

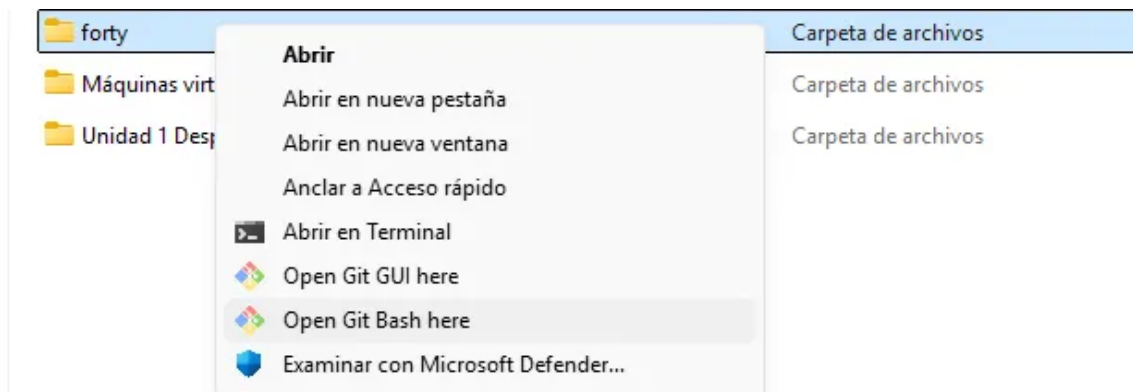


Práctica GIT (Forty)

Trabajo en local

1. Inicializa un nuevo repositorio Git en una carpeta llamada "forty" y agrega los archivos proporcionados en el aula virtual.

Abrimos Git Bash desde la carpeta donde se encuentran los archivos.



```
git init
git add .
```

2. Renombra la rama master a main

```
git branch -m master main
```

3. Haz que los ficheros README.txt , LICENSE.txt y passwords.txt sean ignorados por el control de versiones

Añadimos los archivos que queremos ignorar a .gitignore

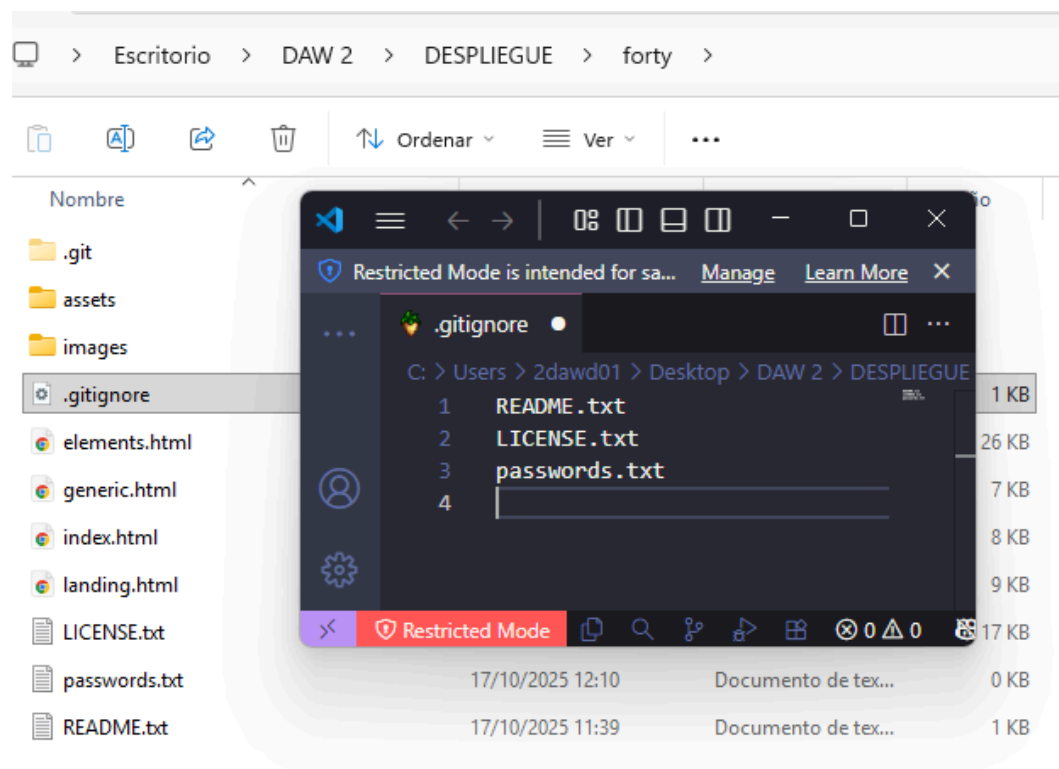
```
> cat .gitignore
README.txt
LICENSE.txt
passwords.txt
```

