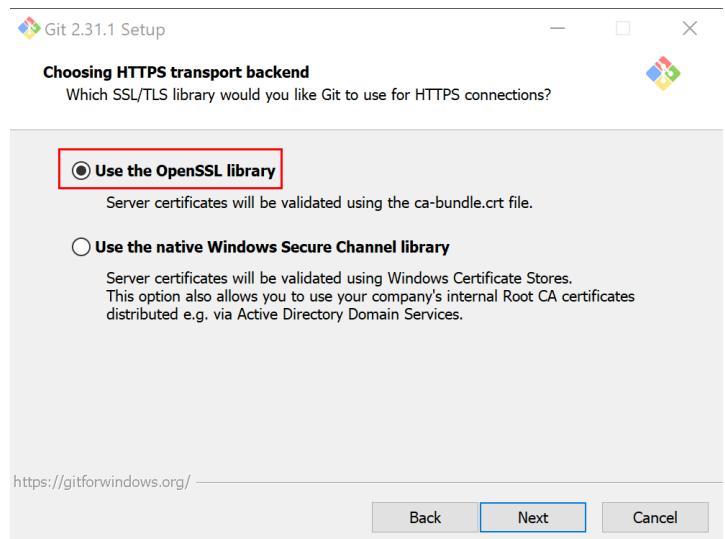
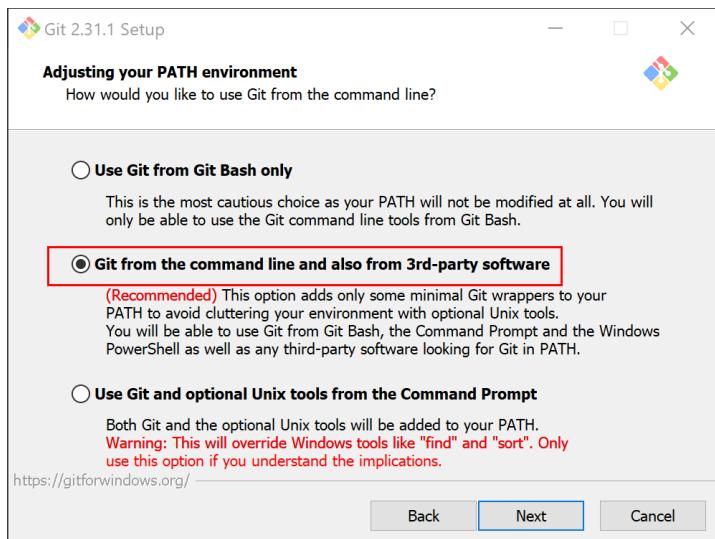
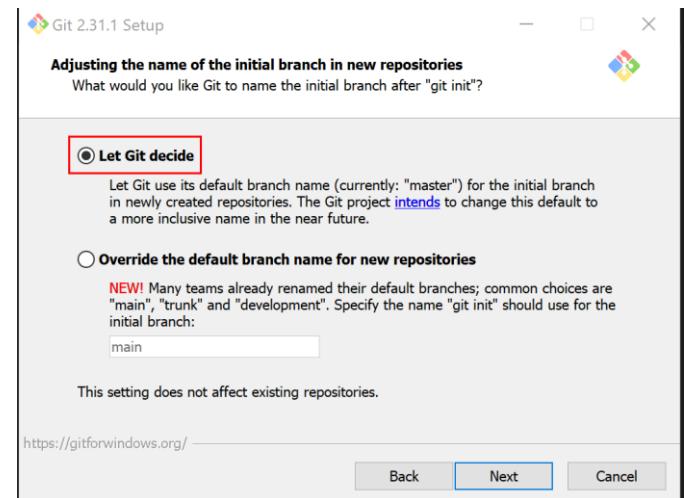
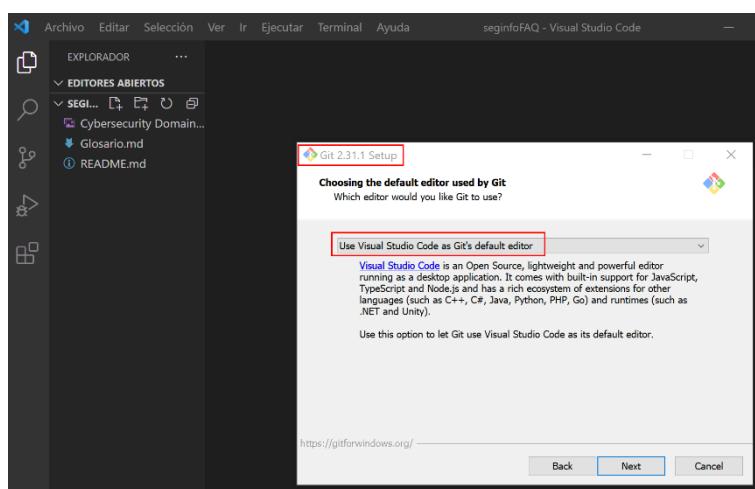
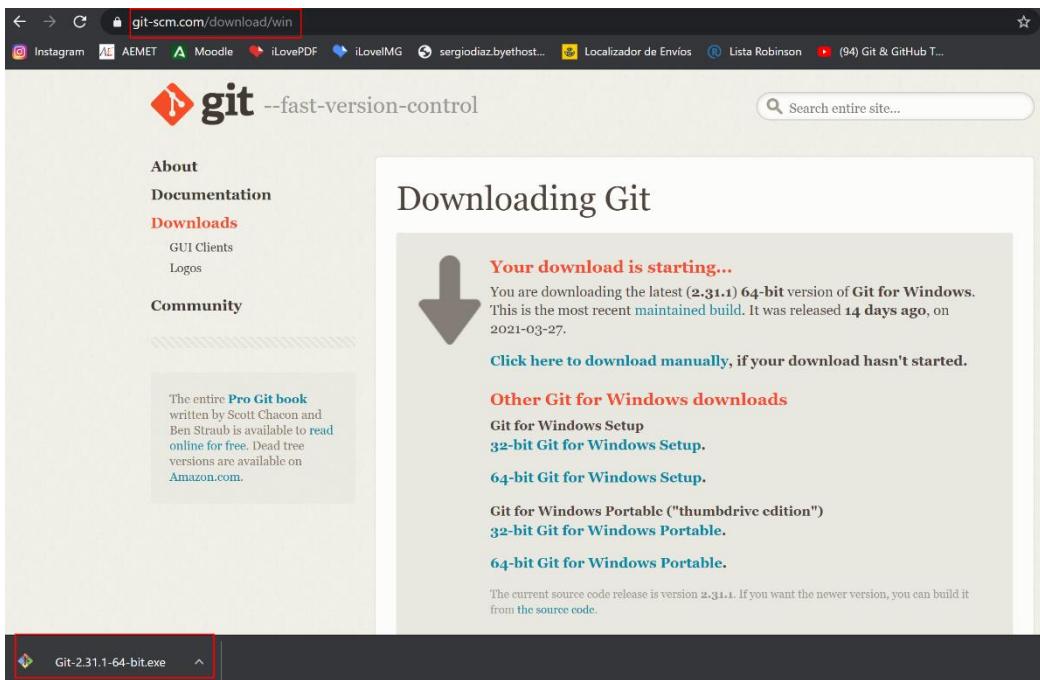
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969\seginfoFAQ)
λ git status
On branch Prueba_Branch_Royal6969\seginfoFAQ
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified: README.md

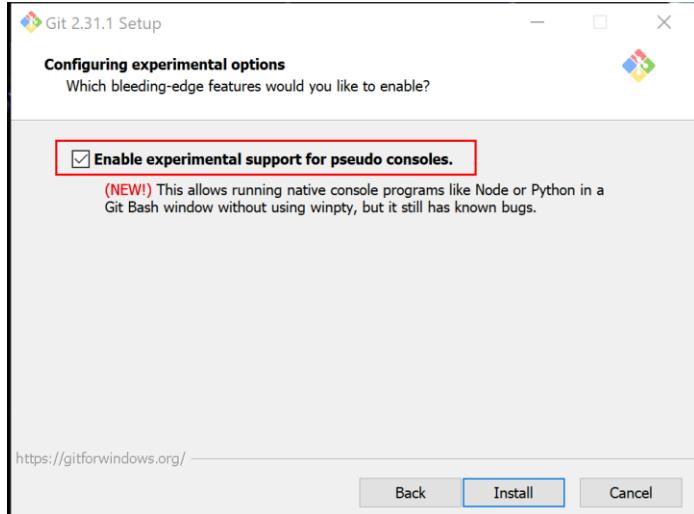
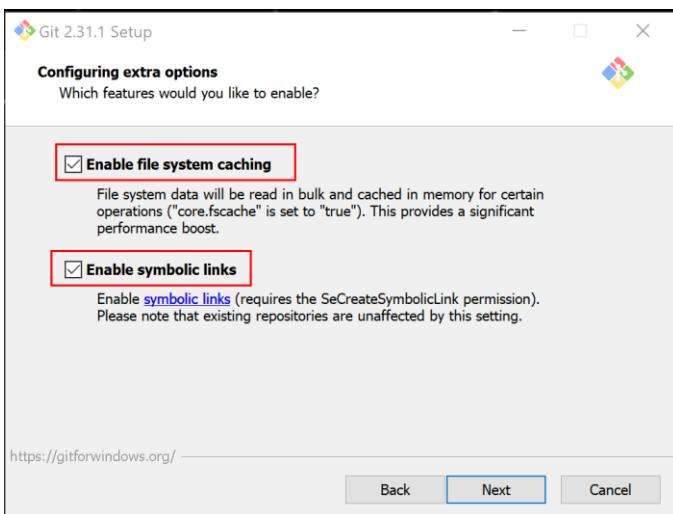
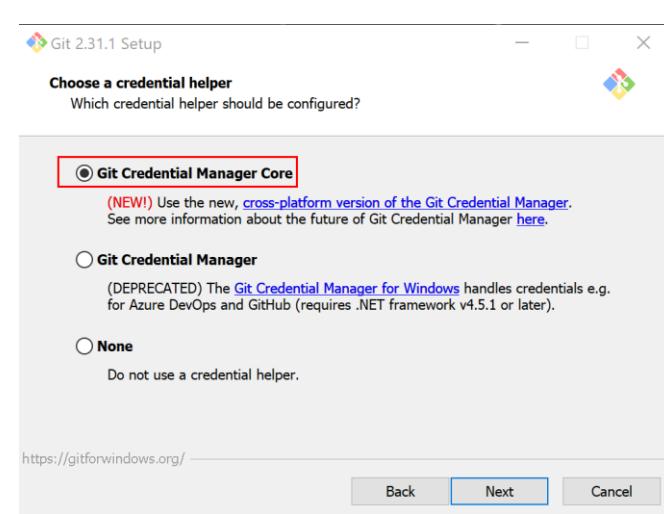
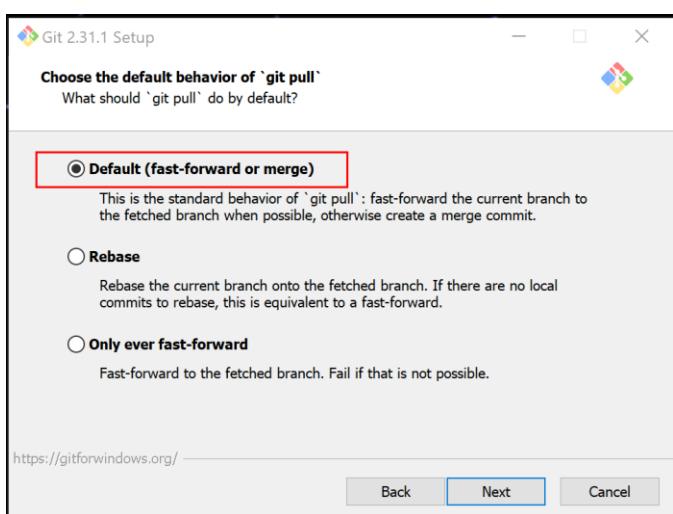
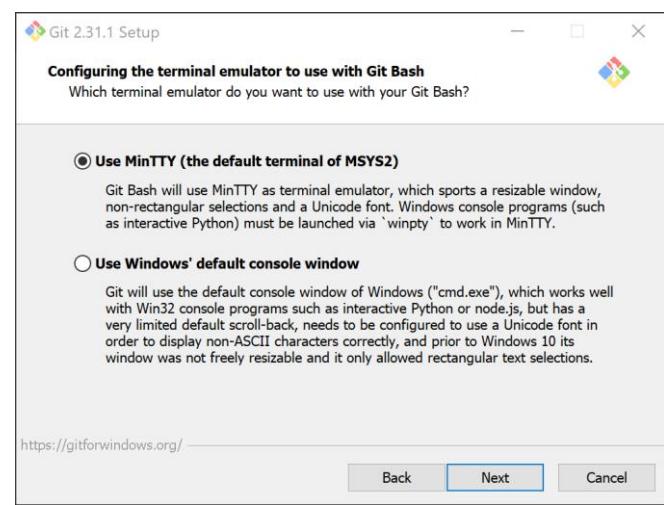
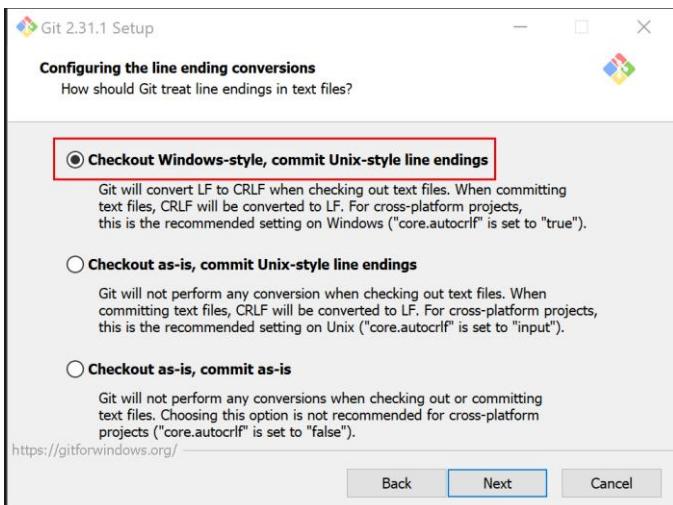
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969\seginfoFAQ)
λ

