

01 - Ejercicio de Git - Forty - local y remoto

Trabajo en local

- 1-Inicializa un nuevo repositorio Git en una carpeta llamada "forty" y agrega los archivos proporcionados en el aula virtual.
- 2-Renombra la rama master a main
- 3-Haz que los ficheros README.txt , LICENSE.txt y passwords.txt sean ignorados por el control de versiones
- 4-Crea el archivo passwords.txt. Comprueba que el control de versiones lo ignora
- 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 de la forma más gráfica posible
- 6-Elimina el archivo "passwords.txt" en la carpeta forty. Verifica el estado del repositorio. ¿Hay cambios pendientes?
- 7-Crea un nuevo archivo llamado "about.html", partiendo del archivo repositorio, haz un nuevo commit.
- 8-Cambia a la rama generic.html y agrégalo al main. Examina los logs del repositorio de forma gráfica.
- 9-Modifica algo en el archivo generic.html, comprueba que hay cambios, y realiza otro commit. Examina los logs del repositorio de forma gráfica.
- 10-Modifica algo en el fichero elements.html. Confirma los cambios, pero no hagas commit.
- 11-Mira las diferencias de elements.html. Los cambios no nos gustan, deshaz los cambios de elements.html. Comprueba que no hay cambios pendientes.
- 12-Muestra las diferencias entre dos ramas.
- 13-Fusiona la rama "feature-content" con la rama principal (main). Muestra los logs del repositorio de una forma gráfica y completa.
- 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)
- 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?
- 16-Muestra el historial de cambios limitado a los últimos 3 commits.
- 17-Etiqueta el commit actual como "v1.0" y muestra las etiquetas existentes.

Trabajo en Remoto

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

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 cambios.

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

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

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.

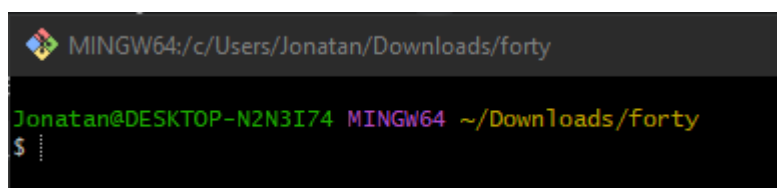
9-Muestra en local los cambios en el archivo index.html entre la versión actual y la anterior.

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

Trabajo en local

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

1.1- Abrimos el terminal desde la carpeta "forty" haciendo click derecho y dándole a Open Git Bash here



1.2- Inicializamos el repositorio git con "git init" y añadimos todos los cambios de todos los cheros no guardados en la zona de intercambio temporal.

```
$ git init
```

```
$ git add .
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty
$ git init
Initialized empty Git repository in C:/Users/Jonatan/Downloads/forty/.git/

Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (master)
$ git add .
warning: in the working copy of 'assets/css/fontawesome-all.min.css', LF will be
replaced by CRLF the next time Git touches it
```

1.3- Creo un commit inicial con todos los archivos

```
$ git commit -m "Commit inicial con todos los archivos del proyecto"
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (master)
$ git commit -m "Commit inicial con todos los los archivos del proyecto"
[master (root-commit) 6d4d50d] Commit inicial con todos los los archivos del pro
yecto
```

2-Renombra la rama master a main

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

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (master)
$ git branch -m master main

Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ !
```

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

3.1-Elimina `LICENSE.txt` y `README.txt` del repositorio (pero permanecerá en el directorio) usando `--cached` :

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

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git rm --cached LICENSE.txt
git rm --cached README.txt
rm 'LICENSE.txt'
rm 'README.txt'
```

3.2-Creamos un archivo `.gitignore` para listar los ficheros a ignorar:

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

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ echo "README.txt" >> .gitignore
echo "LICENSE.txt" >> .gitignore
echo "passwords.txt" >> .gitignore
```

3.3-Añade el nuevo archivo `.gitignore` al staging:

