

Examen Entornos de Desarrollo. IES Marcos Zaragoza

Recuperación Git. 14/05/24

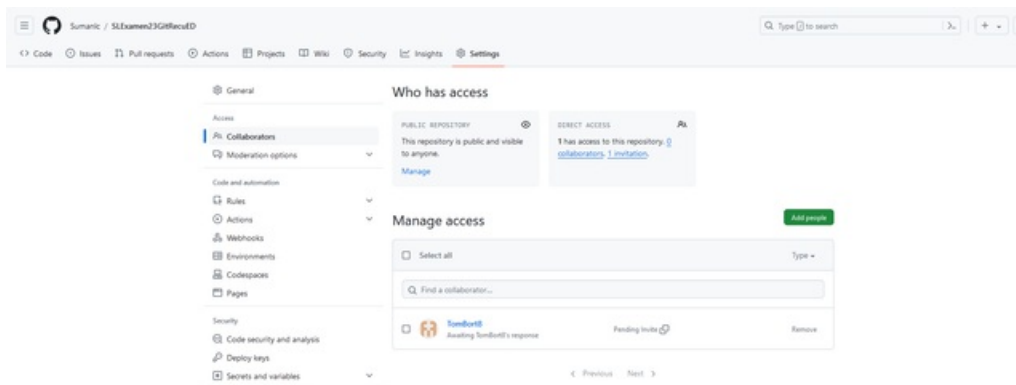
Nombre y apellidos: **Suman Lamichhane**

LINK DEL REPOSITORIO: <https://github.com/Sumanlc/SLExamen23GitRecuED.git>

Vamos a trabajar sobre Strings.

Debajo de cada ejercicio deberán aparecer las capturas de pantalla pertinentes que justifiquen su realización. Intentad hacer todo lo posible desde la consola si no se indica lo contrario. Se valorará negativamente las malas prácticas de GIT

1.- (0,5) Crea un repositorio privado en github "AAExamen23GitRecuED" e invítame a colaborar: "TomBort8" . AAA serán las primeras letras de tu nombre, 1er apellido y 2º apellido respectivamente.



2.- (0,5) Inicializa el repositorio en local y vincúlalo al repositorio de github

```
dam_ls@A111 MINGW64 ~/Documents/GitRecuperacion
$ git init
Initialized empty Git repository in C:/Users/dam_ls/Documents/GitRecuperacion/.git/

dam_ls@A111 MINGW64 ~/Documents/GitRecuperacion (master)
$ git config --global user.email "sumlam@alu.edu.gva.es"

dam_ls@A111 MINGW64 ~/Documents/GitRecuperacion (master)
$ git init SLExamen23GitRecuED
Initialized empty Git repository in C:/Users/dam_ls/Documents/GitRecuperacion/SLExamen23GitRecuED/.git/

dam_ls@A111 MINGW64 ~/Documents/GitRecuperacion (master)
$ |
```

3.- Crea un main que pida una cadena por teclado .

```
import java.util.Scanner;

public class ProcesamientoTexto {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.println("Por favor, ingresa un texto:");
        String texto = scanner.nextLine();
        scanner.close();
    }
}
```

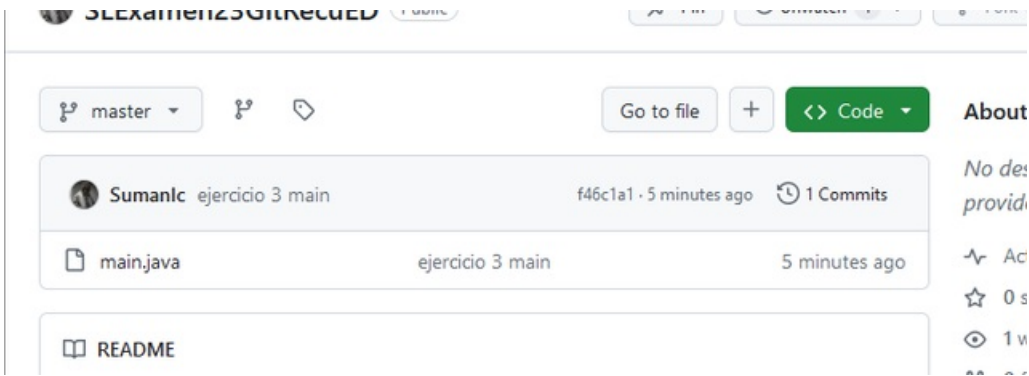
Sube los cambios al repositorio.