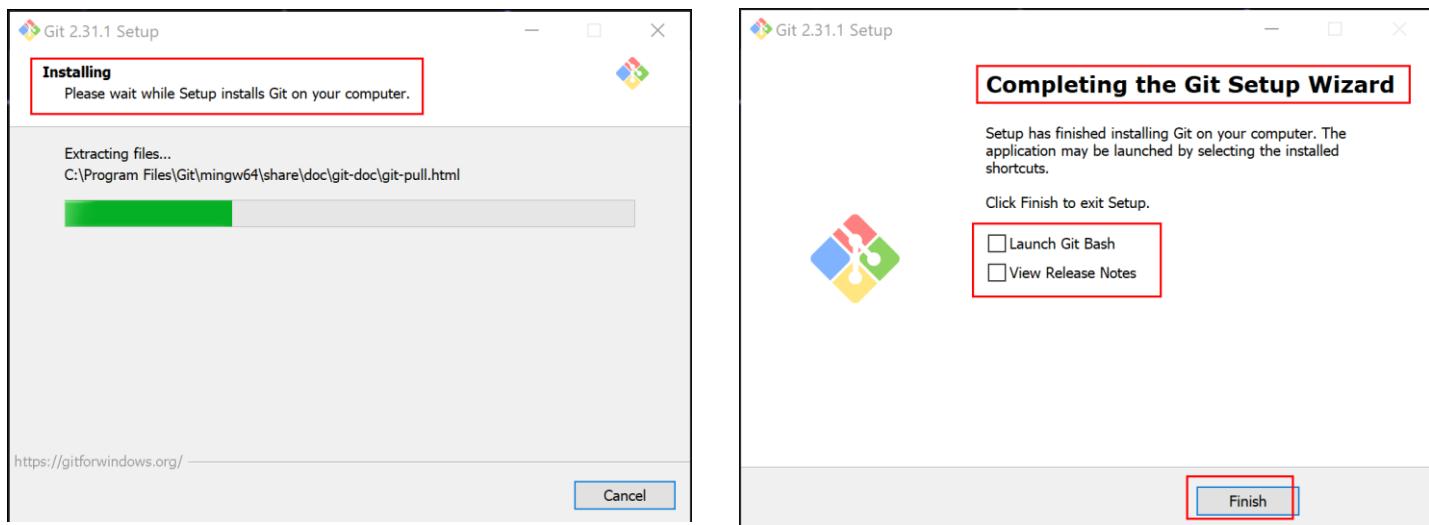
```

En este momento, cuando a través del Visual Studio abro la carpeta de `seginfoFAQ`, éste me lanza un mensaje en el que me dice que es necesario que instale Git para vincularlo con él...

Yo pensaba que con la instalación de `cmder` ya tenía Git, pero se ve que no es así del todo. Entonces hice click en tal mensaje, y me redireccionó a la web oficial donde se descarga Git, y lo descargué e instalé marcando las siguientes opciones de configuración.

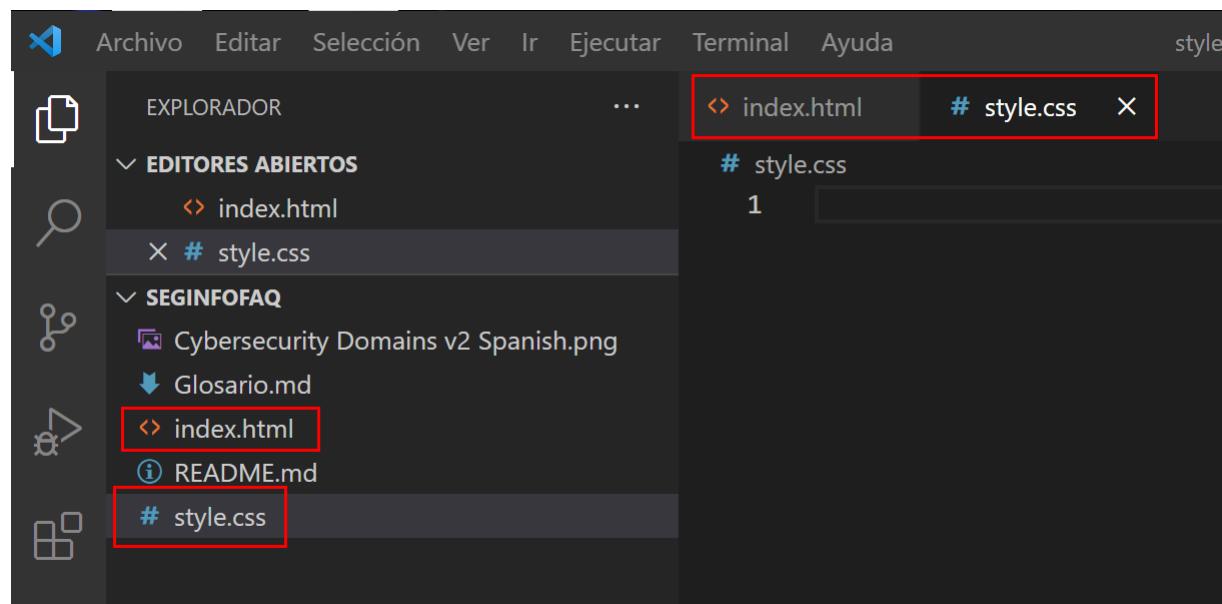






Nota: estas opciones marcadas de configuración previa las he marcado yo mismo bajo mi propio criterio. The Net Ninja no ha hecho esto en su video.

Una vez hecho instalado Git, volvemos con The Net Ninja, y creamos dos archivos dentro de la carpeta de `seginfoFAQ`.



The screenshot shows the Visual Studio Code interface. On the left is the Explorer sidebar with a tree view of files. Under 'EDITORES ABIERTOS', 'index.html' and '# style.css' are listed. Under 'SEGINFOFAQ', there are files: 'Cybersecurity Domains v2 Spanish.png', 'Glosario.md', 'index.html', 'README.md', and '# style.css'. The main area is the code editor, showing the following code for index.html:

```

1  <!DOCTYPE html>
2
3  <html>
4      <head>
5          <meta charset="utf-8">
6          <title></title>
7      </head>
8
9      <body>
10
11     </body>
12 </html>

```

The terminal window shows the following output:

```

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ git status
On branch Prueba_Branch_Royal6969/seginfoFAQ
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   README.md

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    index.html
    style.css

```

The terminal window shows the following output:

```

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ git commit -m "added index and style files"
[Prueba_Branch_Royal6969/seginfoFAQ ecc5a58] added index and style files
 1 file changed, 1 insertion(+), 1 deletion(-)

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ

```

The screenshot shows the Visual Studio Code interface again. The code editor now displays the following code for index.html, with the title element modified:

```

1  <!DOCTYPE html>
2
3  <html>
4      <head>
5          <meta charset="utf-8">
6          <title>My second commit</title>
7      </head>
8
9      <body>
10
11     </body>
12 </html>

```

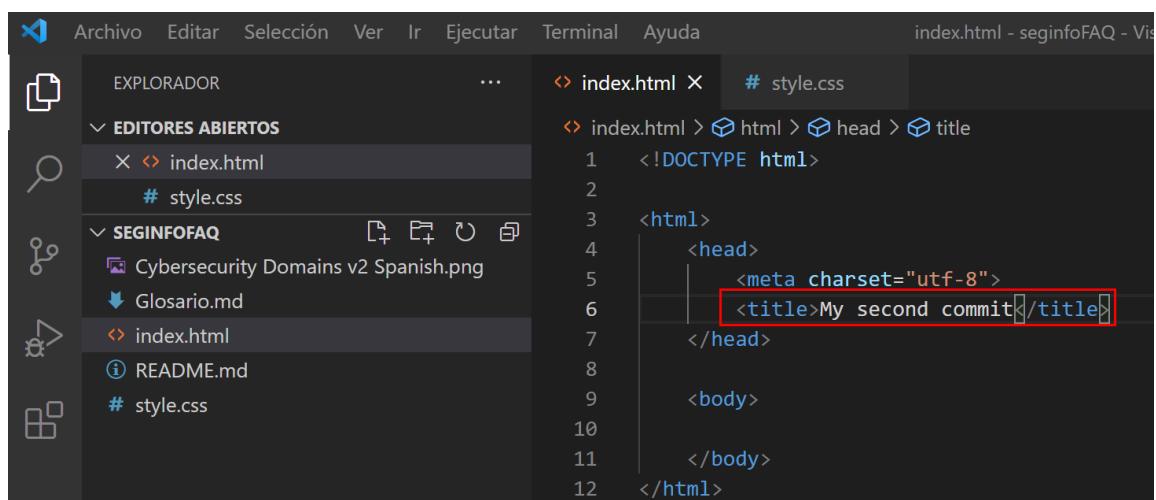
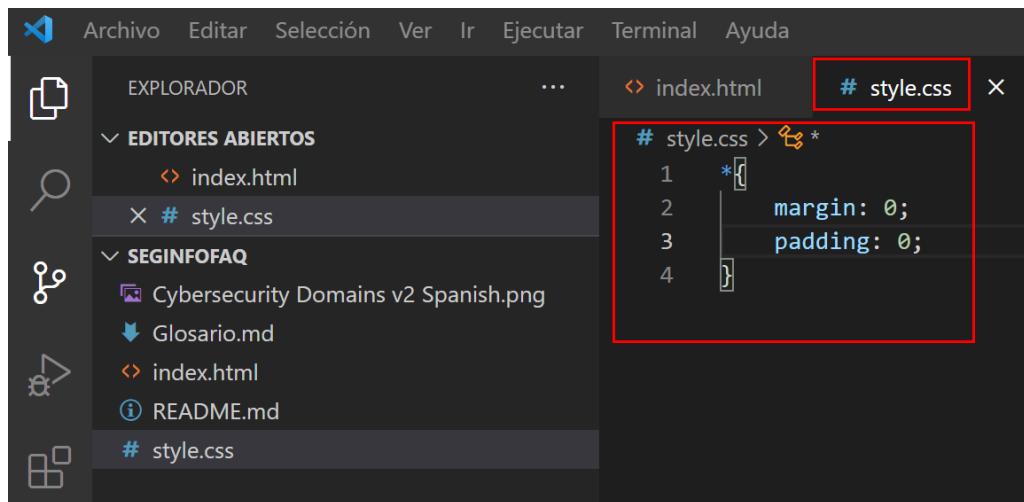
The terminal window shows the following output:

```

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ git add .
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ git status
On branch Prueba_Branch_Royal6969/seginfoFAQ
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   index.html
    new file:   style.css

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ

```



```
Cmder
On branch Prueba_Branch_Royal6969/seginfoFAQ
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:  index.html
    new file:  style.css

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ git status
On branch Prueba_Branch_Royal6969/seginfoFAQ
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:  index.html
    new file:  style.css

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   style.css

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ git add .

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ git status
On branch Prueba_Branch_Royal6969/seginfoFAQ
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:  index.html
    new file:  style.css

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ 
```

index.html - seginfoFAQ - Visual Studio Code

```

<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title>My second commit</title>
</head>
<body>
</body>
</html>

```

index.html # style.css

SEGINFOFAQ

- Cybersecurity Domains v2 Spanish.png
- Glosario.md
- index.html
- README.md
- # style.css

index.html # style.css

CMder

```

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ git commit -m "added index tittle and basic doc styles"
[Prueba_Branch_Royal6969/seginfoFAQ 8df6829] added index tittle and basic doc styles
2 files changed, 16 insertions(+)
create mode 100644 index.html
create mode 100644 style.css

```

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ

style.css - seginfoFAQ - Visual Studio Code

```

* {
    margin: 0;
    padding: 0;
    box-sizing: border-box;
}

```

index.html # style.css

EXPLORADOR

EDITORES ABIERTOS

- index.html
- # style.css

SEGINFOFAQ

- Cybersecurity Domains v2 Spanish.png
- Glosario.md
- index.html
- README.md
- # style.css

style.css - seginfoFAQ - Visual Studio Code

```

* {
    margin: 0;
    padding: 0;
    box-sizing: border-box;
}

```

index.html # style.css

EXPLORADOR

EDITORES ABIERTOS

- # style.css

SEGINFOFAQ

- Cybersecurity Domains v2 Spanish.png
- Glosario.md
- index.html
- README.md
- # style.css

CMder

```

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ git commit -m "added index tittle and basic doc styles"
[Prueba_Branch_Royal6969/seginfoFAQ 8df6829] added index tittle and basic doc styles
2 files changed, 16 insertions(+)
create mode 100644 index.html
create mode 100644 style.css

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ git commit -m "added box sizing style"
error: pathspec '-' did not match any file(s) known to git
error: pathspec 'm' did not match any file(s) known to git
error: pathspec 'added box sizing style' did not match any file(s) known to git

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ git commit -m "added box sizing style"
On branch Prueba_Branch_Royal6969/seginfoFAQ
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   style.css

no changes added to commit (use "git add" and/or "git commit -a")

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ 

```

```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ git commit -m "added box sizing style"
On branch Prueba_Branch_Royal6969/seginfoFAQ
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
    (use "git restore <file>..." to discard changes in working directory)
      modified:  style.css

no changes added to commit (use "git add" and/or "git commit -a")

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ git add .
[red box around git add .]

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ git status
[red box around git status]
On branch Prueba_Branch_Royal6969/seginfoFAQ
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:  style.css

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ [red box around ]
[red box around cmd.exe] Search [red box around ] [red box around ]
```

```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ git add .
[red box around git add .]

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ git status
[red box around git status]
On branch Prueba_Branch_Royal6969/seginfoFAQ
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:  style.css [red box around modified: style.css]

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ git commit -m "added box sizing style"
[Prueba_Branch_Royal6969/seginfoFAQ 0883746] added box sizing style
[red box around 1 file changed, 1 insertion(+)] [red box around 1 file changed, 1 insertion(+)]

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ [red box around ]
[red box around cmd.exe] Search [red box around ] [red box around ]
```

```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ git log
[red box around git log]
commit 088374692bc1afc71fbe51b92c3b0b70149396f1 (HEAD -> Prueba_Branch_Royal6969/seginfoFAQ)
Author: Royal6969 <sergiodiazcampos@gmail.com>
Date:   Sat Apr 10 16:37:21 2021 +0200

  added box sizing style

commit 8df68292f067171be5b12a4b530393ad8a194a11
Author: Royal6969 <sergiodiazcampos@gmail.com>
Date:   Sat Apr 10 16:32:07 2021 +0200

  added index tittle and basic doc styles

commit ecc5a581409b5e6138b1c1c63d73032ef16b2edd
Author: Royal6969 <sergiodiazcampos@gmail.com>
Date:   Sat Apr 10 16:24:30 2021 +0200

  added index and style files

commit bec2d110948dbe528e114ac5cf67405a3da60cc5 (master)
Author: Royal6969 <sergiodiazcampos@gmail.com>
Date:   Sat Apr 10 11:45:54 2021 +0200

  Add existing file
```

```
C:\Users\Sergio_DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ git log --oneline
0883746 (HEAD -> Prueba_Branch_Royal6969/seginfoFAQ) added box sizing style
8df6829 added index tittle and basic doc styles
ecc5e58 added index and style files
bec2d11 (master) Add existing file
```

Con todo esto llego a la conclusión de que, cuando en VS modifique-añada-elimine cualquier archivo, si en la consola escribo “git status” me muestra el estado de la carpeta de trabajo según los cambios que haya habido.

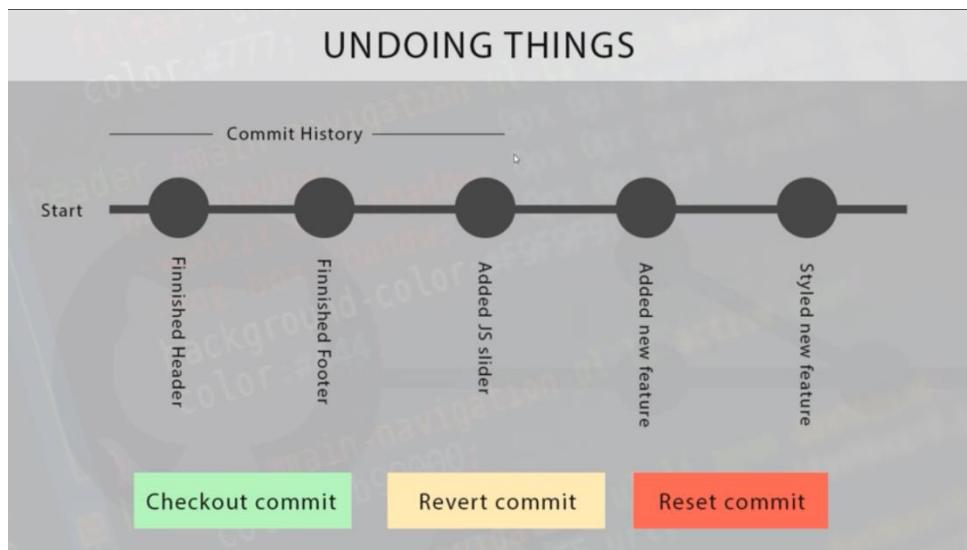
Si aparece un archivo en color rojo, es que se ha modificado-añadido-eliminado y no se han guardado los cambios.

Si aparece un archivo en color verde, es que se ha modificado-añadido-eliminado y sí se han guardado los cambios.

Para guardar los cambios hay que ejecutar el comando “git add .” y después de éste se debe escribir el comando “git commit -m [descripción cambios]” para crear el commit de los cambios producidos.

Con el comando “git log” podemos ver un listado de los cambios que se han producido sobre la carpeta de trabajo y quienes han hecho esos cambios.

Finalmente, con el comando “git log –oneline” se registran verdaderamente estos cambios hacia GitHub.



Voy a crear dos nuevos commits para probar los tres comandos de:

- Checkout commit
- Revert commit
- Reset commit

Para ello, vamos a añadir un `<h1>` en el body y un estilo de body; y por otro lado creamos un archivo .js

The figure consists of three vertically stacked screenshots of a Windows desktop environment. The top two screenshots show the Visual Studio Code interface. The first screenshot shows the 'index.html' file open, with the line '`<h1>This is a heading</h1>`' highlighted with a red box. The second screenshot shows the 'style.css' file open, with the CSS rule for the body element (`body { background: red; font-size: 2em; }`) highlighted with a red box. The bottom screenshot shows a terminal window titled 'Cmder' with the following command-line session:

```

λ git status
On branch Prueba_Branch_Royal6969/seginfoFAQ
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   index.html
    modified:   style.css

no changes added to commit (use "git add" and/or "git commit -a")

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ git add .

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ git status
On branch Prueba_Branch_Royal6969/seginfoFAQ
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   index.html
    modified:   style.css

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ git commit -m "added index heading and style body"
[Prueba_Branch_Royal6969/seginfoFAQ 98f9eda] added index heading and style body
  2 files changed, 7 insertions(+), 2 deletions(-)

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ

```

The screenshot shows the Visual Studio Code interface. In the center-right is the code editor window with the file 'feature.js' open. The code inside is:

```
1 console.log('Hey there');
```

To the left of the editor is the Explorer sidebar, which lists several files and folders under the 'SEGINFOFAQ' folder. One file, 'feature.js', is highlighted with a red box.

The screenshot shows a terminal window titled 'Cmder'. It displays the following command-line session:

```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ git status
On branch Prueba_Branch_Royal6969/seginfoFAQ
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    feature.js

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

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ git add .