Añadimos también el siguiente comando para eliminar del repositorio pero para que permanezca en mi directorio de trabajo como archivo ignorado

```
git rm --cached README.txt LICENSE.txt
```

Después añadimos .gitignore al repositorio y hacemos un commit

```
git add .gitignore  
git commit -m ".gitignore añadido"
```



4. Crea el archivo passwords.txt . Comprueba que el control de versiones lo ignora

```
touch passwords.txt  
git status
```

5. Crea una rama llamada "feature-content" . Muévete a esa rama. Cambia, en la línea 3477, el font-size por 1.5em en el archivo main.css . Confirma cambios y haz commit. Muestra los logs de la forma más gráfica posible.

```
git branch feature-content
git checkout feature-content
git add .
git commit -m "Cambio en la línea 3477 del main.css"
git log
```

6. Elimina el archivo "passwords.txt" en la carpeta forty . Verifica el estado del repositorio. ¿Hay cambios pendientes?

```
rm passwords.txt
git add .
git commit -m "Archivo passwords.txt borrado"
git status
```

```
2dawd01@A205PC05 MINGW64 ~/Desktop/DAW 2/DESPLIEGUE/forty (feature-content)
$ rm passwords.txt

2dawd01@A205PC05 MINGW64 ~/Desktop/DAW 2/DESPLIEGUE/forty (feature-content)
$ git add .

2dawd01@A205PC05 MINGW64 ~/Desktop/DAW 2/DESPLIEGUE/forty (feature-content)
$ git commit -m "Archivo passwords.txt borrado"
On branch feature-content
nothing to commit, working tree clean

2dawd01@A205PC05 MINGW64 ~/Desktop/DAW 2/DESPLIEGUE/forty (feature-content)
$ git status
On branch feature-content
nothing to commit, working tree clean
```

7. Crea un nuevo archivo llamado " about.html ", partiendo del archivo generic.html y agrégalo al repositorio, haz un nuevo commit.

```
cp generic.html about.html
git add .
git commit -m "Archivo about.html creado"
```

8. Cambia a la rama main . Examina los logs del repositorio de forma gráfica.

```
git checkout main
git log --graph
```

```
2dawd01@A205PC05 MINGW64 ~/Desktop/DAW 2/DESPLIEGUE/forty (feature-content)
$ git checkout main
Switched to branch 'main'

2dawd01@A205PC05 MINGW64 ~/Desktop/DAW 2/DESPLIEGUE/forty (main)
$ git log --graph
* commit acb14aaf682b198c5aee131606a22beeddf5a9c (HEAD -> main)
  Author: AlbaAlonsoGlez <albaalonsoglez@gmail.com>
  Date:   Fri Oct 17 13:18:03 2025 +0200

    .gitignore añadido
```