```
$git add .gitignore
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git add .gitignore
warning: in the working copy of '.gitignore', LF will be replaced by CRLF
the next time Git touches it
```

3.4-Confirmamos los cambios

```
$git commit -m "Ignorar README.txt, LICENSE.txt y passwords.txt"
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git commit -m "Ignorar README.txt, LICENSE.txt y passwords.txt"
[main de736de] Ignorar README.txt, LICENSE.txt y passwords.txt
3 files changed, 3 insertions(+), 95 deletions(-)
create mode 100644 .gitignore
delete mode 100644 LICENSE.txt
delete mode 100644 README.txt
```

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

4.1-Creamos el archivo passwords.txt

```
$echo "clave-secreta-123" > passwords.txt
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ echo "clave-secreta123" > passwords.txt
```

4.2- Comprobamos el estado del repositorio

```
$git status
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git status
On branch main
nothing to commit, working tree clean
```

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 de la forma más gráfica posible

5.1-Creamos la rama `feature-content` y nos cambiamos a ella

```
$git checkout -b feature-content
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git checkout -b feature-content
Switched to a new branch 'feature-content'
```

5.2-Abromos el archivo `assets/css/main.css` con visual studio code. Buscamos la **línea 3477** y cambiamos el `font-size` para que sea `1.5em`.

```
#banner > .inner .content p {
  font-size: 1.5em;
  font-weight: 600;
  letter-spacing: 0.25em;
  text-transform: uppercase;
}
```

5.3-Añadimos el archivo modificado al staging

```
$git add assets/css/main.css
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (feature-content)
$ git add assets/css/main.css
warning: in the working copy of 'assets/css/main.css', LF will be repl
LF the next time Git touches it
```

5.4-Confirmamos el cambio

```
$git commit -m "Modificar font-size a 1.5em en main.css"
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (feature-content)
$ git commit -m "Modificar font-size a 1.5em en main.css"
```

5.5-Mostramos los logs de forma grafica

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

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (feature-content)
$ git log --graph --oneline --all --decorate
* 8cbc1fa (HEAD -> feature-content) Modificar font-size a 1.5em en main.css
* de736de (main) Ignorar README.txt, LICENSE.txt y passwords.txt
* 6d4d50d Commit inicial con todos los los archivos del proyecto
```

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

6.1-Eliminamos el archivo passwords.txt

```
$rm passwords.txt
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (feature-content)
$ rm passwords.txt
```

6.2-Verificamos el estado del repositorio

```
$git status
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/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 repositorio, haz un nuevo commit.

7.1-Copiamos `generic.html` para crear `about.html`

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

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (feature-content)
$ cp generic.html about.html
```

7.1-Añadimos el nuevo archivo al repositorio

```
$git add about.html
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (feature-content)
$ git add about.html
```

7.3-Hacemos un commit

```
$git commit -m "Añadimos página about.html"
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (feature-content)
$ git commit -m "Añadimos página about.html"
[feature-content 7db9df3] Añadimos página about.html
1 file changed, 141 insertions(+)
create mode 100644 about.html
```

8-Cambia a la rama generic.html y agrégalo al main. Examina los logs del repositorio de forma gráfica.

8.1-Regresamos a la rama principal:

```
$git checkout main
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (feature-content)
$ git checkout main
Switched to branch 'main'
```

8.2-Examinamos los logs

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

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git log --graph --oneline --all --decorate
* 7db9df3 (feature-content) Añadimos página about.html
* 8cbc1fa Modificar font-size a 1.5em en main.css
* de736de (HEAD -> main) Ignorar README.txt, LICENSE.txt y passwords.txt
* 6d4d50d Commit inicial con todos los los archivos del proyecto
```

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

9.1-Modificar `generic.html` en `main`

```
<head>
  <title>Página modificada- Forty</title>
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width,
  <link rel="stylesheet" href="assets/css/main.css"
  <noscript><link rel="stylesheet" href="assets/css/
</head>
```

9.2-Comprobamos que hay cambios:

```
$git status
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git status
On branch main
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:   generic.html

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

9.3-Añadimos y confirmamos el cambio


```
$git add generic.html  
git commit -m "Modificamos título en generic.html"
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)  
$ git add generic.html  
git commit -m "Modificamos título en generic.html"  
[main 390446a] Modificamos título en generic.html  
1 file changed, 1 insertion(+), 1 deletion(-)
```

9.4-Examinamos los logs de nuevo.

