

Ejercicios de Forty

Ejercicios de Forty

- 1 - Trabajo en local
- 2 - Trabajo en remoto

1 - Trabajo en local

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

Creamos el repositorio e inicializamos:

```
ferna@PORTATIL_Oscar MINGW64 ~  
$ mkdir forty  
cd forty  
git init  
Initialized empty Git repository in C:/Users/ferna/forty/.git/
```

```
mkdir forty  
cd forty  
git init
```

Hacemos un ls para ver los archivos que hemos agregado a la carpeta:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (master)  
$ ls  
LICENSE.txt  README.txt  assets/  elements.html  generic.html  images/  index.html  landing.html
```

```
ls
```

2. Renombra la rama master a main.

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (master)  
$ git branch -m master main  
  
ferna@PORTATIL_Oscar MINGW64 ~/forty (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.

Creamos el archivo .gitignore:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ touch .gitignore
```

```
touch .gitignore
```

Añadimos los archivos README.txt, LICENSE.txt y password.txt al archivo .gitignore escribiendo en cada archivo .gitignore:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ echo "README.txt" >> .gitignore
echo "LICENSE.txt" >> .gitignore
echo "passwords.txt" >> .gitignore
```

```
echo "README.txt" >> .gitignore
echo "LICENSE.txt" >> .gitignore
echo "passwords.txt" >> .gitignore
```

Comprobamos el contenido del archivo .gitignore:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ cat .gitignore
README.txt
LICENSE.txt
passwords.txt
```

```
cat .gitignore
```

Añadimos el cambio:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git add .gitignore
warning: in the working copy of '.gitignore', LF will be replaced by CRLF the next time Git touches it
```

```
git add .gitignore
```

Guardamos el cambio:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git commit -m "AÑADIDOS README, LICENSE Y password A .gitignore"
[main 0a4b5fb] AÑADIDOS README, LICENSE Y password A .gitignore
1 file changed, 3 insertions(+)
```

```
git commit -m "AÑADIDOS README, LICENSE Y password A .gitignore"
```

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

El archivo fue creado en el ejercicio anterior.

Añadimos texto al archivo:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ echo "HOLA" > passwords.txt
```

```
echo "HOLA" > passwords.txt
```

Comprobamos estado y vemos que es ignorado:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git status
On branch main
nothing to commit, working tree clean
```

```
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.

Creamos la rama featur-content:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git branch feature-content
```

```
git branch feature-content
```

Cambiamos a la rama feature-content:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git checkout feature-content
Switched to branch 'feature-content'

ferna@PORTATIL_Oscar MINGW64 ~/forty (feature-content)
```

```
git checkout feature-content
```

Cambiamos desde VSCode font-size en el archivo main.css:

```
3476      #banner > .inner .content p {
3477      font-size: 1.5em;
3478      font-weight: 600;
3479      letter-spacing: 0.25em;
3480      text-transform: uppercase;
3481      }
3482
```

Miramos el estado:

```

ferna@PORTATIL_Oscar MINGW64 ~/forty (feature-content)
$ git status
On branch feature-content
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:   assets/css/main.css

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

```

```
git status
```

Añadimos el cambio:

```

ferna@PORTATIL_Oscar MINGW64 ~/forty (feature-content)
$ git add .
warning: in the working copy of 'assets/css/main.css', LF will be replaced by CRLF the next time Git to
hes it

```

```
git add .
```

Guardamos el cambio:

```

ferna@PORTATIL_Oscar MINGW64 ~/forty (feature-content)
$ git commit -m "LINEA 3477 DEL ARCHIVO MAIN.CSS CAMBIADA"
[feature-content 180b81b] LINEA 3477 DEL ARCHIVO MAIN.CSS CAMBIADA
1 file changed, 1 insertion(+), 1 deletion(-)

```

```
git commit -m "LINEA 3477 DEL ARCHIVO MAIN.CSS CAMBIADA"
```

Vemos el historial con un log:

```

commit 180b81b6eacde8d31b94d2d31be8b69450d43673
Author: Oscar <fernandezdiazoscar@gmail.com>
Date:   Tue Nov 4 19:25:57 2025 +0100

