

# PRÁCTICA ALEJANDRO MARTÍNEZ MARTÍNEZ DAM 2CFSU

## 2. Fundamentos de Git I

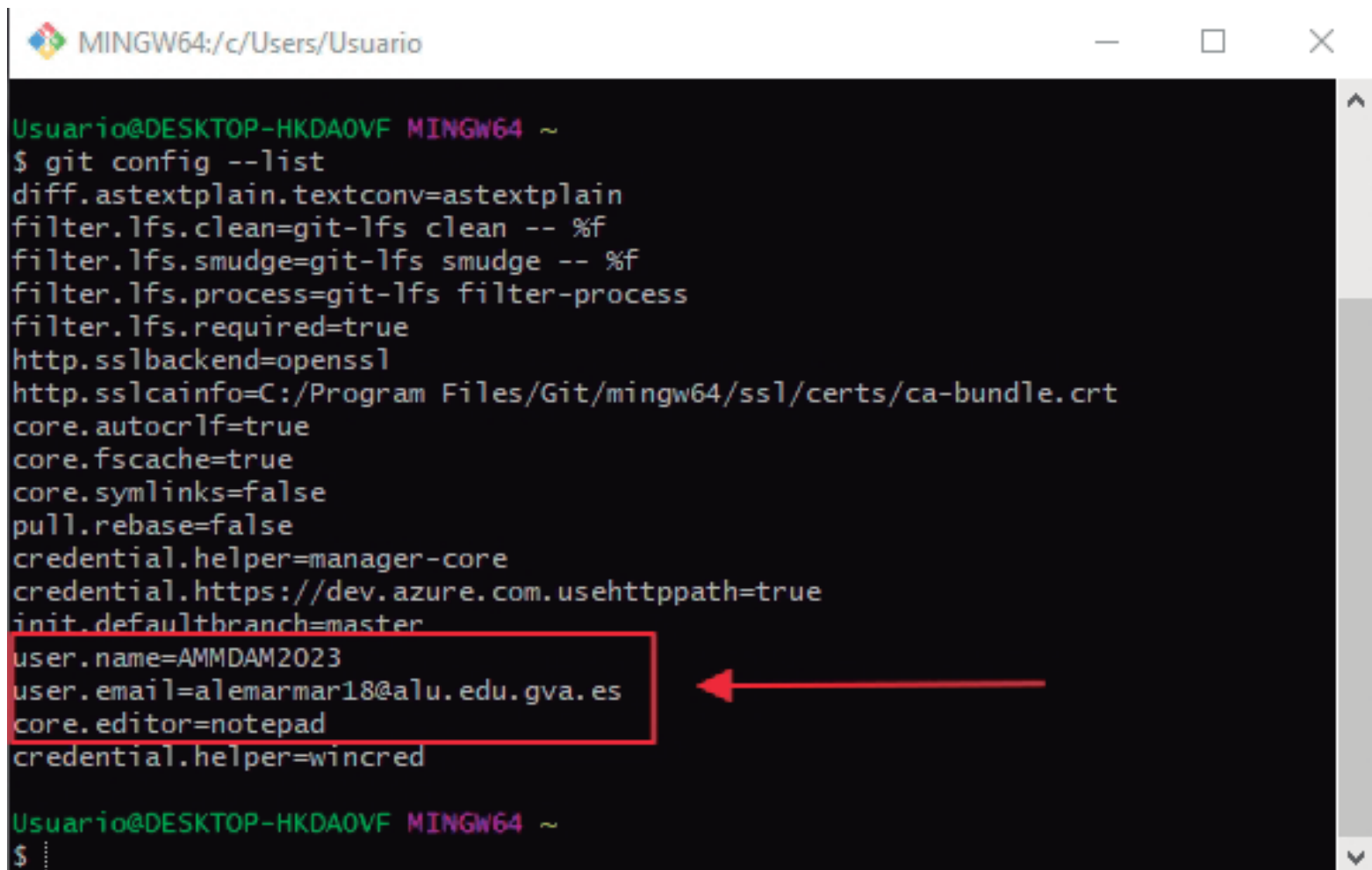
1. Instala Git en tu sistema operativo. Adjunta una captura de pantalla en la que aparezca el resultado de la ejecución del comando `git --version`.



```
MINGW64:/c/Users/Usuario

Usuario@DESKTOP-HKDA0VF MINGW64 ~
$ git --version
git version 2.33.0.windows.2
```

2. Realiza la configuración de Git según lo indicado en el tema (nombre, correo electrónico y editor de preferencia). Adjunta una captura de pantalla con el resultado de la ejecución de los comandos de configuración.