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ git status
On branch Prueba_Branch_Royal6969/seginfoFAQ
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   feature.js

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ git commit -m "added feature.js"
[Prueba_Branch_Royal6969/seginfoFAQ 7c9c08a] added feature.js
  1 file changed, 1 insertion(+)
   create mode 100644 feature.js

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ
```

The screenshot shows a terminal window titled 'Cmder'. It displays the following command-line session:

```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969/seginfoFAQ)
λ git log --oneline
7c9c08a (HEAD -> Prueba_Branch_Royal6969/seginfoFAQ) added feature.js
98f9eda added index heading and style body
0883746 added box sizing style
8df6829 added index tittle and basic doc styles
ecc5a58 added index and style files
bec2d11 (master) Add existing file
```

The first line of the log output ('7c9c08a (HEAD -> Prueba_Branch_Royal6969/seginfoFAQ) added feature.js') is highlighted with a red box.

The screenshot shows a Windows desktop environment. In the center is a Visual Studio Code window displaying an HTML file named 'index.html'. The code includes a title 'My second commit' and a body section. To the left of the code editor is a file explorer window showing a directory structure under 'SEGINFOFAQ'. A file named 'index.html' is selected and highlighted with a red box. Below the code editor is a terminal window titled 'Cmdr' showing a git command:

```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (Prueba_Branch_Royal6969\seginfoFAQ)
λ git checkout 8df6829
Note: switching to '8df6829'.
```

The terminal continues with:

```
You are in 'detached HEAD' state. You can look around, make experimental changes and commit them, and you can discard any commits you make in this state without impacting any branches by switching back to a branch.
```

If you want to create a new branch to retain commits you create, you may do so (now or later) by using -c with the switch command. Example:

```
git switch -c <new-branch-name>
```

Or undo this operation with:

```
git switch -
```

Turn off this advice by setting config variable advice.detachedHead to false

```
HEAD is now at 8df6829 added index tittle and basic doc styles
```

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (HEAD detached at 8df6829)

Esta acción es de tipo “read only” 1

The screenshot shows a Windows desktop environment. In the center is a Visual Studio Code window displaying an HTML file named 'index.html'. Both the file name and its preview in the sidebar are highlighted with red boxes. To the left of the code editor is a file explorer window showing a directory structure under 'SEGINFOFAQ'. A file named 'index.html' is selected and highlighted with a red box. Below the code editor is a terminal window titled 'Cmdr' showing a git command:

```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (HEAD detached at 8df6829)
λ git checkout master
Previous HEAD position was 8df6829 added index tittle and basic doc styles
Switched to branch 'master'
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

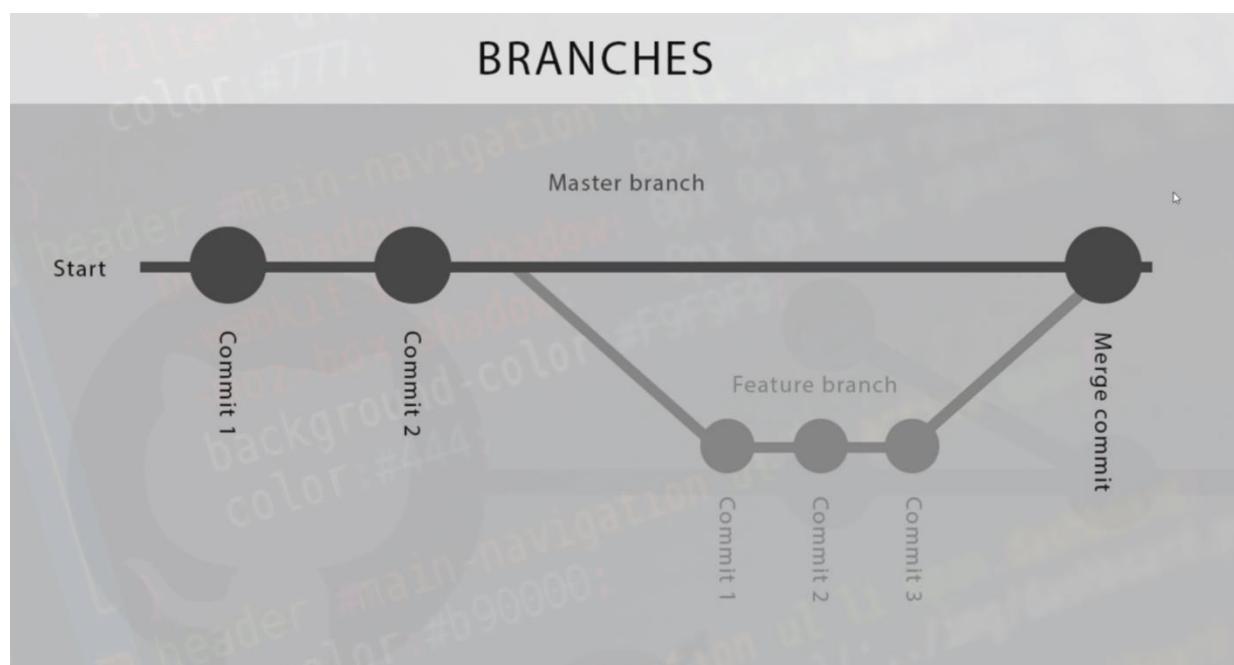
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
```

Volvemos completamente al principio

Llegados a este punto, si queremos volver hacia adelante, bastaría con ejecutar de nuevo el comando “git log --oneline” y buscamos el commit al que queremos volver a estar, copiamos su identificador que sale a su izquierda, y ejecutamos el comando “git revert [identificador]” y con esto volveríamos a avanzar hacia adelante.

Si por el contrario queremos ir a un commit, y “volver a empezar” el proyecto estableciendo como punto inicial de partida tal commit, para ello usaremos el comando con “reset”, es decir, Imaginemos que estamos en el commit inicial, el del “branch master” y queremos volver hacia adelante, por ejemplo al commit de “added index and style files” (identificador en la imagen de ecc5a58) pero estableciendo tal commit como el primero de todos los commits, como el punto original de partida... para ello ejecutaríamos el comando “git reset [identificador]” y con ello apreciaríamos que vamos hacia tal commit y que automáticamente se han eliminado todos los anteriores commits que había antes de él, como si este fuese el nuevo punto de partida.

A continuación, vamos con lo último de la práctica, crear un branch.



Para ello, volvemos a crear y a guardar un index.html

```

index.html - seginfoFAQ - Visual Studio...
EXPLORADOR
EDITORES ABIERTOS
  index.html
  SEGINFOFAQ
    Cybersecurity Domains v2 Spanish.png
    Glosario.md
    index.html
  README.md

index.html > html
1  <!DOCTYPE html>
2  <html>
3  <head>
4  |   <title>Ejer_1</title>
5  |   <meta charset="UTF-8">
6  </head>
7  <body>

Cmder
On branch master
Your branch is ahead of 'origin/master' by 2 commits.
(use "git push" to publish your local commits)

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

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

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ git add .

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ git status
On branch master
Your branch is ahead of 'origin/master' by 2 commits.
(use "git push" to publish your local commits)

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   index.html

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ git commit -m "added index.html"
[master 575906d] added index.html
 1 file changed, 10 insertions(+)
 create mode 100644 index.html
λ
cmd.exe

```

Ahora, hacemos otro commit de cambiar el título del index.html

```

index.html - seginfoFAQ - Visual Studio...
EXPLORADOR
EDITORES ABIERTOS
  index.html M
  SEGINFOFAQ
    Cybersecurity Domains v2 Spanish.png
    Glosario.md
    index.html M
  README.md

index.html > html > head > title
1  <!DOCTYPE html>
2  <html>
3  <head>
4  |   <title>Wooooow</title>
5  |   <meta charset="UTF-8">
6  </head>
7  <body>

Cmder
On branch master
Your branch is ahead of 'origin/master' by 3 commits.
(use "git push" to publish your local commits)

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   index.html

no changes added to commit (use "git add" and/or "git commit -a")

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ git add .

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ git commit -m "change index title"
[master de3e27d] change index title
 1 file changed, 1 insertion(+), 1 deletion(-)

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ

```

```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ git log --oneline
de3e27d (HEAD -> master) change index title
575906d added index.html
45d7108 Revert "Add existing file"
bec2d11 Add existing file
```

```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ git branch feature-1
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ git branch -a
Prueba_Branch_Royal6969/seginfoFAQ ←
feature-1 ←
* master ←
  remotes/origin/HEAD -> origin/master
  remotes/origin/master

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ
```

Con “git branch [nombre branch]” creamos un branch poniéndole un nombre.

Con “git branch -a” consultamos un listado de los branch que tiene el proyecto.

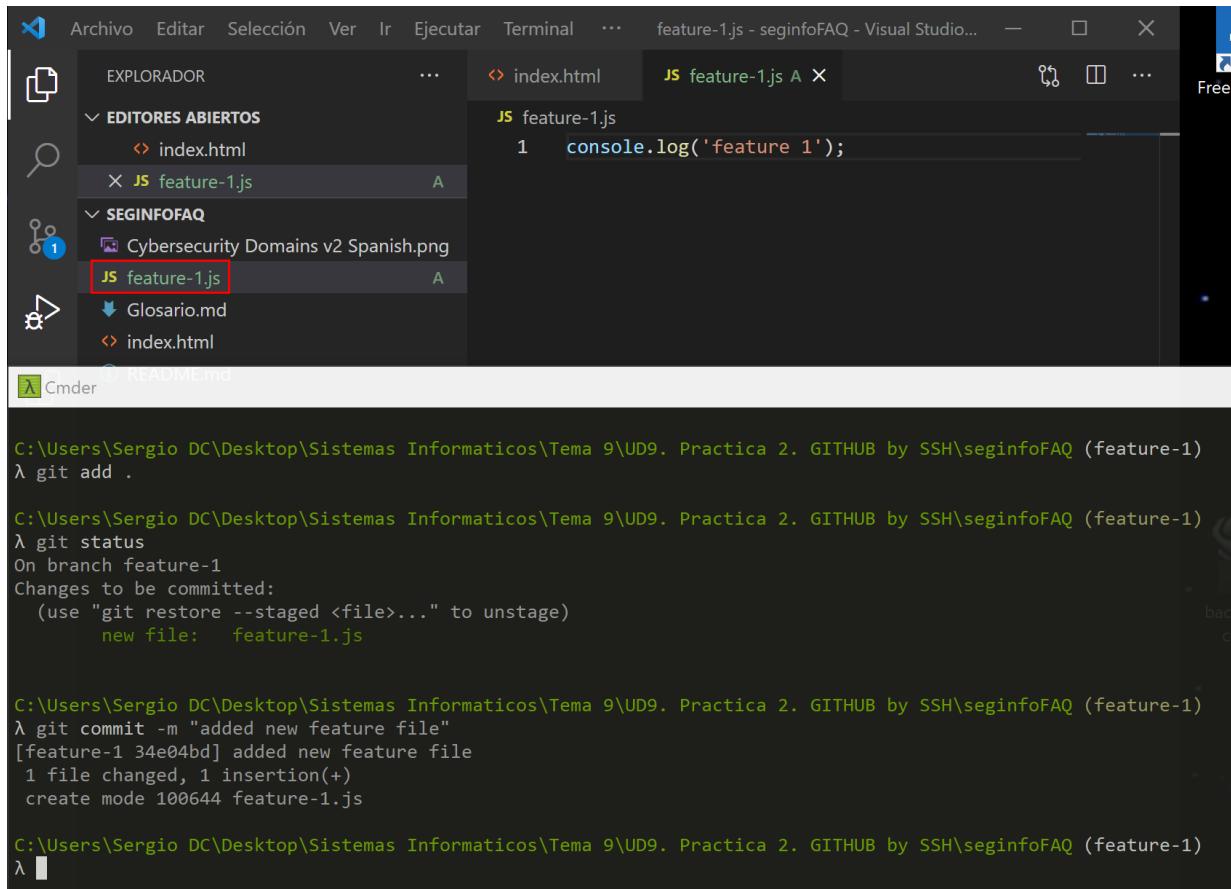
Observamos que tenemos el branch de “Prueba_Branch_Royal6969/seginfoFAQ” el cual creamos a modo de prueba cuando no sabíamos bien esto antes de empezar a seguir a The Net Ninja,
y también tenemos el nuevo branch que acabamos de crear el “feature-1”,
y por otro lado, sale marcado con un (*) el branch en el que nos encontramos actualmente (master).

Para cambiarnos al nuevo branch que acabamos de crear, ejecutamos el comando “git checkout [nombre branch]”