    LINEA 3477 DEL ARCHIVO MAIN.CSS CAMBIADA

```

```
git log
```

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

Borramos el archivo:

```

ferna@PORTATIL_Oscar MINGW64 ~/forty (feature-content)
$ rm passwords.txt

```

```
rm passwords.txt
```

Comprobamos estado y vemos que no hay ningún cambio pendiente:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (feature-content)
$ git status
On branch feature-content
nothing to commit, working tree clean
```

```
git status
```

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

Copiamos el archivo generic.html al archivo about.html:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (feature-content)
$ cp generic.html about.html
```

```
cp generic.html about.html
```

Añadimos los cambios:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (feature-content)
$ git add .
```

```
git add .
```

Guardamos los cambios:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (feature-content)
$ git commit -m "REPOSITORIO ABOUT.HTML AGREGADO"
[feature-content 21409cf] REPOSITORIO ABOUT.HTML AGREGADO
1 file changed, 141 insertions(+)
create mode 100644 about.html
```

```
git commit -m "REPOSITORIO ABOUT.HTML AGREGADO"
```

Vemos el historial con el log:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (feature-content)
$ git log
commit 21409cfa4a30de7bdad5164bc7afcc919650c75b (HEAD -> feature-content)
Author: Oscar <fernandezdiazoscar@gmail.com>
Date: Tue Nov 4 19:37:48 2025 +0100

    REPOSITORIO ABOUT.HTML AGREGADO
```

```
git log
```

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

Cambiamos a la rama main:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (feature-content)
$ git checkout main
Switched to branch 'main'

ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
```

```
git checkout main
```

Examinamos los logs de forma gráfica:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git log --graph --oneline --decorate --all
* 21409cf (feature-content) REPOSITORIO ABOUT.HTML AGREGADO
* 180b81b LINEA 3477 DEL ARCHIVO MAIN.CSS CAMBIADA
* 0a4b5fb (HEAD -> main) AÑADIDOS README, LICENSE Y password A .gitignore
* c9c35be CONFIRMACION IGNORAR README Y LICENSE
```

```
git log --graph --oneline --decorate --all
```

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

Modificamos el archivo generic.html:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ echo añadir>>generic.html
```

```
echo añadir>>generic.html
```

Añadimos los cambios:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git add generic.html
warning: in the working copy of 'generic.html', LF will be replaced by CRLF the next time Git touches it
```

```
git add generic.html
```

Guardamos los cambios:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git commit -m "ARCHIVO GENERIC.HTML MODIFICADO"
[main 46c270c] ARCHIVO GENERIC.HTML MODIFICADO
1 file changed, 1 insertion(+), 1 deletion(-)
```

```
git commit -m "ARCHIVO GENERIC.HTML MODIFICADO"
```

Examinamos los logs de forma gráfica:


```
ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git restore --staged elements.html
```

```
git restore --staged elements.html
```

Restauramos el contenido como tal del archivo:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git restore elements.html
```

```
git restore elements.html
```

12. Muestra las diferencias entre las dos ramas:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git diff main..feature-content
diff --git a/about.html b/about.html
new file mode 100644
index 0000000..c8fd926
--- /dev/null
+++ b/about.html
@@ -0,0 +1,141 @@
+<!DOCTYPE HTML>
+<!--
+    Forty by HTML5 UP
+    html5up.net | @ajlkn
+    Free for personal and commercial use under the CCA 3.0 license (http://creativecommons.org/licenses/by/3.0/)
+-->
+<html>
+    <head>
+        <title>Generic - Forty by HTML5 UP</title>
+        <meta charset="utf-8" />
```

```
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.