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

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)  
$ git log --graph --oneline --all --decorate  
* 390446a (HEAD -> main) Modificamos título en generic.html  
| * 7db9df3 (feature-content) Añadimos página about.html  
| * 8cbc1fa Modificar font-size a 1.5em en main.css  
|/  
* de736de Ignorar README.txt, LICENSE.txt y passwords.txt  
* 6d4d50d Commit inicial con todos los los archivos del proyecto
```

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

10.1-Abrimos el fichero elements.html (En mi caso eliminare el titulo)

```
<head>  
  
  <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>  
</head>
```

10.2-Confirmamos los cambios

```
$git add elements.html
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)  
$ git add elements.html
```

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

11.1-Mira las diferencias de lo que está en staging:

```
$git diff --staged elements.html
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git diff --staged elements.html
diff --git a/elements.html b/elements.html
index 20b0e8b..723fb74 100644
--- a/elements.html
+++ b/elements.html
@@ -6,7 +6,7 @@
-->
<html>
  <head>
-    <title>Elements - Forty by HTML5 UP</title>
+    <title>Elements - Forty by HTML5 UP</title>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, in
=1, user-scalable=no" />
    <link rel="stylesheet" href="assets/css/main.css" />
```

11.2-Sacamos los cambios del staging:

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

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git restore --staged elements.html
```

11.3-Deshaz los cambios en el archivo

```
$git restore elements.html
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git restore elements.html
```

11.4-Comprobamos que no hay cambios pendientes

```
$git status
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git status
On branch main
nothing to commit, working tree clean
```

12-Muestra las diferencias entre dos ramas.

```
$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 CC BY-SA license
+-->
+<html>
+    <head>
```

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

13.1-Nos aseguramos de estar en la rama principal

```
$git checkout main
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git checkout main
Already on 'main'
```

13.2-Fusiona la rama `feature-content` en `main`

```
$git merge feature-content
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (feature-content)
Merge branch 'feature-content'
```

13.3-Mostramos los logs de forma grafica

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

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git log --graph --oneline --all --decorate
* 999ec48 (HEAD -> main) Merge branch 'feature-content'
| \
| * 7db9df3 (feature-content) Añadimos página about.html
| * 8cbc1fa Modificar font-size a 1.5em en main.css
| * 390446a Modificamos título en generic.html
| /
* de736de Ignorar README.txt, LICENSE.txt y passwords.txt
* 6d4d50d Commit inicial con todos los los archivos del proyecto
```

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)

14.1-Creamos y nos movemos a la rama hotfix

```
$git checkout -b hotfix
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git checkout -b hotfix
Switched to a new branch 'hotfix'
```

14.2-Abrimos `index.html` y añadimos el enlace a la nueva página `about.html`

```
<nav id="menu">
  <ul class="links">
    <li><a href="index.html">Home</a></li>
    <li><a href="landing.html">Landing</a></li>
    <li><a href="generic.html">Generic</a></li>
    <li><a href="elements.html">Elements</a></li>
    <li><a href="about.html">About</a></li>
  </ul>
```

14.3-Añadimos y confirmamos el cambio

```
$git add index.html  
git commit -m "Hotfix: Añadimos enlace a about.html"
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (hotfix)  
$ git add index.html  
git commit -m "Hotfix: Añadimos enlace a about.html"  
On branch hotfix  
nothing to commit, working tree clean
```

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?

15.1-Volvemos a la rama principal

```
$git checkout main
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (hotfix)  
$ git checkout main  
Switched to branch 'main'
```

15.2-Fusionamos la rama hotfix