9. Modifica algo en el archivo generic.html , comprueba que hay cambios, y realiza otro commit. Examina los logs del repositorio de forma gráfica.

```
git add .
git commit -m "Cambios en generic.html"
git log --graph
```

```
2dawd01@A205PC05 MINGW64 ~/Desktop/DAW 2/DESPLIEGUE/forty (main)
$ git commit -m "Cambios en generic.html"
[main 70b19e7] Cambios en generic.html
 1 file changed, 1 insertion(+), 1 deletion(-)

2dawd01@A205PC05 MINGW64 ~/Desktop/DAW 2/DESPLIEGUE/forty (main)
$ git log --graph
* commit 70b19e72264f01731312f1e59d1af6aecaac9a4 (HEAD -> main)
  Author: AlbaAlonsoGlez <albaalonsoglez@gmail.com>
  Date:   Thu Oct 23 15:03:16 2025 +0200

    Cambios en generic.html

* commit acb14aaf682b198c5aee131606a22beeddf5a9c
  Author: AlbaAlonsoGlez <albaalonsoglez@gmail.com>
  Date:   Fri Oct 17 13:18:03 2025 +0200

    .gitignore añadido
```

10. Modifica algo en el fichero elements.html . Confirma los cambios, pero no hagas commit.

```
git add .
```

11. Mira las diferencias de elements.html . Los cambios no nos gustan, deshaz los cambios de elements.html . Comprueba que no hay cambios pendientes.

```
git diff elements.html  
git checkout -- elements.html
```

12. Muestra las diferencias entre dos ramas.

```
git diff main feature-content
```

13. Fusiona la rama "feature-content" con la rama principal (main). Muestra los logs del repositorio de una forma gráfica y completa.

```
git commit -m "cambios revertidos"  
git merge feature-content
```

14. Crea una nueva rama llamada " hotfix " y en ella, corrige un error crítico en el archivo "index.html ". (Por ejemplo, añade el enlace a la nueva página about.html)

```
git branch hotfix  
git checkout hotfix  
git add index.html  
git commit -m "Fix de error crítico"
```

15. Fusiona la rama "hotfix" con la rama principal y verifica el historial de commits de forma que se vean todas las ramas gráficamente. ¿Borrarías la rama hotfix ? ¿En qué caso? ¿Cómo?

```
git checkout main  
git merge hotfix
```

Si la rama no se va a utilizar más y ya está fusionada con main se puede borrar. Lo haría con el siguiente comando:

```
git branch -d hotfix
```

16. Muestra el historial de cambios limitado a los últimos 3 commits.

```
git log -3
```

```
2dawd01@A205PC05 MINGW64 ~/Desktop/DAW 2/DESPLIEGUE/forty (main)
$ git log -3
commit 692550e38f5a48747eb1e7de919b7857d18aa322 (HEAD -> main, hotfix)
Author: AlbaAlonsoGlez <albaalonsoglez@gmail.com>
Date: Thu Oct 30 13:58:45 2025 +0100

    Fix de error crítico

commit edd8ac687267aa23a6632835ae3cfcbeee1d3dcb
Merge: dd5bedb 58dbc74
Author: AlbaAlonsoGlez <albaalonsoglez@gmail.com>
Date: Thu Oct 30 13:51:51 2025 +0100

    Merge branch 'feature-content'

commit dd5bedb5bb8589aa90b7f524cf02c4104e438c7c
Author: AlbaAlonsoGlez <albaalonsoglez@gmail.com>
Date: Thu Oct 30 13:51:47 2025 +0100

    cambios revertidos
```

17. Etiqueta el commit actual como "v1.0" y muestra las etiquetas existentes.

```
git tag v1.0
git tag
```

```
2dawd01@A205PC05 MINGW64 ~/Desktop/DAW 2/DESPLIEGUE/forty (main)
$ git tag v1.0

2dawd01@A205PC05 MINGW64 ~/Desktop/DAW 2/DESPLIEGUE/forty (main)
$ git tag
v1.0
```

Trabajo en remoto

1. Sube al remoto los ficheros de tu repositorio local.

```
git remote add origin https://github.com/AlbaAlonsoGlez/forty.git
git branch -M main
git push -u origin main
```