Fusionamos las 2 ramas:

```
Merge made by the 'ort' strategy.
about.html | 141 +++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
assets/css/main.css | 2 +-
2 files changed, 142 insertions(+), 1 deletion(-)
create mode 100644 about.html
```

```
git merge feature-content
```

Log de forma gráfica y completa:


```

ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git log --oneline --graph --decorate --all
* 2b691d9 (HEAD -> main) wmdwaodmawldmwadadw RAMAS MAIN Y FEATURE-CONTENT FUSIONADAS
|
| * 21409cf (feature-content) REPOSITORIO ABOUT.HTML AGREGADO
| * 180b81b LINEA 3477 DEL ARCHIVO MAIN.CSS CAMBIADA
| * 46c270c ARCHIVO GENERIC.HTML MODIFICADO
|/
* 0a4b5fb AÑADIDOS README, LICENSE Y password A .gitignore
* c9c35be CONFIRMACION IGNORAR README Y LICENSE

```

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)

Creemos la nueva rama hotfix:

```

ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git branch hotfix

```

```
git branch hotfix
```

Añadimos enlace a página about.html:

```

<> index.html M X
<> index.html > html > head > a
4      html5up.net | @ajlkn
5      Free for personal and commercial use under the CCA 3.
6      -->
7      <html>
8          <head>
9              <title>Forty by HTML5 UP</title>
10             <meta charset="utf-8" />
11             <meta name="viewport" content="width=device-width
12             <link rel="stylesheet" href="assets/css/main.css"
13             <noscript><link rel="stylesheet" href="assets/css
14 |      <a href="about.html">ABOUT</a>
15          </head>
16          <body class="is-preload">
17

```

Añadimos y guardamos:

```

ferna@PORTATIL_Oscar MINGW64 ~/forty (hotfix)
$ git add .

ferna@PORTATIL_Oscar MINGW64 ~/forty (hotfix)
$ git commit -m "ERROR CRÍTICO ARREGLADO EN INDEX.HTML"
[hotfix 241b767] ERROR CRÍTICO ARREGLADO EN INDEX.HTML
1 file changed, 1 insertion(+)

```

```
git add .
```

```
git commit -m "ERROR CRÍTICO ARREGLADO EN INDEX.HTML"
```

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?

Fusionamos estas dos ramas:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git merge hotfix
Updating 2b691d9..241b767
Fast-forward
 index.html | 1 +
 1 file changed, 1 insertion(+)
```

```
git merge hotfix
```

Log de forma gráfica y completa:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git log --oneline --graph --decorate --all
* 241b767 (HEAD -> main, hotfix) ERROR CRÍTICO ARREGLADO EN INDEX.HTML
* 2b691d9 wmdwaodmawl dnmwadaw RAMAS MAIN Y FEATURE-CONTENT FUSIONADAS
| \
|  * 21409cf (feature-content) REPOSITORIO ABOUT.HTML AGREGADO
|  * 180b81b LINEA 3477 DEL ARCHIVO MAIN.CSS CAMBIADA
|  * | 46c270c ARCHIVO GENERIC.HTML MODIFICADO
| /
* 0a4b5fb AÑADIDOS README, LICENSE Y password A .gitignore
* c9c35be CONFIRMACION IGNORAR README Y LICENSE
```

```
git log --oneline --graph --decorate --all
```

La borraría porque si ya se fusionó con la principal para que voy a mantenerla.

Si esta rama solo se creó para resolver ese error y ya fue resuelto y ahora está en la main ya no es necesario.

La borraría con el siguiente comando:

```
git branch -d hotfix
```

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

```

ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git log -3
commit 241b767436b5df13cb0ac706ae66851ed32bce84 (HEAD -> main, hotfix)
Author: Oscar <fernandezdiazoscar@gmail.com>
Date: Tue Nov 4 21:05:29 2025 +0100

    ERROR CRÍTICO ARREGLADO EN INDEX.HTML

commit 2b691d9d338c68e2af0c12c5a8371fb917d0358e
Merge: 46c270c 21409cf
Author: Oscar <fernandezdiazoscar@gmail.com>
Date: Tue Nov 4 20:49:41 2025 +0100

    wmdwaodmawl dnmwadadw
    RAMAS MAIN Y FEATURE-CONTENT FUSIONADAS

commit 46c270cf54969a63cccf4350272fe03cdab92fc
Author: Oscar <fernandezdiazoscar@gmail.com>
Date: Tue Nov 4 19:57:20 2025 +0100

    ARCHIVO GENERIC.HTML MODIFICADO

```

```
git log -3
```

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

Etiquetamos el commit actual:

```

ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git tag v1.0

```

```
git tag v1.0
```

Vemos todos los git existentes:

```

ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git tag
v1.0