```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ git checkout feature-1
Switched to branch 'feature-1' ←
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (feature-1)
λ git branch -a
Prueba_Branch_Royal6969/seginfoFAQ
* feature-1 ←
  master
  remotes/origin/HEAD -> origin/master
  remotes/origin/master

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (feature-1)
λ
```

Creamos un archivo feature.js y guardamos el commit.



The screenshot shows the Visual Studio Code interface. In the Explorer sidebar, under the 'EDITORES ABIERTOS' section, there is a file named 'JS feature-1.js'. This file is highlighted with a red box. In the main editor area, the code `console.log('feature 1');` is visible. The terminal at the bottom shows the command-line history:

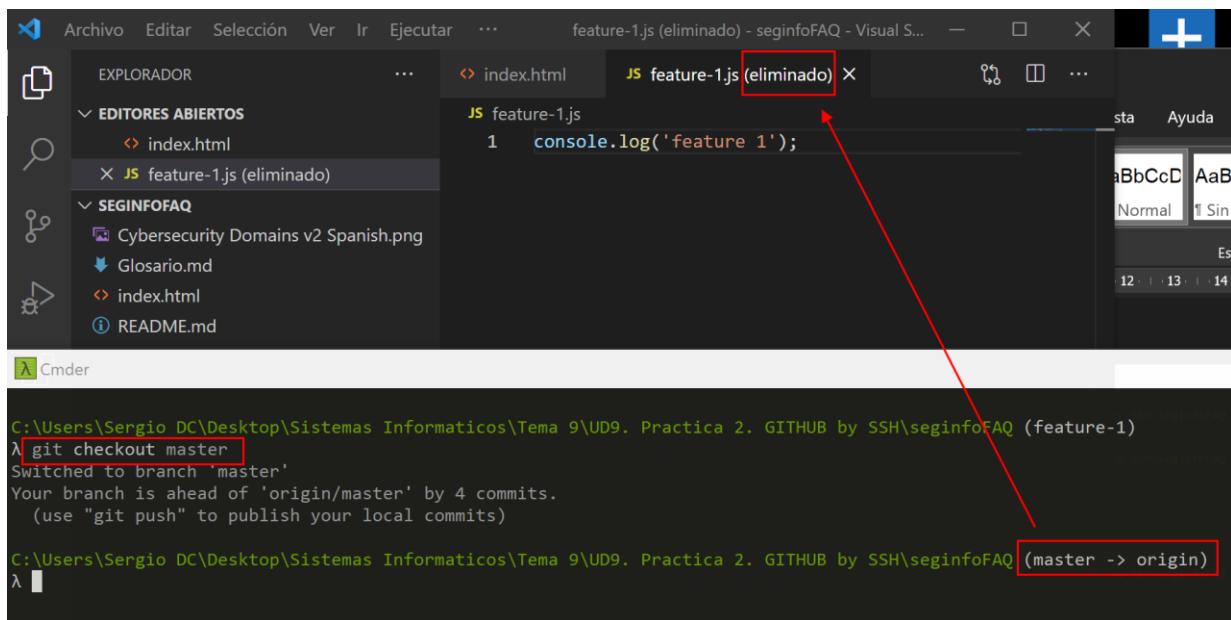
```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (feature-1)
λ git add .

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (feature-1)
λ git status
On branch feature-1
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   feature-1.js

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (feature-1)
λ git commit -m "added new feature file"
[feature-1 34e04bd] added new feature file
 1 file changed, 1 insertion(+)
 create mode 100644 feature-1.js

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (feature-1)
λ 
```

Para ver que esto ha salido bien, nos cambiamos al branch del “master” y... efectivamente vemos que no existe el archivo feature.js que habíamos creado estando en el branch del “feature” ¡!



The screenshot shows the Visual Studio Code interface again. In the Explorer sidebar, under the 'EDITORES ABIERTOS' section, there is a file named 'JS feature-1.js (eliminado)'. This file is highlighted with a red box. In the main editor area, the code `console.log('feature 1');` is visible. The terminal at the bottom shows the command-line history:

```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (feature-1)
λ git checkout master
Switched to branch 'master'
Your branch is ahead of 'origin/master' by 4 commits.
  (use "git push" to publish your local commits)

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ 
```

Y si quiero eliminar el branch de “feature-1”... ¿qué ocurriría?...

```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginFAQ (master -> origin)
λ git branch -d feature-1
error: The branch 'feature-1' is not fully merged. ←
If you are sure you want to delete it, run 'git branch -D feature-1'.
```

The terminal window shows a command being entered: "git branch -d feature-1". The error message "error: The branch 'feature-1' is not fully merged." is displayed, followed by a note that says "If you are sure you want to delete it, run 'git branch -D feature-1'". A red box highlights the command "git branch -d feature-1". A red arrow points from the error message back to the command line.

Sopresa! ... me dice que no se puede eliminar porque aún no se han fusionado los branch y no han confluido en un mismo punto.

```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginFAQ (master -> origin)
λ git branch -d feature-1
error: The branch 'feature-1' is not fully merged. ←
If you are sure you want to delete it, run 'git branch -D feature-1'.
```

```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginFAQ (master -> origin)
λ git branch -D feature-1
Deleted branch feature-1 (was 34e04bd). ←
```

```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginFAQ (master -> origin)
λ git branch -a
  Prueba_Branch_Royal6969/seginFAQ
* master
  remotes/origin/HEAD -> origin/master
  remotes/origin/master
```

The terminal window shows a command being entered: "git branch -D feature-1". The error message "error: The branch 'feature-1' is not fully merged." is displayed, followed by a note that says "If you are sure you want to delete it, run 'git branch -D feature-1'". A red box highlights the command "git branch -D feature-1". A red arrow points from the error message back to the command line. The command is then run again, and the output shows "Deleted branch feature-1 (was 34e04bd)". A red arrow points from this output back to the command line. Finally, the command "git branch -a" is run to show the current branches, which includes the deleted branch "feature-1". A red box highlights "git branch -a".

Aún así, la consola nos sugiere un comando para eliminar el branch por la fuerza... y efectivamente comprobamos que se elimina (posteriormente no sale en la lista de branch).

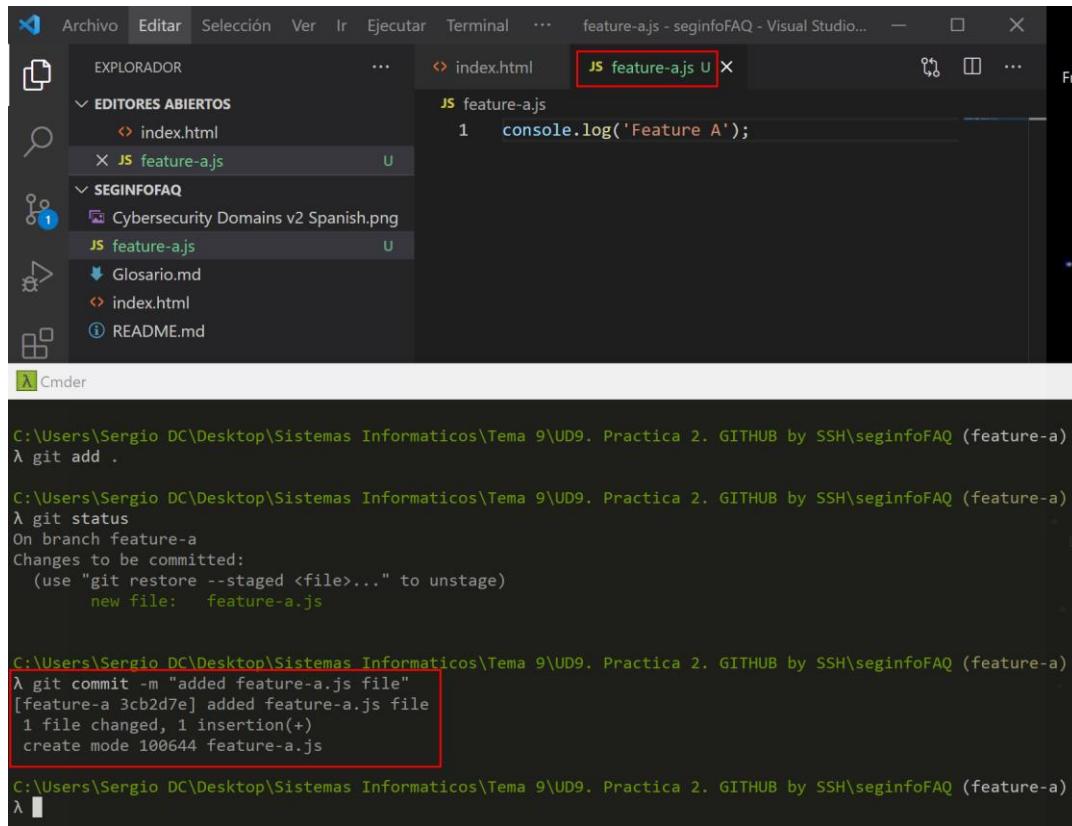
También podemos crear un branch y cambiarnos a él en un solo comando:

```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginFAQ (master -> origin)
λ git checkout -b feature-a
Switched to a new branch 'feature-a'
```