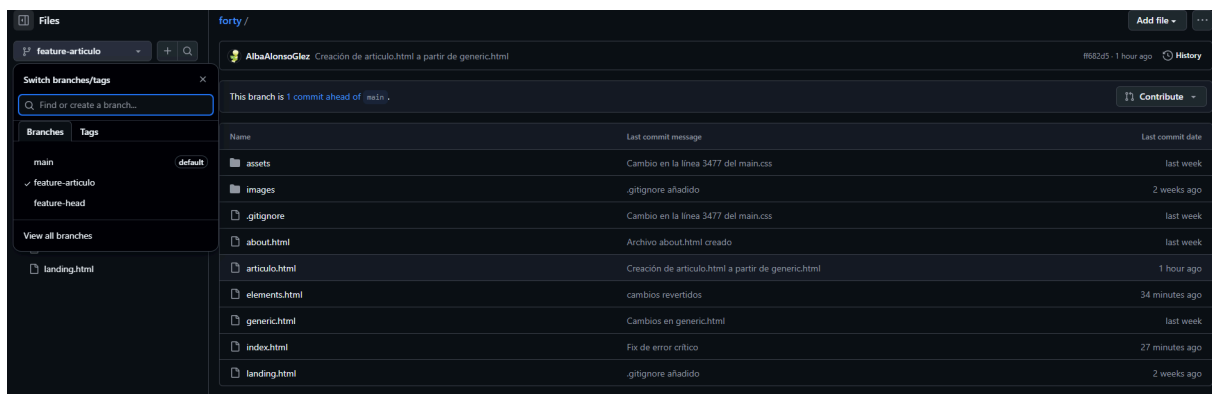
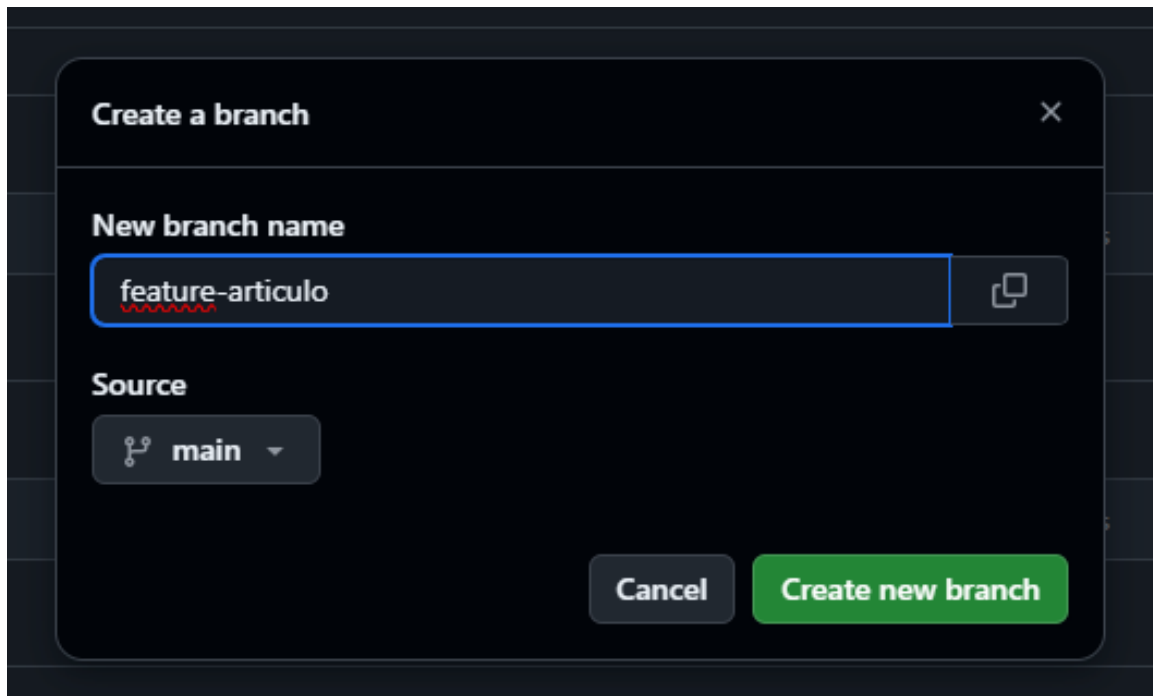
 AlbaAlonsoGlez Fix de error crítico 692550e · 16 minutes ago 7 Commits		
assets	Cambio en la línea 3477 del main.css	last week
images	.gitignore añadido	2 weeks ago
.gitignore	Cambio en la línea 3477 del main.css	last week
about.html	Archivo about.html creado	last week
elements.html	cambios revertidos	23 minutes ago
generic.html	Cambios en generic.html	last week
index.html	Fix de error crítico	16 minutes ago
landing.html	.gitignore añadido	2 weeks ago