```

```
git tag
```

2 - Trabajo en remoto

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

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git remote add origin https://github.com/oscarfd81/forty.git
git push -u origin main
info: please complete authentication in your browser...
Enumerating objects: 107, done.
Counting objects: 100% (107/107), done.
Delta compression using up to 12 threads
Compressing objects: 100% (105/105), done.
Writing objects: 100% (107/107), 2.13 MiB | 1.28 MiB/s, done.
Total 107 (delta 13), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (13/13), done.
To https://github.com/oscarfd81/forty.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
```

```
git remote add origin https://github.com/oscarfd81/forty.git
```

```
git push -u origin main
```

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.

Creemos y nos movemos al mismo tiempo a feature-head:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git checkout -b feature-head
Switched to a new branch 'feature-head'

ferna@PORTATIL_Oscar MINGW64 ~/forty (feature-head)
```

```
git checkout -b feature-head
```

Editamos el archivo index.html con nano index.html:

```

<!-- Header -->
<!DOCTYPE HTML>
<html>
  <head>
    <title>Forty</title>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1, user-scalable=no" />
    <link rel="stylesheet" href="assets/css/main.css" />
    <noscript><link rel="stylesheet" href="assets/css/noscript.css" /></noscript>
    <a href="about.html">ABOUT</a>
  </head>
  <body class="is-preload">
    <!-- Wrapper -->
    <div id="wrapper">
      <!-- Header -->
      <header id="header" class="alt">
        <a href="index.html" class="logo"><strong>Forty</strong></a>
        <nav>
          <a href="#menu">Menu</a>
        </nav>
      </header>
    </div>
  </body>
</html>
```

Añadimos y guardamos los cambios del index.html:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (feature-head)
$ git add .

ferna@PORTATIL_Oscar MINGW64 ~/forty (feature-head)
$ git commit "ARCHIVO INDEX ENCABEZADO Y COMENTARIOS MODIFICADO"
error: pathspec 'ARCHIVO INDEX ENCABEZADO Y COMENTARIOS MODIFICADO' did not match any file(s) known to git
```

```
git add .
```

```
git commit "ARCHIVO INDEX ENCABEZADO Y COMENTARIOS MODIFICADO"
```

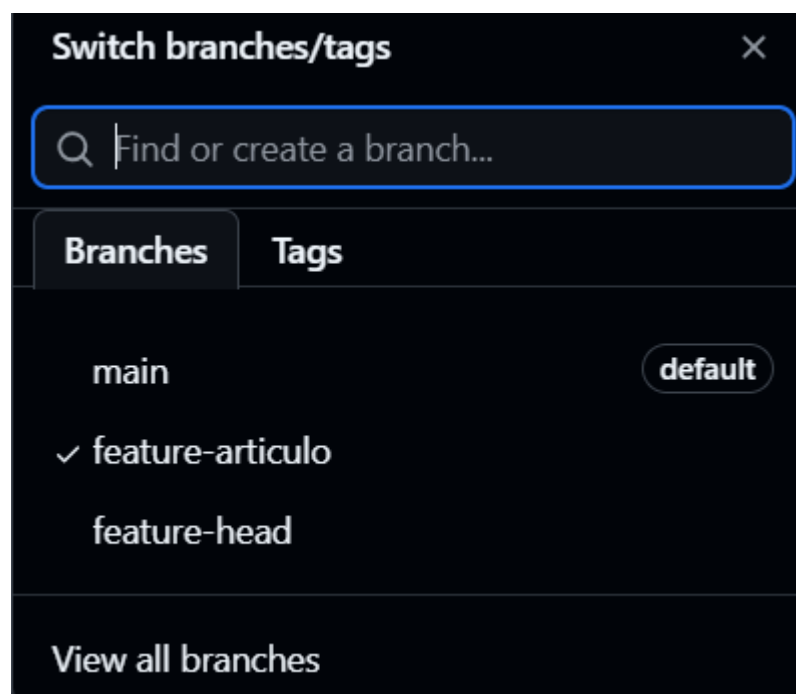
Por último hacemos un push para subir los cambios al repositorio remoto:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (feature-head)
$ git push -u origin feature-head
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'feature-head' on GitHub by visiting:
remote:   https://github.com/oscarfd81/forty/pull/new/feature-head
remote:
To https://github.com/oscarfd81/forty.git
 * [new branch]      feature-head -> feature-head
branch 'feature-head' set up to track 'origin/feature-head'.
```

```
git push -u origin feature-head
```

- 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.