```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginFAQ (feature-a)
λ
```

The terminal window shows a command being entered: "git checkout -b feature-a". The output shows "Switched to a new branch 'feature-a'". A red box highlights "git checkout -b feature-a".

Ahora vamos a añadir un commit a este nuevo branch.



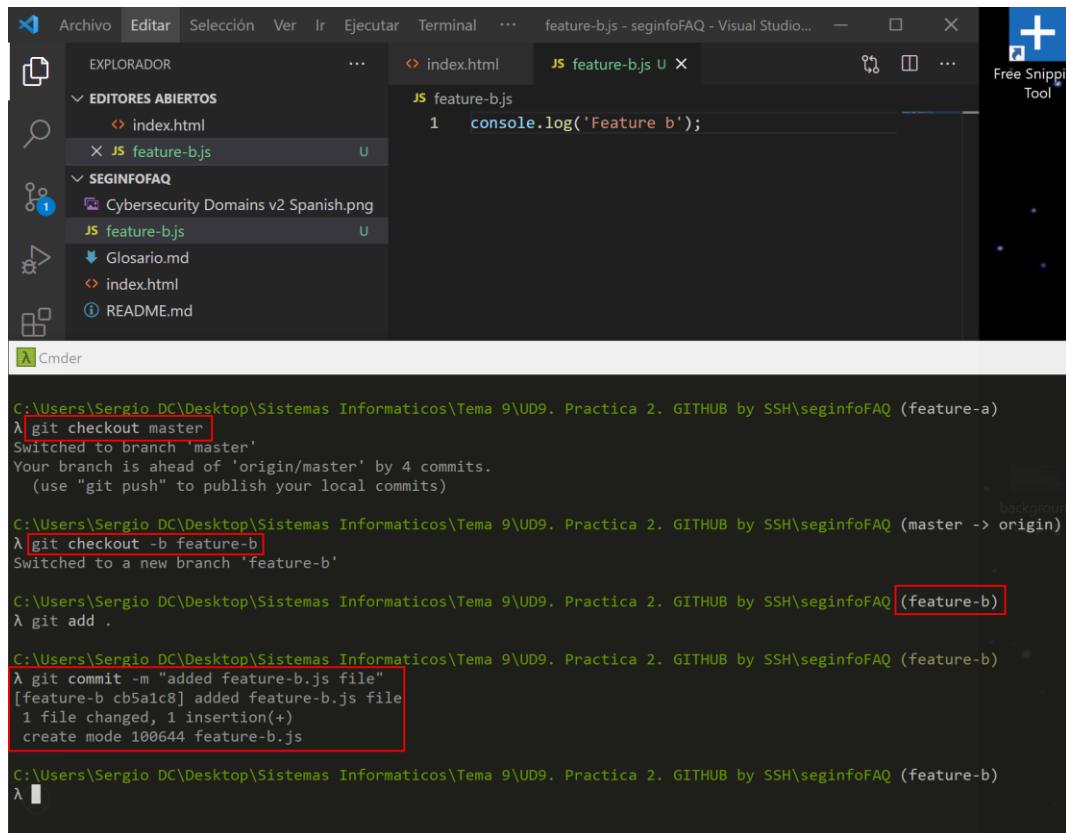
```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (feature-a)
λ git add .

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (feature-a)
λ git status
On branch feature-a
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:  feature-a.js

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (feature-a)
λ git commit -m "added feature-a.js file"
[feature-a 3cb2d7e] added feature-a.js file
 1 file changed, 1 insertion(+)
 create mode 100644 feature-a.js

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (feature-a)
λ 
```

Ahora, volvemos al branch del “master” y desde él, creamos otro branch, y otro .js, y guardamos este otro commit.



```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (feature-a)
λ git checkout master
Switched to branch 'master'
Your branch is ahead of 'origin/master' by 4 commits.
  (use "git push" to publish your local commits)

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ git checkout -b feature-b
Switched to a new branch 'feature-b'

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (feature-b)
λ git add .

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (feature-b)
λ git commit -m "added feature-b.js file"
[feature-b cb5a1c8] added feature-b.js file
 1 file changed, 1 insertion(+)
 create mode 100644 feature-b.js

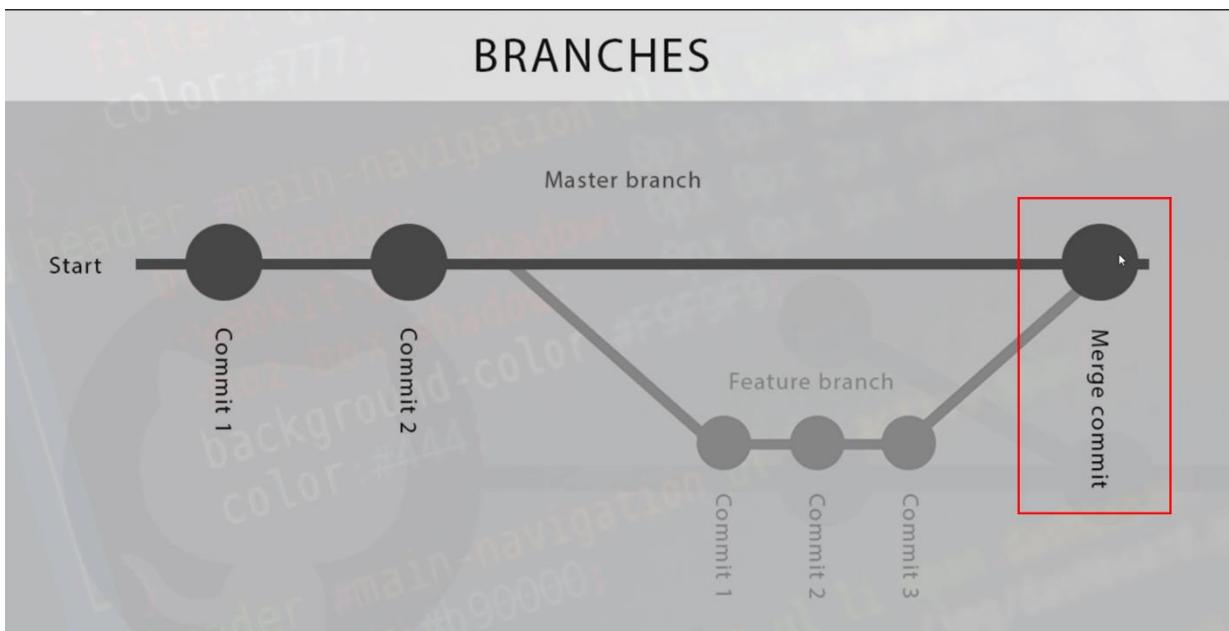
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (feature-b)
λ 
```

Recapitulando...

```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ git branch -a
  Prueba_Branch_Royal6969/seginfoFAQ
    feature-a
    feature-b
* master
  remotes/origin/HEAD -> origin/master
  remotes/origin/master

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ 
```

Ahora, vamos a fusionar los branch, es decir, hacer que confluyan en el mismo punto. Para este ejemplo, supongamos que nos ha gustado más el desarrollo que hemos hecho sobre el branch “feature-a” y queremos que se acabe uniendo éste con el “master”.



```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ git merge feature-a
Updating de3e27d..3cb2d7e
Fast-forward ←
  feature-a.js | 1 +
  1 file changed, 1 insertion(+)
  create mode 100644 feature-a.js

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ 
```

Con el comando “git merge [feature-a]” hemos unido el branch “feature-a” al branch del “master”, y el dato de “fast-forward” significa que no se han guardado los cambios en el “master” si no que se han implantado con prioridad los cambios y el camino del “feature-a”.

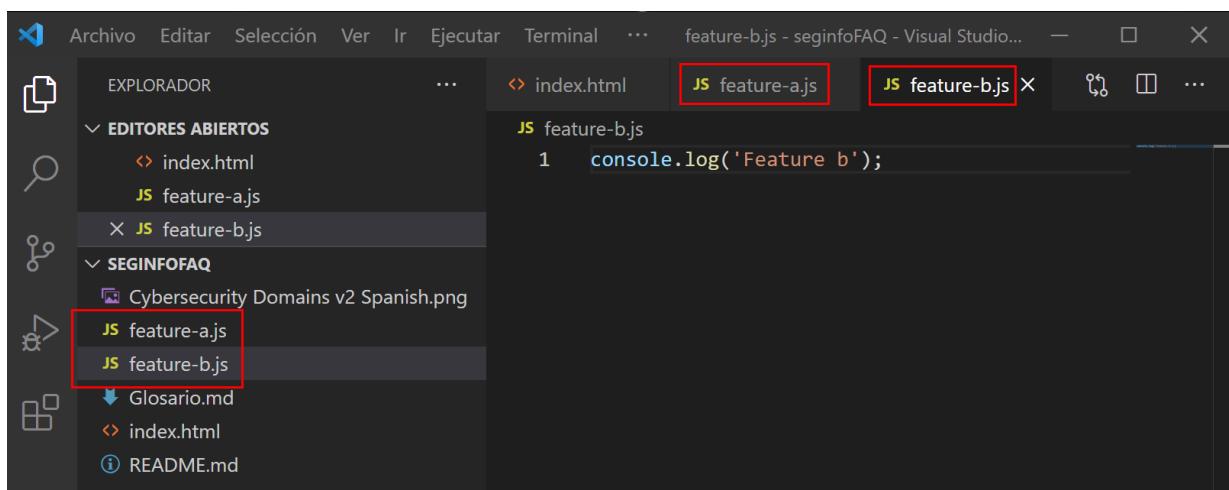
A screenshot of a terminal window titled "Cmder". The command entered is "git merge feature-b". The output shows:

```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ git merge feature-b
Merge made by the 'recursive' strategy.
feature-b.js | 1 +
1 file changed, 1 insertion(+)
create mode 100644 feature-b.js
```

The output is highlighted with red boxes around the command, the strategy message, and the file details. A red arrow points to the strategy message.

En este caso, al unirle también al “master” el “feature-b”, como y atenía de antes del “feature-a”, no ha usado el Fast-forward.

Y comprobamos en el VS que efectivamente todos los branch se han unido al branch principal del “master”.



Una vez claro todo esto, vamos a suponer un último ejemplo que podría pasarnos...

Creamos un nuevo branch... y hacemos una modificación en el “master” en vez de en el nuevo branch... añadiendo un style.css, con par de estilos en el <body>

```

# style.css - seginfoFAQ - Visual Studio ...
EXPLORADOR
EDTORES ABIERTOS
  index.html
  # style.css
SEGINFOFAQ
  Cybersecurity Domains v2 Spanish.png
  feature-a.js
  feature-b.js
  Glosario.md
  index.html
  README.md
  # style.css

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ git checkout -b feature-c
Switched to a new branch 'feature-c'

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (feature-c)
λ git checkout master
Switched to branch 'master'
Your branch is ahead of 'origin/master' by 7 commits.
(use "git push" to publish your local commits)

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ git add .

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ git commit -m "added body style.css"
[master 0c4928f] added body style.css
 1 file changed, 4 insertions(+)
 create mode 100644 style.css

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ

```

Ahora nos cambiamos el “feature-c” y creamos un style.css con otro par de estilos diferentes para el <body>.

```

# style.css - seginfoFAQ - Visual Studio ...
EXPLORADOR
EDTORES ABIERTOS
  index.html
  # style.css
SEGINFOFAQ
  Cybersecurity Domains v2 Spanish.png
  feature-a.js
  feature-b.js
  Glosario.md
  index.html
  README.md
  # style.css

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ git checkout feature-c
Switched to branch 'feature-c'

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (feature-c)
λ git add .