2. En local, crea una rama 'feature-head'. Cambia el título en la sección head de index.html , borra los comentarios del head , o previos, también. Confirma y sube los cambios al remoto.

```
git branch feature-head
git checkout feature-head
```

```
2dawd01@A205PC05 MINGW64 ~/Desktop/DAW 2/DESPLIEGUE/forty (feature-head)
$ git push origin feature-head
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 16 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 331 bytes | 331.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
remote:
remote: Create a pull request for 'feature-head' on GitHub by visiting:
remote:   https://github.com/AlbaAlonsoGlez/forty/pull/new/feature-head
remote:
To https://github.com/AlbaAlonsoGlez/forty.git
* [new branch]   feature-head -> feature-head
```

3. En remoto, crea una rama 'feature-articulo'. Duplica la página generic , nómbrala como articulo.html , y añade como contenido un artículo sobre Git. Confirma los cambios y realiza un commit. Muestra los commits del repositorio tal como se ven en GitHub.

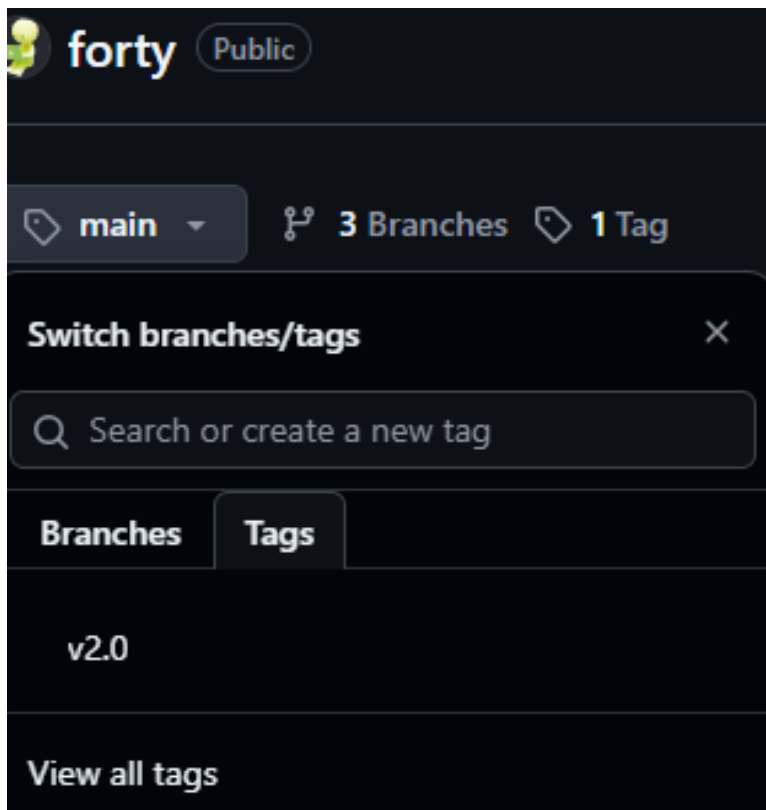
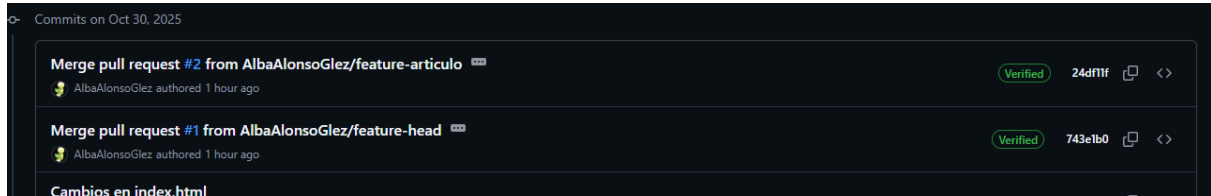


