

EN ESTE INFORME SE EXPLICA TODO LO QUE HAY QUE HACER CUANDO ALGUIEN YA HIZO MERGE.

PASO 1: SE PONE EL COMANDO INIT PARA INICIALIZAR

PASO 2: SE PONE EL COMANDO GIT STATUS PARA MIRAR EL ESTADO DE NUESTROS ARCHIVOS, EN ESTE CASO VEMOS QUE HAY UN ARCHIVO QUE NO ESTA SUBIDO Y SU NOMBRE ES PARTE2.PY

```
BOGDFPCSRFSP406+Aprendiz@BOGDFPCSRFSP406 MINGW64 ~/Desktop/guia2 (master)
• $ git init
Initialized empty Git repository in C:/Users/Aprendiz/Desktop/guia2/.git/

BOGDFPCSRFSP406+Aprendiz@BOGDFPCSRFSP406 MINGW64 ~/Desktop/guia2 (master)
• $ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
  parte2.py
```

PASO 3: SE PONE EL COMANDO ADD PARA AGREGAR EL ARCHIVO QUE NO ESTA SUBIDO.

PASO 4: SE PONE EL COMANDO GIT STATUS DE NUEVO PARA VER SI SE AGREGO EL ARCHIVO, PODEMOS VER QUE SI YA QUE NUEVAMENTE LO PODEMOS VER CON UN COMENTARIO QUE DICE "NEW FILE: PARTE2.PY"

```
BOGDFPCSRFSP406+Aprendiz@BOGDFPCSRFSP406 MINGW64 ~/Desktop/guia2 (master)
• $ git add .

BOGDFPCSRFSP406+Aprendiz@BOGDFPCSRFSP406 MINGW64 ~/Desktop/guia2 (master)
• $ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   parte2.py
```

PASO 5: PONEMOS EL COMANDO GIT COMMIT -M "COMENTARIO DEL ARCHIVO" (ESTO ES MUY IMPORTANTE PARA SABER QUE SE MODIFICO O DE QUE TRATA ESE ARCHIVO, SON BUENAS PRACTICAS EN EL MUNDO DE LA PROGRAMACIÓN)

```
BOGDFPCSRFSP406+Aprendiz@BOGDFPCSRFSP406 MINGW64 ~/Desktop/guia2 (master)
• $ git commit -m "parte resta"
[master (root-commit) 62da3e9] parte resta
 1 file changed, 4 insertions(+)
 create mode 100644 parte2.py
```

PASO 6: PONEMOS EL COMANDO GIT REMOTE ADD ORIGIN LINK\_DEL\_REPOSITORIO, CON EL FIN DE QUE GIT BASH SE CONECTE CON EL REPOSITORIO DE GIT HUB Y SE SUBA TODO LO QUE QUEREMOS

```
BOGDFPCSRFSP406+Aprendiz@BOGDFPCSRFSP406 MINGW64 ~/Desktop/guia2 (master)
• $ git remote add origin https://github.com/COMODOJ21/PARTE2_GUIA.git
```

PASO 7: PONEMOS EL COMANDO GIT BRANCH -LIST PARA VER QUE RAMAS TENEMOS

PASO 8: PONEMOS EL COMANDO GIT BRANCH NOMBRE\_DE\_LA\_NUEVA\_RAMA PARA CREAR UNA NUEVA RAMA.

PASO 9: PONEMOS EL COMANDO GIT BRANCH -LIST, PARA VER SI SE CREO CORRECTAMENTE NUESTRA RAMA.

```
BOGDFPCSRFSP406+Aprendiz@BOGDFPCSRFSP406 MINGW64 ~/Desktop/guia2 (master)
• $ git branch --list
* master

BOGDFPCSRFSP406+Aprendiz@BOGDFPCSRFSP406 MINGW64 ~/Desktop/guia2 (master)
• $ git branch Isabella

BOGDFPCSRFSP406+Aprendiz@BOGDFPCSRFSP406 MINGW64 ~/Desktop/guia2 (master)
• $ git branch --list
  Isabella
* master
```

PASO 10: LUEGO PONEMOS EL COMANDO GIT CHECKOUT NOMBRE\_DE\_LA\_RAMA, ESTO PARA UBICARNOS EN LA RAMA EN LA QUE QUEREMOS SUBIR EL ARCHIVO.

```
BOGDFPCSRFSP406+Aprendiz@BOGDFPCSRFSP406 MINGW64 ~/Desktop/guia2 (master)
• $ git checkout Isabella
Switched to branch 'Isabella'
```

PASO 11: PONEMOS EL COMANDO PUSH -U ORIGIN RAMA\_EN\_DONDE\_ESTAS, ESTO PARA SUBIR EL ARCHIVO AL GIT HUB

```
BOGDFPCSRFSP406+Aprendiz@BOGDFPCSRFSP406 MINGW64 ~/Desktop/guia2 (Isabella)
• $ git push -u origin Isabella
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 16 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 284 bytes | 284.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'Isabella' on GitHub by visiting:
remote:   https://github.com/COMODOJ21/PARTE2_GUIA/pull/new/Isabella
remote:
To https://github.com/COMODOJ21/PARTE2_GUIA.git
 * [new branch]      Isabella -> Isabella
```

PASO 12: LUEGO SE PONE EL COMANDO GIT CHECKOUT PARA ESTAR EN LA RAMA MASTER

```
BOGDFPCSRFSP406+Aprendiz@BOGDFPCSRFSP406 MINGW64 ~/Desktop/guia2 (Isabella)
• $ git checkout master
Switched to branch 'master'
```

PASO 13: AGREGA EL GIT PULL ORIGIN MASTER --ALLOW-UNRELATED-HISTORIES, ESTO PARA HACER EL MERGE, EN ESTA PARTE APARECERA EL CODIGO DE LA OTRA RAMA EN LA PARTER DE ARRIBA ABRA UNAS OPCIONES

```
BOGDFPCSRFSP406+Aprendiz@BOGDFPCSRFSP406 MINGW64 ~/Desktop/guia2 (master)
③ $ git pull origin master --allow-unrelated-histories
From https://github.com/COMODOJ21/PARTE2_GUIA
 * branch          master      -> FETCH_HEAD
Auto-merging parte2.py
CONFLICT (add/add): Merge conflict in parte2.py
Automatic merge failed; fix conflicts and then commit the result.
```

PASO 14: AGRAGA LOS SIGUIENTES COMANDOS:

GIT ADD

GIT COMMIT -M "COMENTARIO"

GIT PUSH ORIGIN MASTER

CON ESTO SE SUBIRA EN EL REPOSITORIO CORRECTAMENTE

```
BOGDFPCSRFSP406+Aprendiz@BOGDFPCSRFSP406 MINGW64 ~/Desktop/guia2 (master|MERGING)
$ git add .

BOGDFPCSRFSP406+Aprendiz@BOGDFPCSRFSP406 MINGW64 ~/Desktop/guia2 (master|MERGING)
$ git commit -m "Resuelto conflicto de fusión"
[master 06e608d] Resuelto conflicto de fusión

BOGDFPCSRFSP406+Aprendiz@BOGDFPCSRFSP406 MINGW64 ~/Desktop/guia2 (master)
$ git push origin master
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 16 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 449 bytes | 449.00 KiB/s, done.
```