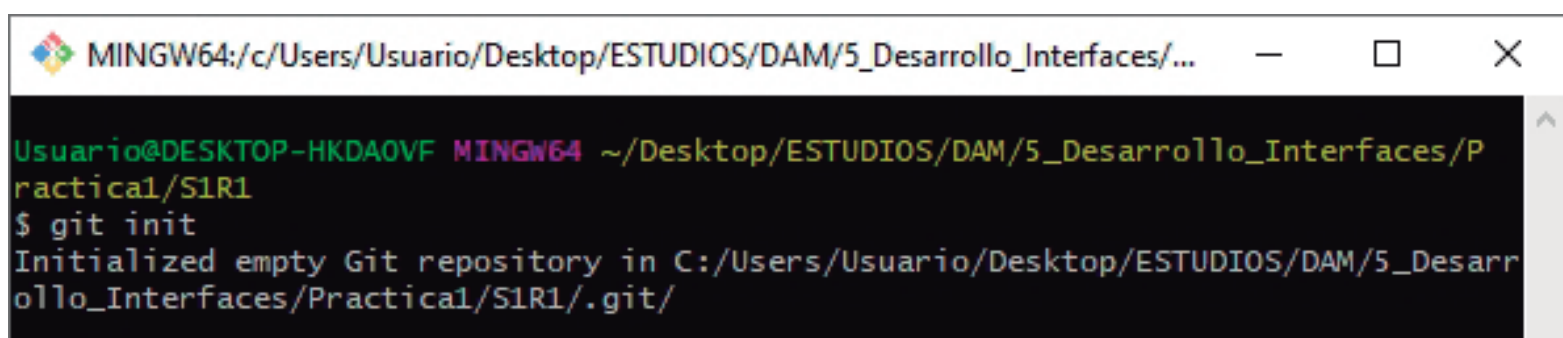
```
MINGW64:/c/Users/Usuario

Usuario@DESKTOP-HKDA0VF MINGW64 ~
$ git config --list
diff.astextplain.textconv=astextplain
filter.lfs.clean=git-lfs clean -- %f
filter.lfs.smudge=git-lfs smudge -- %f
filter.lfs.process=git-lfs filter-process
filter.lfs.required=true
http.sslbackend=openssl
http.sslcainfo=C:/Program Files/Git/mingw64/ssl/certs/ca-bundle.crt
core.autocrlf=true
core.fscache=true
core.symlinks=false
pull.rebase=false
credential.helper=manager-core
credential.https://dev.azure.com.usehttppath=true
init.defaultbranch=master
user.name=AMMDAM2023
user.email=alemarmar18@alu.edu.gva.es
core.editor=notepad
credential.helper=wincred

Usuario@DESKTOP-HKDA0VF MINGW64 ~
$
```

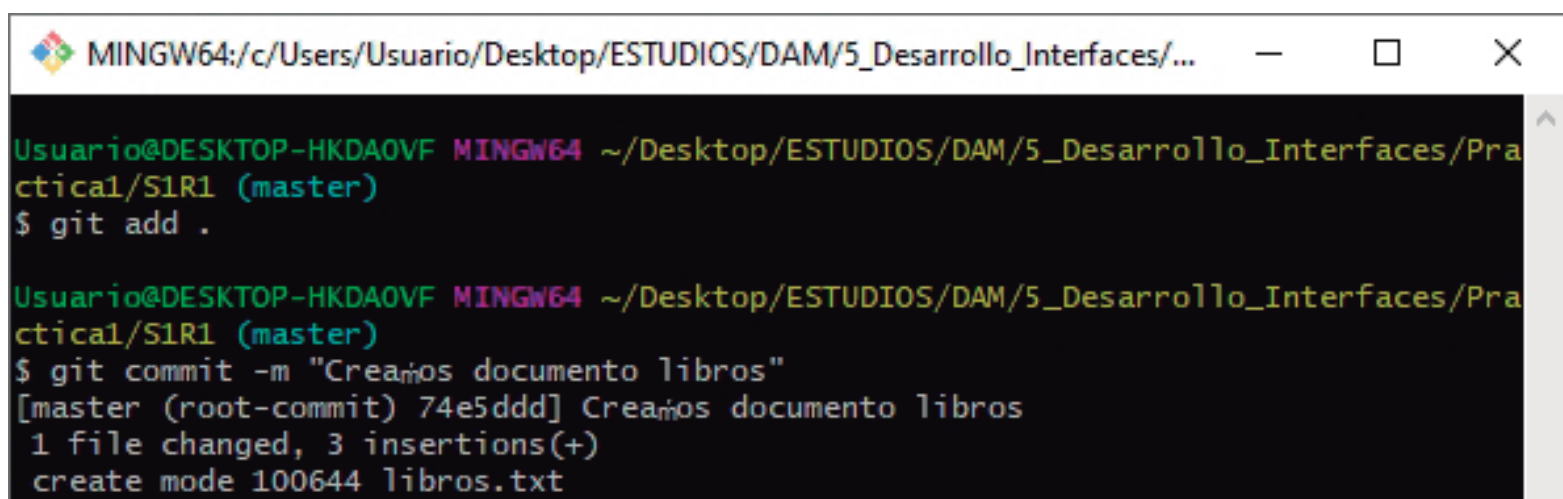
3. Crea una carpeta denominada S1R1. Realiza las siguientes acciones en ella:

- Crea un repositorio Git.
- Crea un fichero denominado libros.txt. Añade tres títulos de libros cada uno en una línea distinta.
- Haz un primer commit.



```
MINGW64:/c/Users/Usuario/Desktop/ESTUDIOS/DAM/5_Desarrollo_Interfases/...

Usuario@DESKTOP-HKDA0VF MINGW64 ~/Desktop/ESTUDIOS/DAM/5_Desarrollo_Interfases/P
ractical/S1R1
$ git init
Initialized empty Git repository in C:/Users/Usuario/Desktop/ESTUDIOS/DAM/5_Desar
ollo_Interfases/Practical/S1R1/.git/
```



```
MINGW64:/c/Users/Usuario/Desktop/ESTUDIOS/DAM/5_Desarrollo_Interfases/...

Usuario@DESKTOP-HKDA0VF MINGW64 ~/Desktop/ESTUDIOS/DAM/5_Desarrollo_Interfases/Pra
ctical/S1R1 (master)
$ git add .

Usuario@DESKTOP-HKDA0VF MINGW64 ~/Desktop/ESTUDIOS/DAM/5_Desarrollo_Interfases/Pra
ctical/S1R1 (master)
$ git commit -m "Creañs documento libros"
[master (root-commit) 74e5ddd] Creañs documento libros
1 file changed, 3 insertions(+)
create mode 100644 libros.txt
```

- d) Añade dos libros al archivo libros.txt.
- e) Haz un segundo commit.

```
Usuario@DESKTOP-HKDA0VF MINGW64 ~/Desktop/ESTUDIOS/DAM/5_DesarrolloInterfaces/Practica1/S1R1 (master)
$ git add .

Usuario@DESKTOP-HKDA0VF MINGW64 ~/Desktop/ESTUDIOS/DAM/5_DesarrolloInterfaces/Practica1/S1R1 (master)
$ git commit -m "Añadimos dos libros más"
[master 8daa930] Añadimos dos libros más
1 file changed, 3 insertions(+), 1 deletion(-)
```

- f) Crea un fichero denominado peliculas.txt. Añade tres títulos de películas a dicho archivo.
- g) Haz una captura de pantalla del comando git status.

```
MINGW64:/c/Users/Usuario/Desktop/ESTUDIOS/DAM/5_DesarrolloInterfaces/Prac...
Usuario@DESKTOP-HKDA0VF MINGW64 ~/Desktop/ESTUDIOS/DAM/5_DesarrolloInterfaces/Practica1/S1R1 (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    peliculas.txt

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

Usuario@DESKTOP-HKDA0VF MINGW64 ~/Desktop/ESTUDIOS/DAM/5_DesarrolloInterfaces/Practica1/S1R1 (master)
$
```