Rama creada:



Commits:

assets	LINEA 3477 DEL ARCHIVO MAIN.CSS CAMBIADA	2 hours ago
images	CONFIRMACION IGNORAR README Y LICENSE	3 hours ago
.gitignore	AÑADIDOS README, LICENSE Y password A .gitignore	3 hours ago
LICENSE.txt	CONFIRMACION IGNORAR README Y LICENSE	3 hours ago
README.txt	CONFIRMACION IGNORAR README Y LICENSE	3 hours ago
about.html	REPOSITORIO ABOUT.HTML AGREGADO	2 hours ago
articulo.html	ARTICULO CREADO Y ACTUALIZADO	now
elements.html	CONFIRMACION IGNORAR README Y LICENSE	3 hours ago
generic.html	ARCHIVO GENERIC.HTML MODIFICADO	2 hours ago
index.html	ERROR CRÍTICO ARREGLADO EN INDEX.HTML	1 hour ago
landing.html	CONFIRMACION IGNORAR README Y LICENSE	3 hours ago

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.

Primero descargue las ramas remotas del Git:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (feature-head)
$ git fetch origin
remote: Enumerating objects: 7, done.
remote: Counting objects: 100% (7/7), done.
remote: Compressing objects: 100% (6/6), done.
remote: Total 6 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (6/6), 4.40 KiB | 187.00 KiB/s, done.
From https://github.com/oscarfd81/forty
* [new branch]      feature-articulo -> origin/feature-articulo
```

```
git fetch origin
```

Después me fui a la clase main:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (feature-head)
$ git checkout main
```

```
git checkout main
```

Fusionamos clase main con feature:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git merge origin/feature-head
Already up to date.

ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git merge origin/feature-articulo
Updating 241b767..147849a
Fast-forward
 articulo.html | 137 ++++++
 1 file changed, 137 insertions(+)
 create mode 100644 articulo.html
```

```
git merge origin/feature-head
git merge origin/feature-articulo
```

Creamos etiqueta:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git tag v2.0
```

```
git tag v2.0
```

Logs commits en local y remoto:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git log --oneline --graph --decorate --all
* 147849a (HEAD -> main, tag: v2.0, origin/feature-articulo) ARTICULO CREADO Y ACTUALIZADO
* d8d86b7 ARTICULO CREADO Y ACTUALIZADO
* 276a039 Add new HTML file for Generic template
* 241b767 (tag: v1.0, origin/main, origin/feature-head, origin/HEAD, hotfix, feature-head) ERROR CRÍTICO
ARREGLADO EN INDEX.HTML
* 2b691d9 wmdwaodmawldmwadadw RAMAS MAIN Y FEATURE-CONTENT FUSIONADAS
| \
| * 21409cf (feature-content) REPOSITORIO ABOUT.HTML AGREGADO
| * 180b81b LINEA 3477 DEL ARCHIVO MAIN.CSS CAMBIADA
| | 46c270c ARCHIVO GENERIC.HTML MODIFICADO
| /
* 0a4b5fb AÑADIDOS README, LICENSE Y password A .gitignore
* c9c35be CONFIRMACION IGNORAR README Y LICENSE
```

```
git log --oneline --graph --decorate --all
```

Ramas en local y remoto:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git branch -a
feature-content
feature-head
hotfix
* main
remotes/origin/HEAD -> origin/main
remotes/origin/feature-articulo
remotes/origin/feature-head
remotes/origin/main
```

```
git branch -a
```

Tags local y remoto:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git tag
v1.0
v2.0
```

```
git tag
```

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.

Creamos y nos dirigimos a la nueva rama a la que llame feature-links debido a que los demás se llama feature seguido de lo que sirve cada uno:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git checkout -b feature-links
Switched to a new branch 'feature-links'
```

```
git checkout -b feature-links
```

Actualizamos links index.html:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (feature-links)
$ nano index.html
```