4. En el repositorio local examina los cambios. Actualiza el repositorio con el remoto. Fusiona en 'main' las dos ramas 'feature'. Crea la etiqueta 'v2.0'. Muestra los logs, commits, etiquetas y ramas actuales, en local y en remoto.

```
git status
git log --graph
git fetch origin
git pull origin main
git merge feature-main
git merge feature-articulo
git tag v2.0
```



```
git tag
git push tag v2.0
git log --graph
```



5. En tu copia local, crea una rama nueva . En la rama nueva, cambia los enlaces de la página index.html para que apunten correctamente a la nueva página articulo.html . Confirma los cambios.

```
git branch enlaces
git checkout enlaces
git add .
git commit -m "Enlaces cambiados"
```

6. Muestra los logs de forma que se vean las ramas en tu copia local.

```
git log --graph
```

7. Te gusta el resultado de los cambios. Incorpora los cambios de la rama nueva a la principal.

```
git checkout main  
git merge enlaces
```

8. Sube los cambios al remoto borrando la rama nueva , si es necesario. Comprueba primero con un comando en local, las ramas que hay en el repositorio remoto.

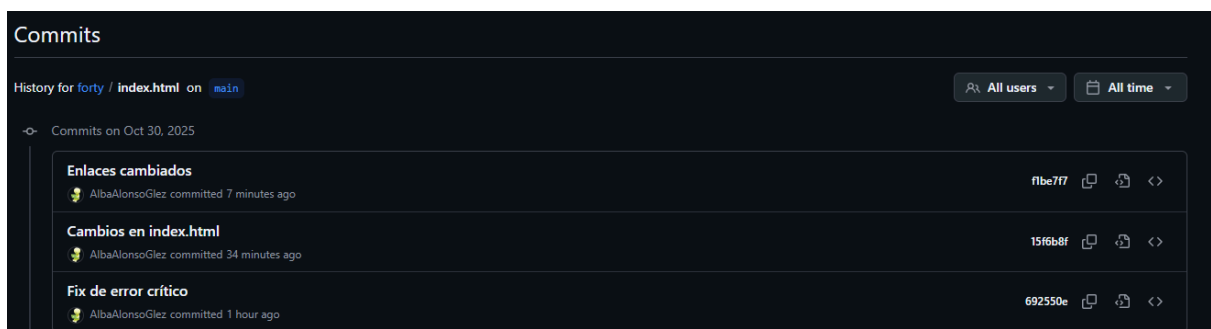
```
git push origin main  
git branch -d enlaces  
git branch -r  
git push origin --delete enlaces
```

```
2dawd01@A205PC05 MINGW64 ~/Desktop/DAW 2/DESPLIEGUE/Forty (main)  
$ git branch -r  
origin/HEAD -> origin/main  
origin/feature-articulo  
origin/feature-head  
origin/main
```

9. Muestra en local los cambios en el archivo index.html entre la versión actual y la anterior.

```
git diff index.html
```

10. En el repositorio en GitHub, navega hasta el archivo index.html y selecciona la opción "History".



Commits

History for forty / index.html on main

Search All users All time

Commits on Oct 30, 2025

Enlaces cambiados AlbaAlonsoGlez committed 7 minutes ago	f1be7f7	Commit Diff View file
Cambios en index.html AlbaAlonsoGlez committed 34 minutes ago	15f6b8f	Commit Diff View file
Fix de error crítico AlbaAlonsoGlez committed 1 hour ago	692550e	Commit Diff View file