- h) Crea un fichero denominado comidas.txt. Añade tres nombres de comidas a dicho archivo.
- i) Haz un tercer commit que incluya los archivos peliculas.txt y comidas.txt.

```
MINGW64:/c/Users/Usuario/Desktop/ESTUDIOS/DAM/5_DesarrolloInterfaces/Prac...
Usuario@DESKTOP-HKDA0VF MINGW64 ~/Desktop/ESTUDIOS/DAM/5_DesarrolloInterfaces/Practica1/S1R1 (master)
$ git commit -m "Añadimos archivo peliculas y comidas"
[master 4502a82] Añadimos archivo peliculas y comidas
2 files changed, 6 insertions(+)
create mode 100644 comidas.txt
create mode 100644 peliculas.txt
```

- j) Elimina el archivo comidas.txt desde el navegador de archivos.
- k) Añade dos películas más al archivo peliculas.txt.
- l) Haz una captura de pantalla que muestre los cambios en el directorio de trabajo.

```
MINGW64:/c/Users/Usuario/Desktop/ESTUDIOS/DAM/5_DesarrolloInterfaces/Prac...
Usuario@DESKTOP-HKDA0VF MINGW64 ~/Desktop/ESTUDIOS/DAM/5_DesarrolloInterfaces/Practica1/S1R1 (master)
$ git status
On branch master
Changes not staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    deleted:    comidas.txt
    modified:   peliculas.txt

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

- m) Añade los cambios al área de preparación.  
 n) Haz una captura de pantalla del comando git status. Debe indicar que se ha borrado el archivo comidas.txt y que se ha modificado el archivo peliculas.txt.

```

MINGW64:/c/Users/Usuario/Desktop/ESTUDIOS/DAM/5_Desarrollo_Interfases/Prac...
Usuario@DESKTOP-HKDA0VF MINGW64 ~/Desktop/ESTUDIOS/DAM/5_Desarrollo_Interfases/Practi
ca1/S1R1 (master)
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        deleted:    comidas.txt
        modified:   peliculas.txt
  
```

- ñ) Haz un cuarto commit.

```

MINGW64:/c/Users/Usuario/Desktop/ESTUDIOS/DAM/5_Desarrollo_Interfases/Prac...
Usuario@DESKTOP-HKDA0VF MINGW64 ~/Desktop/ESTUDIOS/DAM/5_Desarrollo_Interfases/Practi
ca1/S1R1 (master)
$ git commit -m "Modificamos peliculas y borramos comidas"
[master 1526d5b] Modificamos peliculas y borramos comidas
2 files changed, 3 insertions(+), 4 deletions(-)
delete mode 100644 comidas.txt
  
```

- o) Crea un archivo denominado datos.bak. Añade tres títulos de li- bros a dicho archivo. ¡IMPORTANTE! No añadas el archivo al área de prepara- ción ni hagas ningún commit.  
 p) Crea una subcarpeta denominada output. Crea un archivo denomi- nado salida.txt en su interior. Escribe tu nombre y apellidos en dicho archivo. ¡IMPORTANTE! No añadas los archivos al área de preparación ni hagas ningún commit.  
 q) Haz una captura de pantalla del comando git status. Deben apa- recer los archivos datos.bak y output/salida.txt como ar- chivos nuevos (color rojo).

The screenshot shows a Windows Explorer window with the following file structure:

Nombre	Fecha de modificación	Tipo	Tamaño
.git	03/10/2023 13:04	Carpeta de archivos	
output	03/10/2023 13:03	Carpeta de archivos	
datos	03/10/2023 13:02	Archivo BAK	1 KB
libros	03/10/2023 12:40	Documento de te...	1 KB
peliculas	03/10/2023 12:52	Documento de te...	1 KB

The terminal window shows the following output:

```

MINGW64:/c/Users/Usuario/Desktop/ESTUDIOS/DAM/5_Desarrollo_Interfases/Prac...
Usuario@DESKTOP-HKDA0VF MINGW64 ~/Desktop/ESTUDIOS/DAM/5_Desarrollo_Interfases/Practi
ca1/S1R1 (master)
$ git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        datos.bak
        output/

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

- r) Crea un archivo .gitignore para que los ficheros con extensión .bak y el contenido de la carpeta output/ no se incluyan en el repositorio.