```
nano index.html
```

Añadimos cambios:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (feature-links)
$ git add .
```

```
git add .
```

Guardamos cambios:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (feature-links)
$ git commit -m "ENLACES ACTUALIZADOS EN INDEX.HTML"
[feature-links febc509] ENLACES ACTUALIZADOS EN INDEX.HTML
1 file changed, 2 insertions(+), 7 deletions(-)
```

```
git commit -m "ENLACES ACTUALIZADOS EN INDEX.HTML"
```

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

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (feature-links)
$ git log --graph --oneline --decorate --all
* febc509 (HEAD -> feature-links) ENLACES ACTUALIZADOS EN INDEX.HTML
* 147849a (tag: v2.0, origin/feature-articulo, main) ARTICULO CREADO Y ACTUALIZADO
* d8d86b7 ARTICULO CREADO Y ACTUALIZADO
* 276a039 Add new HTML file for Generic template
* 241b767 (tag: v1.0, origin/main, origin/feature-head, origin/HEAD, hotfix, feature-head) ERROR CRÍTICO
ARREGLADO EN INDEX.HTML
* 2b691d9 wmdwaodmawldmwadadw RAMAS MAIN Y FEATURE-CONTENT FUSIONADAS
| \
| * 21409cf (feature-content) REPOSITORIO ABOUT.HTML AGREGADO
| * 180b81b LINEA 3477 DEL ARCHIVO MAIN.CSS CAMBIADA
| * | 46c270c ARCHIVO GENERIC.HTML MODIFICADO
| /
* 0a4b5fb AÑADIDOS README, LICENSE Y password A .gitignore
* c9c35be CONFIRMACION IGNORAR README Y LICENSE
```

```
git log --graph --oneline --decorate --all
```

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

Cambio de rama a main:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (feature-links)
$ git checkout main
Switched to branch 'main'
Your branch is ahead of 'origin/main' by 3 commits.
(use "git push" to publish your local commits)
```

```
git checkout main
```

Fusiono las ramas:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git merge feature-links
Updating 147849a..febc509
Fast-forward
 index.html | 9 ++-----
 1 file changed, 2 insertions(+), 7 deletions(-)
```

```
git merge feature-links
```

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.

Comprobamos las ramas:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git branch -a
feature-content
feature-head
feature-links
hotfix
* main
remotes/origin/HEAD -> origin/main
remotes/origin/feature-articulo
remotes/origin/feature-head
remotes/origin/main
```

Lanzamos el main a remoto:

```
ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git push origin main
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 340 bytes | 340.00 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.
To https://github.com/oscarfd81/forty.git
 241b767..febc509 main -> main
```

```
git push origin main
```

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

El HEAD apunta al commit actual, mientras que el HEAD~1 apunta al commit anterior:

```
Ferna@PORTATIL_Oscar MINGW64 ~/forty (main)
$ git diff HEAD~1 HEAD -- index.html
diff --git a/index.html b/index.html
index 7ec324a..34606fc 100644
--- a/index.html
+++ b/index.html
@@ -1,12 +1,7 @@
<!DOCTYPE HTML>
-<!--
-  Forty by HTML5 UP
-  html5up.net | @ajlkn
-  Free for personal and commercial use under the CCA 3.0 license (html5up.net/license)
--->
<html>
  <head>
-    <title>Forty by HTML5 UP</title>
+    <title>Forty</title>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1, user-scalable=no" />
    <link rel="stylesheet" href="assets/css/main.css" />
@@ -213,4 +208,4 @@
    <script src="assets/js/main.js"></script>

  </body>
-</html>
\ No newline at end of file
+</html>
```

```
git diff HEAD~1 HEAD -- 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 Nov 4, 2025

ENLACES ACTUALIZADOS EN INDEX.HTML oscarfd81 committed 16 minutes ago	febc509	📄	📁	<>
ERROR CRÍTICO ARREGLADO EN INDEX.HTML oscarfd81 committed 1 hour ago	241b767	📄	📁	<>
CONFIRMACION IGNORAR README Y LICENSE oscarfd81 committed 4 hours ago	c9c35be	📄	📁	<>

End of commit history for this file