Nombre	Fecha de modificación	Tipo	Tamaño
.git	14/05/2024 12:28	Carpeta de archivos	
main	14/05/2024 12:26	Archivo de origen ...	1 KB

```
main.java x
C: > Users > dam_ls > Documents > GitRecuperacion > SLExamen23GitRecuED > main.java
1  import java.util.Scanner;
2
3  public class ProcesamientoTexto {
4
5      public static void main(String[] args) {
6          Scanner scanner = new Scanner(System.in);
7
8          System.out.println("Por favor, ingresa un texto:");
9          String texto = scanner.nextLine();
10         scanner.close();
11     }
12 }
13
```

```
dam_ls@A111 MINGW64 ~/Desktop/SLExamen23GitRecuED (master)
$ git push -u origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 6 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 393 bytes | 393.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Suman1c/SLExamen23GitRecuED.git
 * [new branch]      master -> master
branch 'master' set up to track 'origin/master'.

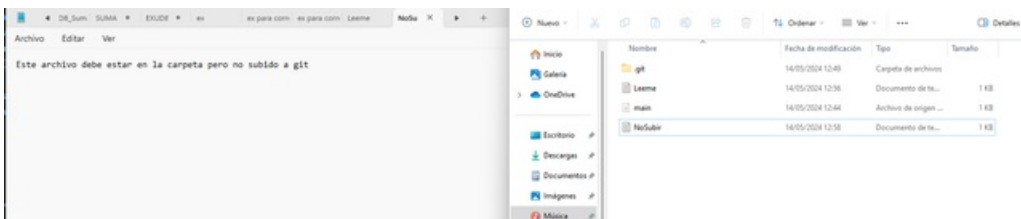
dam_ls@A111 MINGW64 ~/Desktop/SLExamen23GitRecuED (master)
```



4.- (0,2) Crea un fichero Leeme.txt : “Vamos a trabajar sobre Strings con una clase propia, crearemos Ramas para que cada miembro trabaje su parte y luego las uniremos todas”.



- 4.1 (0,3) Crea también un fichero de texto que no debes subir a github pero debe estar dentro de la carpeta NoSubir.txt: (Este archivo debes añadirlo y quitarlo, como si te hubieras confundido). “Este archivo debe estar en la carpeta pero no subido a git”.



Muestra por comandos que no lo has subido

```
dam_ls@A111 MINGW64 ~/Desktop/SLEexamen23GitRecuED (master)
$ git commit -m "ejercicio 4"
[master 9dad7b3] ejercicio 4
2 files changed, 3 insertions(+)
create mode 100644 Leeme.txt
create mode 100644 NoSubir.txt

dam_ls@A111 MINGW64 ~/Desktop/SLEexamen23GitRecuED (master)
$ git reset NoSubir.txt

dam_ls@A111 MINGW64 ~/Desktop/SLEexamen23GitRecuED (master)
$ git push -u origin master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 6 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 470 bytes | 470.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Sumanlc/SLEexamen23GitRecuED.git
f46c1a1..9dad7b3 master -> master
branch 'master' set up to track 'origin/master'.
```

Sube los cambios al repositorio.

5.- (0,5) Muestra la diferencia entre los 2 últimos commits.

```
dam_ls@A111 MINGW64 ~/Desktop/SLExamen23GitRecuED (master)
$ git show
commit 9dad7b37537264433f8c55c6aa25c75100812472 (HEAD -> master, origin/master)
author: sumanlc <sumlam@alu.edu.gva.es>
date: Tue May 14 13:01:26 2024 +0200

    ejercicio 4

diff --git a/Leeme.txt b/Leeme.txt
new file mode 100644
index 0000000..c2b4658
--- /dev/null
+++ b/Leeme.txt
@@ -0,0 +1,2 @@

Vamos a trabajar sobre Strings con una clase propia, crearemos Ramas para que cada miembro trabaje su parte y luego las uniremos todas
diff --git a/NoSubir.txt b/NoSubir.txt
new file mode 100644
index 0000000..aa35068
--- /dev/null
+++ b/NoSubir.txt
@@ -0,0 +1 @@
Este archivo debe estar en la carpeta pero no subido a git
```