```
$git merge hotfix
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)  
$ git merge hotfix  
Already up to date.
```

15.3-Verificamos los logs graficamente

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

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git log --graph --oneline --all --decorate
* 999ec48 (HEAD -> main, hotfix) Merge branch 'feature-content'
| \
| * 7db9df3 (feature-content) Añadimos página about.html
| * 8cbc1fa Modificar font-size a 1.5em en main.css
| * | 390446a Modificamos título en generic.html
|/
* de736de Ignorar README.txt, LICENSE.txt y passwords.txt
* 6d4d50d Commit inicial con todos los los archivos del proyecto
```

Borraría la rama hotfix en el caso en el que haya sido fusionada de manera correcta con main y que se haya verificado que la corrección funciona. Lo haría con el con el siguiente comando:

```
$git branch -d hotfix
```

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

```
$git log -n 3
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git log -n 3
commit 999ec488b89519254997633fc5d2f530bcd8a727 (HEAD -> main, hotfix)
Merge: 390446a 7db9df3
Author: jonatanyuguero <jonatanstrunilina@gmail.com>
Date: Wed Nov 5 19:38:15 2025 +0100

    Merge branch 'feature-content'

commit 390446abde104e901e622b281466fac42ff2c345
Author: jonatanyuguero <jonatanstrunilina@gmail.com>
Date: Wed Nov 5 19:21:19 2025 +0100

    Modificamos título en generic.html

commit 7db9df32d5318e4018740242a222b1fffb6ef937c (feature-content)
Author: jonatanyuguero <jonatanstrunilina@gmail.com>
Date: Wed Nov 5 19:12:56 2025 +0100

    Añadimos página about.html
```

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

17.1-Etiquetamos el commit actual (que es el "HEAD" de `main`) como "v1.0"

```
$git tag -a v1.0 -m "Versión 1.0"
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git tag -a v1.0 -m "Versión 1.0"
```

17.2-Mostramos todas las etiquetas existentes

```
$git tag
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git tag
v1.0
```

Trabajo en Remoto

1-Sube al remoto los ficheros de tu repositorio local

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git remote add origin https://github.com/jonatanyugero/forty.git
git branch -M main
git push -u origin main
Enumerating objects: 103, done.
Counting objects: 100% (103/103), done.
Delta compression using up to 16 threads
Compressing objects: 100% (102/102), done.
Writing objects: 100% (103/103), 2.13 MiB | 1.13 MiB/s, done.
Total 103 (delta 9), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (9/9), done.
To https://github.com/jonatanyugero/forty.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.

Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git push --tags
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 174 bytes | 174.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/jonatanyugero/forty.git
 * [new tag]         v1.0 -> v1.0
```

2-En local, crea una rama 'feature-head'. Cambia el título en la sección los comentarios del head de

index.html, borra head , o previos, también. Confirma y sube los cambios al remoto.

2.1-Creamos en local la rama feature-head

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

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git checkout -b feature-head
Switched to a new branch 'feature-head'
```

2.2-Abrimos `index.html` y cambiaremos el título en la línea 8: `<title>Forty by HTML5` `UP</title>` por `<title>Proyecto Forty</title>` y borraremos los comentarios HTML del `head`

```
<html>
  <head>
    <title>Proyecto 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>
  </head>
```

2.3-Confirmamos y subimos los cambios al remoto

```
$git add index.html
git commit -m "Actualizamos head y eliminamos comentarios en index.htm
l"
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (feature-head)
$ git add index.html
git commit -m "Actualizamos head y eliminamos comentarios en index.html"
[feature-head 4bebbbd] Actualizamos head y eliminamos comentarios en index.html
1 file changed, 2 insertions(+), 6 deletions(-)
```