The screenshot shows a Windows Explorer window with the following file structure:

Nombre	Fecha de modificación	Tipo	Tamaño
.git	03/10/2023 13:04	Carpeta de archivos	
output	03/10/2023 13:03	Carpeta de archivos	
.gitignore	03/10/2023 13:09	Documento de te...	1 KB
datos	03/10/2023 13:02	Archivo BAK	1 KB
libros			
peliculas			

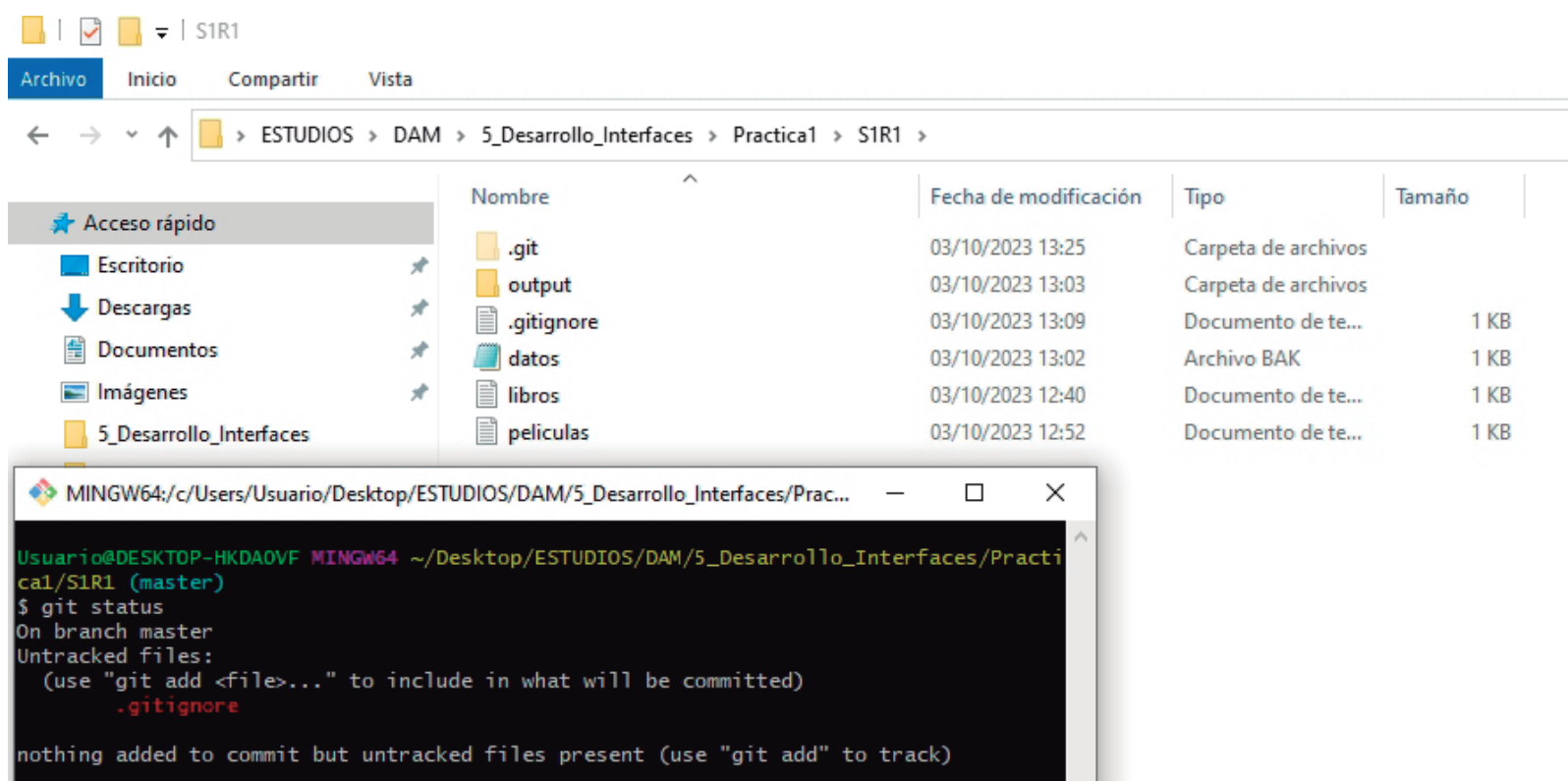
The .gitignore file content is shown in a text editor:

```

*.bak
output/
  
```



s) Haz una nueva captura de pantalla del comando git status. Ahora no deben aparecer los archivos datos.bak y output/salida.txt como archivos nuevos, sino que en su lugar debe aparecer únicamente el archivo .gitignore.



t) Haz un último commit para incluir el archivo .gitignore en el re-positorio.



u) Haz una captura de pantalla que muestre el histórico de cambios del repositorio.