6.- (0,5) Crea 2 ramas Alicia y Bob

```
dam_ls@A111 MINGW64 ~/Desktop/SLExamen23GitRecuED (master)
$ git branch Alicia

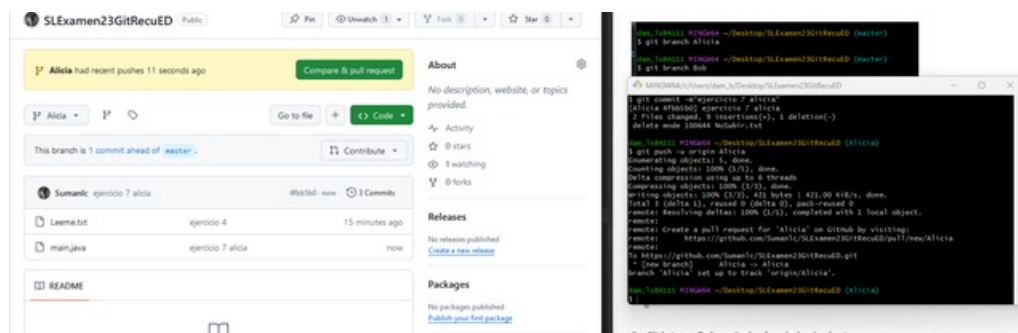
dam_ls@A111 MINGW64 ~/Desktop/SLExamen23GitRecuED (master)
$ git branch Bob
```

7.- Sitúate en Alicia y añade al main las siguientes funcionalidades:

```
public static String convertirAMayusculas(String texto) {
    return texto.toUpperCase();
}

public static String recortarEspacios(String texto) {
    return texto.trim();
}
```

Sube los cambios al repositorio (rama Alicia).



8.- Sitúate en Bob y añade al main lo siguiente:

```

public static String convertirAMinusculas(String texto) {
    return texto.toLowerCase();
}

public static String invertirTexto(String texto) {
    return new StringBuilder(texto).reverse().toString();
}

```

Sube los cambios al repositorio (rama Bob).

```

dam_ls@A111 MINGW64 ~/Desktop/SLExamen23GitRecuED (Bob)
$ git commit -m"ejercicio 8 Bob"
[Bob c1bd2c8] ejercicio 8 Bob
1 file changed, 10 insertions(+)

dam_ls@A111 MINGW64 ~/Desktop/SLExamen23GitRecuED (Bob)
$ git push -u origin Bob
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 6 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 410 bytes | 410.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
remote:
remote: Create a pull request for 'Bob' on GitHub by visiting:
remote:   https://github.com/Sumanlc/SLExamen23GitRecuED/pull/new/Bob
remote:
To https://github.com/Sumanlc/SLExamen23GitRecuED.git
 * [new branch]      Bob -> Bob
branch 'Bob' set up to track 'origin/Bob'.

```

9.- (1) Muestra la diferencia entre las ramas Alicia y Bob y guárdalo en un fichero llamado DIFERENCIA_RAMAS (desde consola). Este fichero debe subirse al repositorio.

```

dam_ls@A111 MINGW64 ~/Desktop/SLExamen23GitRecuED (master)
$ git diff Alicia Bob
diff --git a/NoSubir.txt b/NoSubir.txt
new file mode 100644
index 0000000..aa35068
--- /dev/null
+++ b/NoSubir.txt
@@ -0,0 +1 @@
+Este archivo debe estar en la carpeta pero no subido a git
diff --git a/main.java b/main.java
index 4ee4d24..df3a48c 100644
--- a/main.java
+++ b/main.java
@@ -10,12 +10,13 @@ public class ProcesamientoTexto {
    scanner.close();
}

- public static String convertirAMayusculas(String texto) {
-     return texto.toUpperCase();
+ public static String convertirAMinusculas(String texto) {
+     return texto.toLowerCase();
+ }

```

```

dam_ls@A111 MINGW64 ~/Desktop/SLEexamen23GitRecuED (master)
$ git diff Alicia Bob > DIFERENCIAS_RAMAS

dam_ls@A111 MINGW64 ~/Desktop/SLEexamen23GitRecuED (master)
$ git add DIFERENCIAS_RAMAS
warning: in the working copy of 'DIFERENCIAS_RAMAS', LF will be replaced by
time Git touches it

dam_ls@A111 MINGW64 ~/Desktop/SLEexamen23GitRecuED (master)
$ git commit "ejercicio 9"
error: pathspec 'ejercicio 9' did not match any file(s) known to git

dam_ls@A111 MINGW64 ~/Desktop/SLEexamen23GitRecuED (master)
$ git commit -m "ejercicio 9"
[master d788eb1] ejercicio 9
1 file changed, 29 insertions(+)
create mode 100644 DIFERENCIAS_RAMAS

dam_ls@A111 MINGW64 ~/Desktop/SLEexamen23GitRecuED (master)
$ git push
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.

```

10.- (1,5) Fusi3nalo en main (consola) y resuelve el conflicto (en gitHUB).