2.4-Subimos la nueva rama al repositorio remoto

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



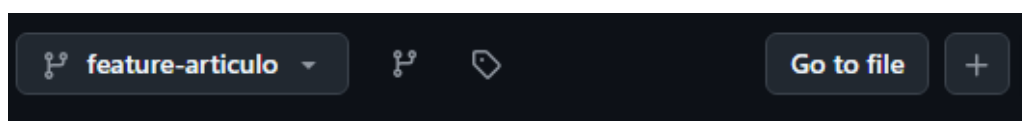
```

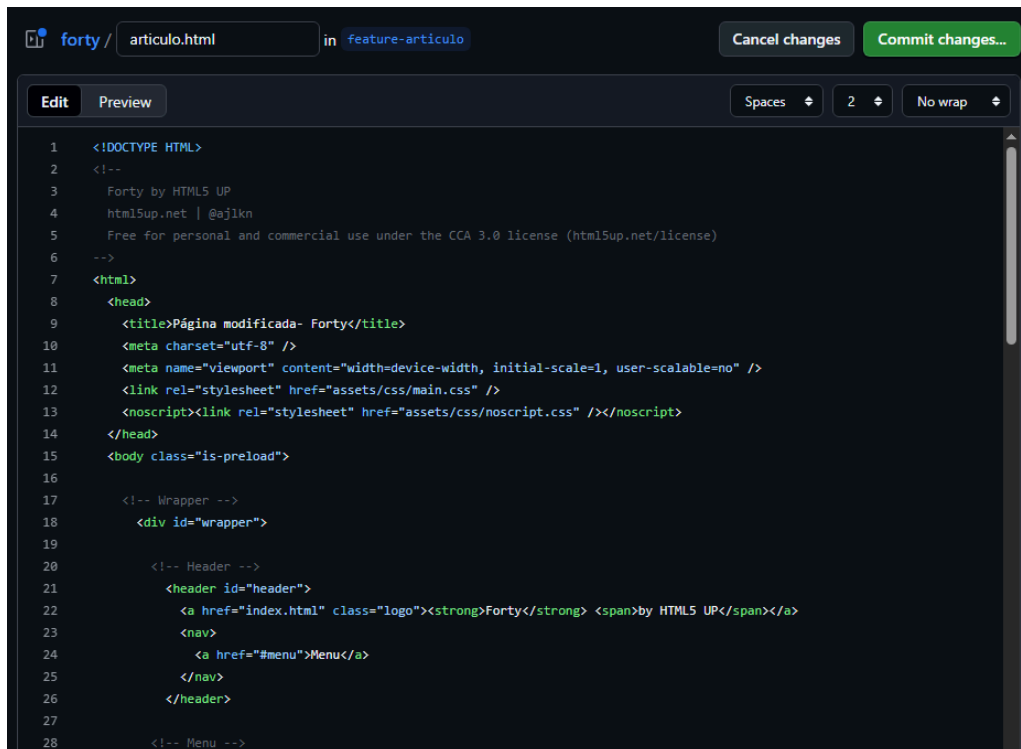
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (feature-head)
$ git push -u origin feature-head
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 16 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 401 bytes | 401.00 KiB/s, done.
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/jonatanyuguero/forty/pull/new/feature-head
remote:
To https://github.com/jonatanyuguero/forty.git
 * [new branch]      feature-head -> feature-head
branch 'feature-head' set up to track 'origin/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

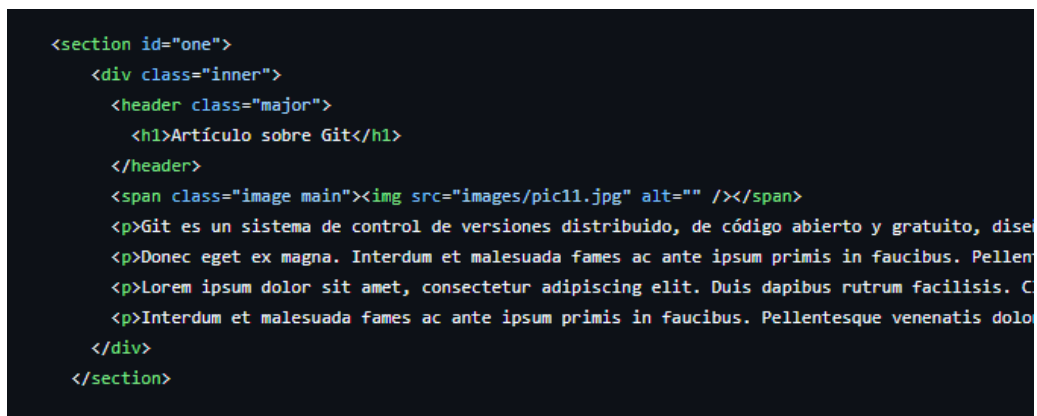
Vamos a nuestro repositorio `forty` en GitHub, Hacemos clic en el selector de ramas (que dirá `main`) y escribimos `feature-articulo`. Luego nos saldra "Create branch: feature-articulo from main" y hacemos click. Ahora copiamos todo el contenido de generic.html, despues le damos al simbolo del + y pegamos todo el contenido.