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (feature-c)
λ git commit -m "added body style.css file"
[feature-c 5c36ad5] added body style.css file
 1 file changed, 4 insertions(+)
 create mode 100644 style.css

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (feature-c)
λ

```

Ahora tenemos dos “style.css” en el “master” y otro en el “feature-c” ... vamos a probar a fusionar el “feature-c” con el “master” como ya hicimos anteriormente, a ver qué pasa...

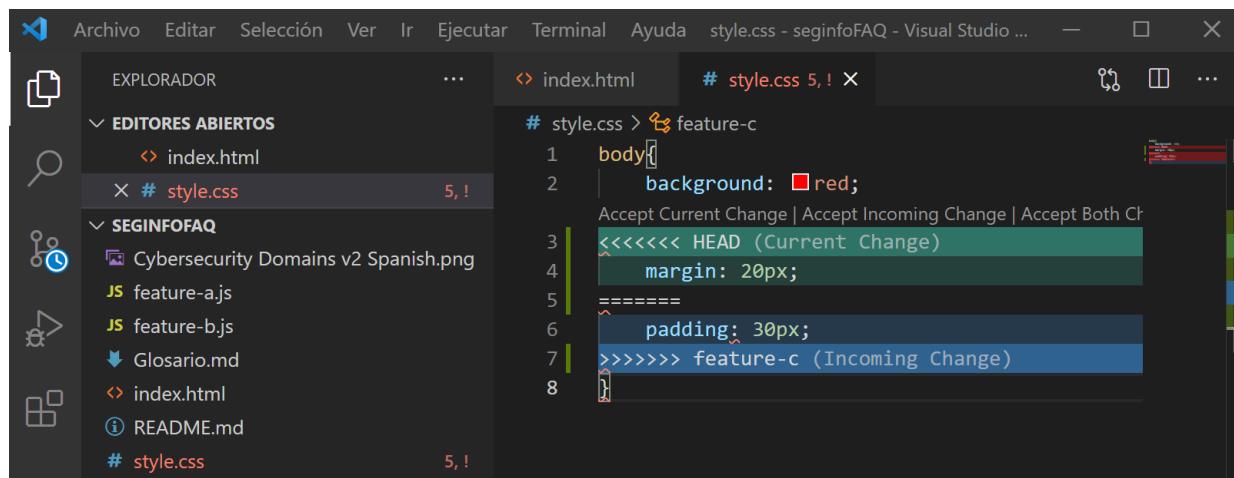
```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (feature-c)
λ git checkout master
Switched to branch 'master'
Your branch is ahead of 'origin/master' by 8 commits.
  (use "git push" to publish your local commits)

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ git merge feature-c
CONFLICT (add/add): Merge conflict in style.css ←
Auto-merging style.css
Automatic merge failed; fix conflicts and then commit the result. ←

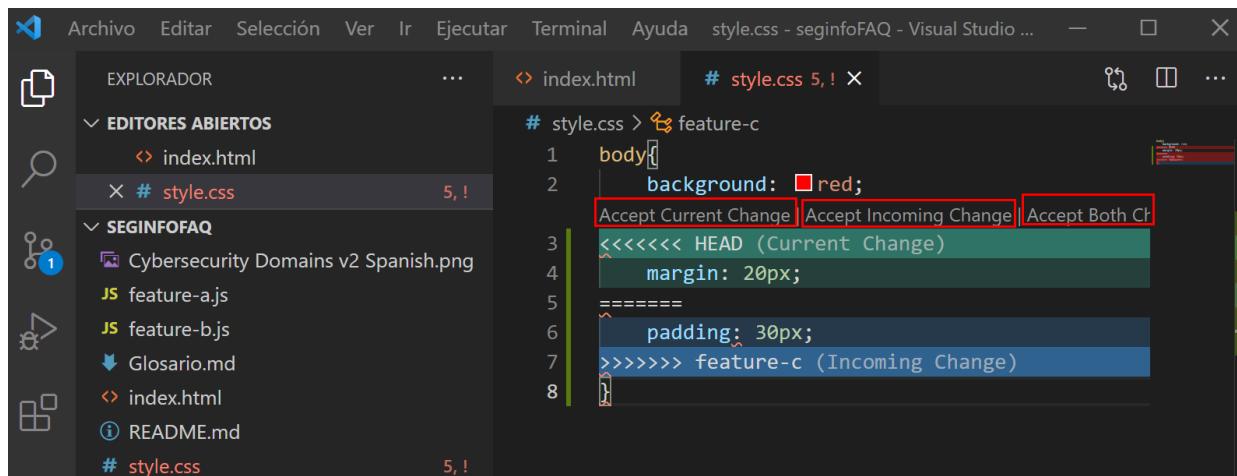
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ
```

Wow! Conflicto entre los style.css ¡!

Si nos vamos al VS observamos lo siguiente...



Observamos que VS nos da tres opciones... aceptar el cambio del “master”, aceptar el cambio del “feature-c”, o combinar ambos “style.css”



Aceptamos ambos cambios para que se combinen los css, quedando algo tal que así:

```

index.html
# style.css ! X

# style.css > body
1 body[background: red; margin: 20px; padding: 30px;]
2
3
4
5

```

```

COMMIT_EDITMSG - seginfoFAQ - Visual Studio Code

EXPLORADOR ... index.html # style.css M COMMIT_EDITMSG X

.git > COMMIT_EDITMSG
1 Merge branch 'feature-c'
2
3 # Conflicts:
4 # style.css
5 #
6 # It looks like you may be committing a merge.
7 # If this is not correct, please run
8 # git update-ref -d MERGE_HEAD
9 # and try again.
10
11
12 # Please enter the commit message for your changes. Lines starting
13 # with '#' will be ignored, and an empty message aborts the commit.
14 #
15 # On branch master
16 # Your branch is ahead of 'origin/master' by 8 commits.
17 # (use "git push" to publish your local commits)
18 #
19 # All conflicts fixed but you are still merging.
20 #
21 # Changes to be committed:
22 # modified: style.css

Cmder

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ git add .

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ git commit
hint: Waiting for your editor to close the file...

```

The screenshot shows a Visual Studio Code interface. On the left, the Explorer sidebar lists files: index.html, style.css, Cybersecurity Domains v2 Spanish.png, feature-a.js, feature-b.js, Glosario.md, index.html, README.md, and another style.css file. The main editor area shows a CSS snippet for a body element with a red background, 20px margin, and 30px padding. Below the editor is a terminal window with the following command history:

```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ git add .

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ git commit
hint: Waiting for your editor to close the file...
[master c0b0e5c] Merge branch 'feature-c'

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ 
```

Y finalmente los commits quedaron así:

The terminal window displays the output of the `git log --oneline` command. It shows the following commit history:

```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ git log --oneline
c0b0e5c (HEAD -> master) Merge branch 'feature-c'.
5c36ad5 (feature-c) added body style.css file
0c4928f added body style.css
3123636 Merge branch 'feature-b'
cb5a1c8 (feature-b) added feature-b.js file
3cb2d7e (feature-a) added feature-a.js file
de3e27d change index title
575906d added index.html
45d7108 Revert "Add existing file"
bec2d11 Add existing file
```

Esta parte la integro después de que hoy en clase (un día después de hacer todo lo anterior), con la continuación de esta práctica por parte del profesor, me doy cuenta de que no llegué hacer el *push* de mis *commits*, es decir, no subí todos los cambios y pruebas que hice al repositorio en GitHub, o sea, todos los cambios quedaron en local (mi laptop) pero no se actualizaron en el repositorio de GitHub en mi perfil.

Para actualizar cualquier cambio y subirlo al repositorio en GitHub, es necesario ejecutar el comando “git push origin”, pero cuando lo he ido a ejecutar, me ha saltado el siguiente error.

```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\segininfoFAQ (master -> origin)
λ git push origin
ERROR: Permission to mattaereal/segininfoFAQ.git denied to Royal6969.
fatal: Could not read from remote repository.

Please make sure you have the correct access rights
and the repository exists.

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\segininfoFAQ (master -> origin)
λ
```

En este punto comencé a investigar y descubrí lo siguiente.

fatal: Could not read from remote repository

Este error es más frecuente de lo que parece. A mucha gente le sale otra información como *fatal: Could not read from remote repository (publickey)* pero esto ya se sabe que es porque la key ssh no está bien, y hay que rehacerla de nuevo, pero mi caso es distinto.

El mismo error adicionalmente nos muestra un mensaje más en el que nos dice que *por favor nos aseguremos de tener el acceso correcto y el repositorio exista.*

Para intentar obtener otra pista de dónde puede estar el fallo, ejecuto el siguiente comando:

“ssh -v git@github.com”

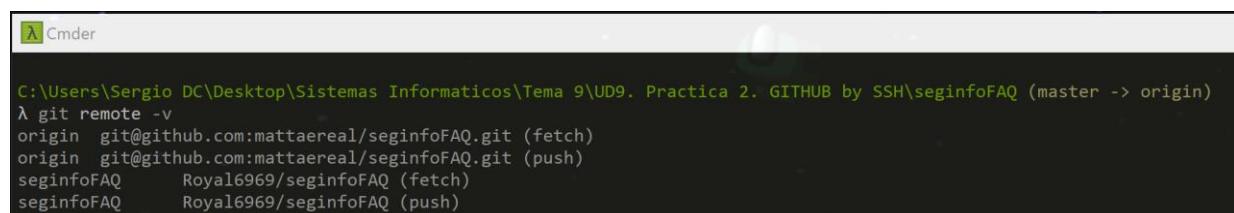
```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\segininfoFAQ (master -> origin)
λ ssh -v git@github.com
OpenSSH_for_Windows_7.7p1, LibreSSL 2.6.5
debug1: Connecting to github.com [140.82.121.3] port 22.
debug1: Connection established.
debug1: identity file C:\\\\Users\\\\Sergio DC\\\\.ssh\\\\id_rsa type 0
debug1: key_load_public: No such file or directory
debug1: identity file C:\\\\Users\\\\Sergio DC\\\\.ssh\\\\id_rsa-cert type -1
debug1: key_load_public: No such file or directory
debug1: identity file C:\\\\Users\\\\Sergio DC\\\\.ssh\\\\id_dsa type -1
debug1: key_load_public: No such file or directory
debug1: identity file C:\\\\Users\\\\Sergio DC\\\\.ssh\\\\id_dsa-cert type -1
debug1: key_load_public: No such file or directory
debug1: identity file C:\\\\Users\\\\Sergio DC\\\\.ssh\\\\id_ecdsa type -1
debug1: key_load_public: No such file or directory
debug1: identity file C:\\\\Users\\\\Sergio DC\\\\.ssh\\\\id_ecdsa-cert type -1
debug1: key_load_public: No such file or directory
debug1: identity file C:\\\\Users\\\\Sergio DC\\\\.ssh\\\\id_ed25519 type -1
debug1: key_load_public: No such file or directory
debug1: identity file C:\\\\Users\\\\Sergio DC\\\\.ssh\\\\id_ed25519-cert type -1
debug1: key_load_public: No such file or directory
debug1: identity file C:\\\\Users\\\\Sergio DC\\\\.ssh\\\\id_xmss type -1
debug1: key_load_public: No such file or directory
debug1: identity file C:\\\\Users\\\\Sergio DC\\\\.ssh\\\\id_xmss-cert type -1
```

Como la lista es muy larga, sólo he capturado en pantalla los errores efectivamente detectados, que son los que devuelven un (-1) y podemos ver que el error tiene que ver con el mensaje que también nos apareció antes cuando intentamos hacer el *push*, en el que dice un mensaje: "No such file or directory".

Tras mucho googlear y buscar por YT, he dado con la siguiente solución, de la cual dejo el link de donde la descubrí:

<https://careerkarma.com/blog/git-please-make-sure-you-have-the-correct-access-rights/>

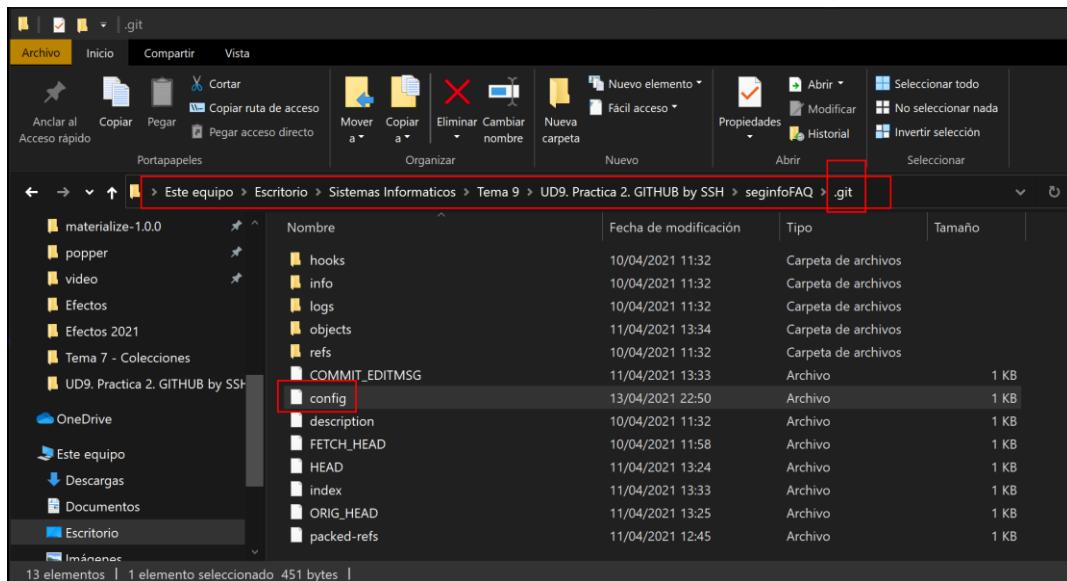
Resulta que cuando ejecutamos el comando "git push origin" le estamos diciendo a nuestro Git, que lleve los cambios de nuestro repositorio local hacia un repositorio "x" en GitHub, es decir, el *origin* guarda un enlace en su interior, y mi fallo era que tal enlace no se correspondía con mi repositorio en GitHub llamado Royal6969/seginfoFAQ, y por ello no se podían actualizar los cambios, porque Git no sabía hacia a dónde apuntar para lanzar los cambios.