Sube los cambios al repositorio.

```

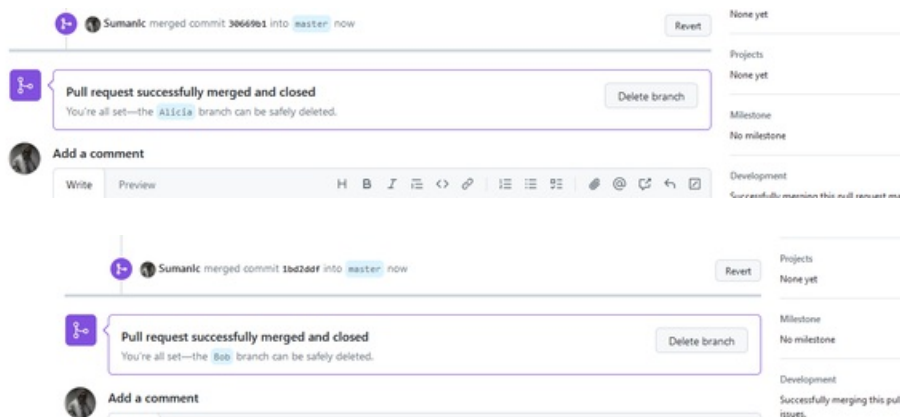
dam_ls@A111 MINGW64 ~/Desktop/SLEexamen23GitRecuED (master)
$ git merge Alicia
Merge made by the 'ort' strategy.
 NoSubir.txt | 1 -
 main.java   | 9 ++++++++
 2 files changed, 9 insertions(+), 1 deletion(-)
 delete mode 100644 NoSubir.txt

```

```

dam_ls@A111 MINGW64 ~/Desktop/SLEexamen23GitRecuED (master)
$ git merge Bob
Auto-merging main.java
CONFLICT (content): Merge conflict in main.java
Automatic merge failed; fix conflicts and then commit the result.

```



11.- (0,5) Borra las ramas Alice Y Bob.

```

dam_1s@A111 MINGW64 ~/Desktop/SLExamen23GitRecuED (master)
$ git merge Bob
Auto-merging main.java
CONFLICT (content): Merge conflict in main.java
Automatic merge failed; fix conflicts and then commit the result.

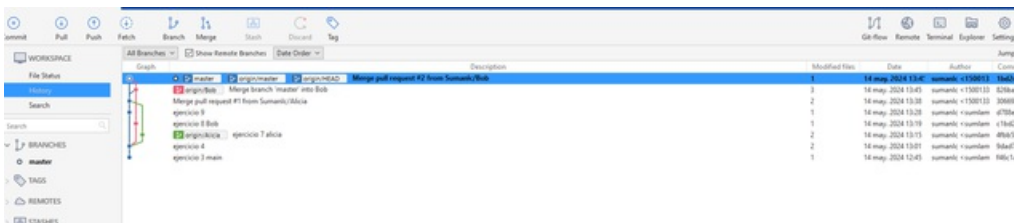
dam_1s@A111 MINGW64 ~/Desktop/SLExamen23GitRecuED (master|MERGING)
$ git branch -d Alicia
Deleted branch Alicia (was 4fbb5b0).

dam_1s@A111 MINGW64 ~/Desktop/SLExamen23GitRecuED (master|MERGING)
$ git branch -d Bob
warning: deleting branch 'Bob' that has been merged to
'refs/remotes/origin/Bob', but not yet merged to HEAD.
Deleted branch Bob (was c1bd2c8).

dam_1s@A111 MINGW64 ~/Desktop/SLExamen23GitRecuED (master|MERGING)

```

12.- (0,5) Entra a SOURCETREE y haz una captura del eje temporal del repositorio. Haz una breve explicación de lo que observas.



13.- (0,75) ¿Cuál es la diferencia entre “git pull” y “git clone” ?

git clone: lo utilizamos para empezar a trabajar ya que agrega el repositorio remoto a tu area de trabajo. **git pull:** descarga una actualización del estado del repositorio remoto al repositorio local y a los archivos (osea combina un fetch y un merge).

14.- (1,25) Abre el main y déjalo inservible poniendo el tipo de dato a StringBuilder. Sube los cambios. A continuación deshaz el último commit.

15.- (1,25) Vuelve al estado en el que estaba el proyecto al acabar el ejercicio 3 en local.

16.- (0,75) Añade este documento al repoitorio, con todas las imágenes para que se pueda ver desde git.

17.- Por último, ejecutad el siguiente comando:

```
history > historial.txt
```

sube el resultado a aules junto al PDF de este documento.