```
1 <!DOCTYPE HTML>
2 <!--
3   Forty by HTML5 UP
4   html5up.net | @ajlkn
5   Free for personal and commercial use under the CCA 3.0 license (html5up.net/license)
6 -->
7 <html>
8   <head>
9     <title>Página modificada- Forty</title>
10    <meta charset="utf-8" />
11    <meta name="viewport" content="width=device-width, initial-scale=1, user-scalable=no" />
12    <link rel="stylesheet" href="assets/css/main.css" />
13    <noscript><link rel="stylesheet" href="assets/css/noscript.css" /></noscript>
14  </head>
15  <body class="is-preload">
16
17    <!-- Wrapper -->
18    <div id="wrapper">
19
20      <!-- Header -->
21      <header id="header">
22        <a href="index.html" class="logo"><strong>Forty</strong> <span>by HTML5 UP</span></a>
23        <nav>
24          <a href="#menu">Menu</a>
25        </nav>
26      </header>
27
28      <!-- Menu -->
```

Cambiamos `<h1>Generic</h1>` por `<h1>Artículo sobre Git</h1>` y añadimos un párrafo de texto sobre Git.



```
<section id="one">
  <div class="inner">
    <header class="major">
      <h1>Artículo sobre Git</h1>
    </header>
    <span class="image main"></span>
    <p>Git es un sistema de control de versiones distribuido, de código abierto y gratuito, dise
    <p>Donec eget ex magna. Interdum et malesuada fames ac ante ipsum primis in faucibus. Pellenn
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis dapibus rutrum facilisis. C
    <p>Interdum et malesuada fames ac ante ipsum primis in faucibus. Pellentesque venenatis dolo
  </div>
</section>
```

Hacemos un commit

Commit changes

Commit message

Create articulo.html

Extended description

Add an optional extended description...

Commit Email

jonatanstrunilina@gmail.com

☒ Commit directly to the feature-articulo branch

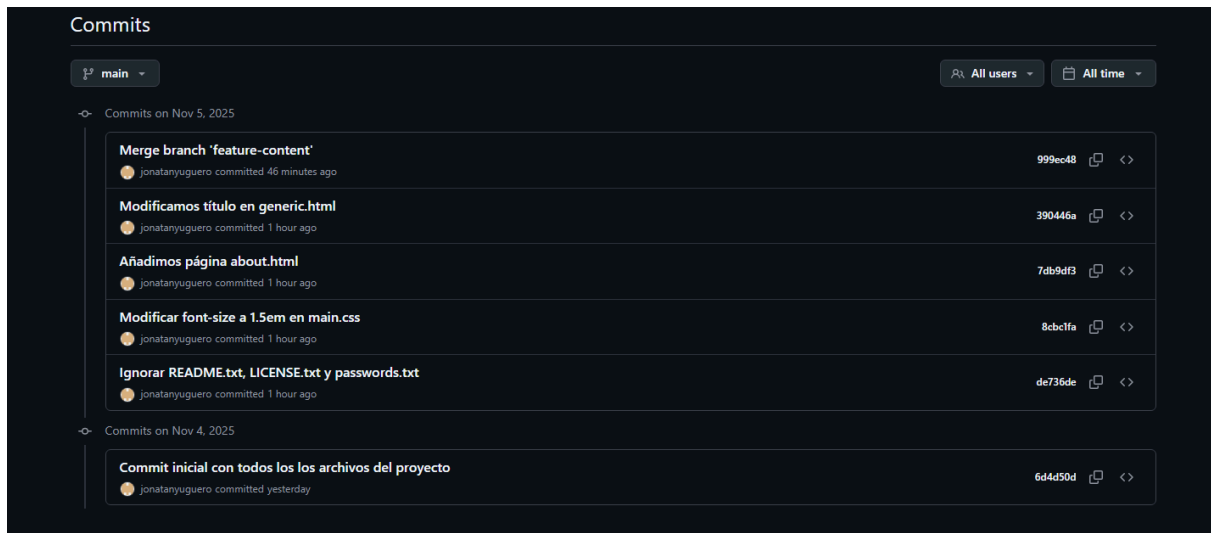
☐ Create a new branch for this commit and start a pull request [Learn more about pull requests](#)

Cancel

Commit changes

Para poder ver el historial de commits le damos donde dice en la imagen 6 commits y finalmente podremos ver todos los commits





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

4.1-Actualizamos el repositorio con el remoto