```
C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ git remote -v
origin  git@github.com:mattaereal/seginfoFAQ.git (fetch)
origin  git@github.com:mattaereal/seginfoFAQ.git (push)
seginfoFAQ    Royal6969/seginfoFAQ (fetch)
seginfoFAQ    Royal6969/seginfoFAQ (push)
```

Comando para ver hacia a dónde está apuntando actualmente nuestro "origin"

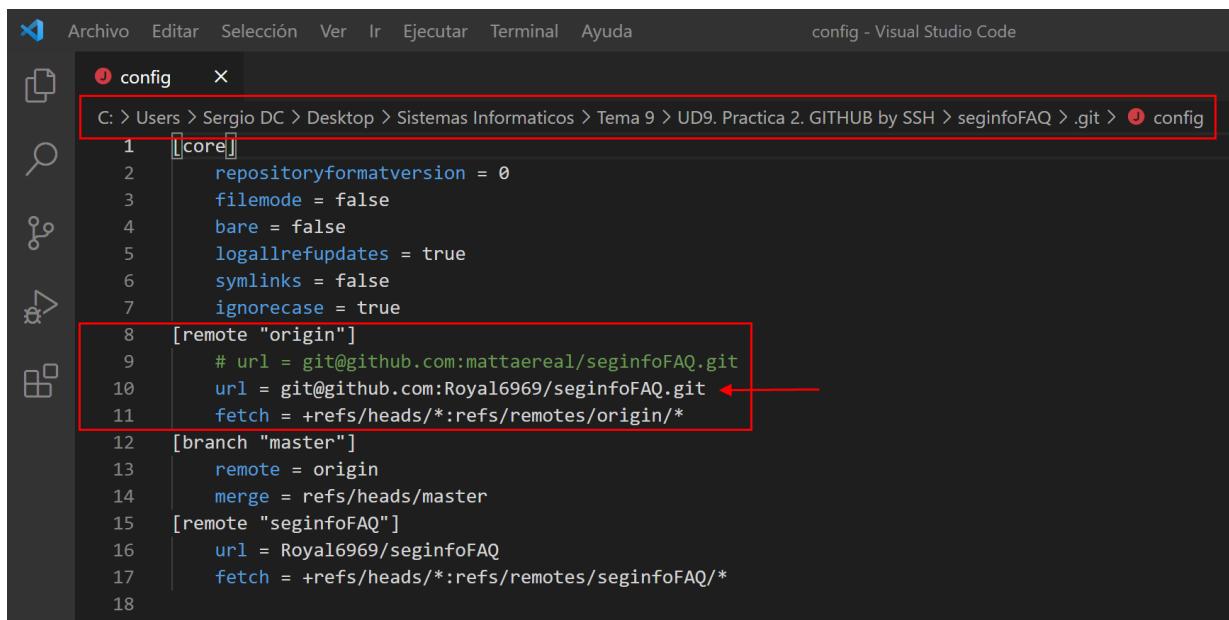
Para cambiar el enlace al que *origin* apunta, tenemos que dirigirnos a nuestra carpeta de nuestro repositorio local, y en nuestro File Explorer, tenemos que irnos a la pestaña de *Vista* y activar el casillero de la derecha que dice "Elementos Ocultos", y de esta forma podremos ver ahora una carpeta oculta llamada *.git*



Una vez dentro de esa carpeta, encontraremos un archivo llamado *config*
 Debemos abrir ese archivo con un editor de texto (VS en mi caso) e irnos al apartado de [remote "origin"]

Vemos como la url del *origin* estaba apuntando hacia donde no debía, es decir, estaba apuntando hacia el repositorio original del autor original, así que claro, ¿cómo iba a poder modificar los archivos de otra persona? Normal que no tenga acceso...

Así que yo voy a cambiar el nombre de perfil del autor original por el mío:

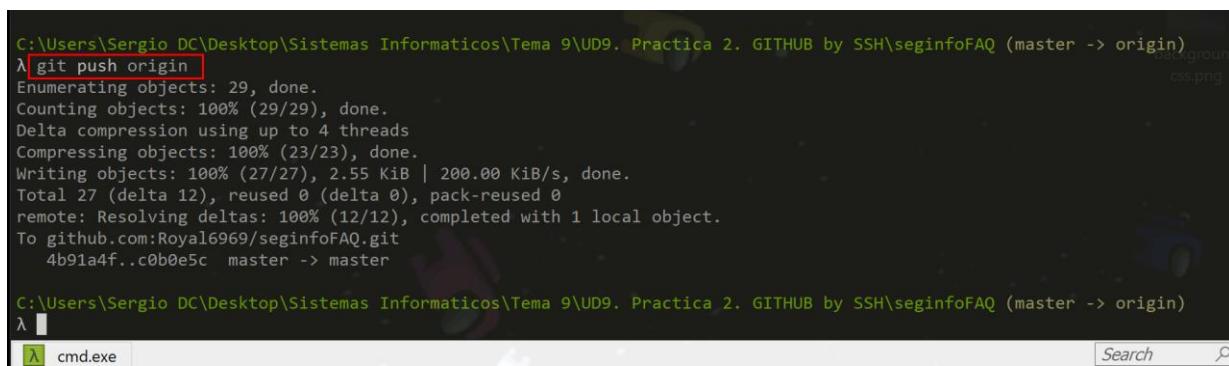


```

Archivo Editar Selección Ver Ir Ejecutar Terminal Ayuda config - Visual Studio Code
config
C: > Users > Sergio DC > Desktop > Sistemas Informaticos > Tema 9 > UD9. Practica 2. GITHUB by SSH > seginfoFAQ > .git > config
1 [core]
2   repositoryformatversion = 0
3   filemode = false
4   bare = false
5   logallrefupdates = true
6   symlinks = false
7   ignorecase = true
8 [remote "origin"]
9   # url = git@github.com:mattaereal/seginfoFAQ.git
10  url = git@github.com:Royal16969/seginfoFAQ.git ←
11  fetch = +refs/heads/*:refs/remotes/origin/*
12 [branch "master"]
13   remote = origin
14   merge = refs/heads/master
15 [remote "seginfoFAQ"]
16   url = Royal16969/seginfoFAQ
17   fetch = +refs/heads/*:refs/remotes/seginfoFAQ/*
18

```

Le damos a guardar (Ctrl s), y volvemos a probar a ejecutar el comando del *push*...



```

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ git push origin
Enumerating objects: 29, done.
Counting objects: 100% (29/29), done.
Delta compression using up to 4 threads
Compressing objects: 100% (23/23), done.
Writing objects: 100% (27/27), 2.55 KiB | 200.00 KiB/s, done.
Total 27 (delta 12), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (12/12), completed with 1 local object.
To github.com:Royal16969/seginfoFAQ.git
  4b91a4f..c0b0e5c master -> master

C:\Users\Sergio DC\Desktop\Sistemas Informaticos\Tema 9\UD9. Practica 2. GITHUB by SSH\seginfoFAQ (master -> origin)
λ

```

Y efectivamente, ya sí que se suben a GitHub los cambios y se actualiza nuestro repositorio en GitHub.

Royal6969 / seginfoFAQ
forked from mattaereal/seginfoFAQ

Code Pull requests Actions Projects Wiki Security Insights Settings

master 1 branch 0 tags Go to file Add file Code

This branch is 10 commits ahead of mattaereal:master. Pull request Compare

| | Royal6969 Merge branch 'feature-c' | c0b0e5c 2 days ago 45 commits |
|--|--------------------------------------|---------------------------------------|
| | Cybersecurity Domains v2 Spanish.png | Cybersecurity Domain v2 13 months ago |
| | Glosario.md | Adding Glosario.md 11 months ago |
| | README.md | Revert "Add existing file" 2 days ago |
| | feature-a.js | added feature-a.js file 2 days ago |
| | feature-b.js | added feature-b.js file 2 days ago |
| | index.html | change index title 2 days ago |
| | style.css | Merge branch 'feature-c' 2 days ago |

README.md

Entrega de la Práctica

Me tenéis que entregar un documento indicando en cada uno de los ejercicios:

- Vuestro usuario de github
- Enlace de vuestro repositorio personal.
- Imágenes de los pasos realizados para cada ejercicio.