```
$git fetch origin
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (feature-head)
$ git fetch origin
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (3/3), 2.84 KiB | 3.00 KiB/s, done.
From https://github.com/jonatanyugero/forty
* [new branch]      feature-articulo -> origin/feature-articulo
```

4.2-Examinamos los cambios

```
$git branch -a
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (feature-head)
$ 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
```

4.3-Nos cambiamos a main

```
$git checkout main
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (feature-head)
$ git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
```

4.4-Fusionamos las dos ramas feature en main

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

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (feature-head)
$ git fetch origin
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (3/3), done.
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
Unpacking objects: 100% (3/3), 2.84 KiB | 3.00 KiB/s, done.
From https://github.com/jonatanyuguero/forty
* [new branch]      feature-articulo -> origin/feature-articulo
Merge remote-tracking branch 'origin/feature-articulo'
```

4.5-Creamos la etiqueta v2.0

```
$git tag -a v2.0 -m "Versión 2.0"
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git tag -a v2.0 -m "Versión 2.0"
```

4.6-Mostramos el estado en local

```
$git log --graph --oneline --all --decorate
git branch -a
git tag
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git log --graph --oneline --all --decorate
git branch -a
git tag
* 259c0fb (HEAD -> main, tag: v2.0) Merge remote-tracking branch 'origin/feature-articulo'
| \
| * 7d8d755 (origin/feature-articulo) Create articulo.html
* | 4bebbbd (origin/feature-head, feature-head) Actualizamos head y eliminamos comentarios en index.html
|/
* 999ec48 (tag: v1.0, origin/main, origin/HEAD, hotfix) Merge branch 'feature-content'
| \
| * 7db9df3 (feature-content) Añadimos página about.html
| * 8cbc1fa Modificar font-size a 1.5em en main.css
* | 390446a Modificamos título en generic.html
|/
* de736de Ignorar README.txt, LICENSE.txt y passwords.txt
* 6d4d50d Commit inicial con todos los los archivos del proyecto
feature-content
feature-head
hotfix
* main
remotes/origin/HEAD -> origin/main
remotes/origin/feature-articulo
remotes/origin/feature-head
remotes/origin/main
v1.0
v2.0
```

4.7-Subios todo a main en el remoto

```
$git push origin main
git push --tags
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git push origin main
git push --tags
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 16 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 314 bytes | 314.00 KiB/s, done.
Total 2 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/jonatanyugero/forty.git
    999ec48..259c0fb main -> main
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 173 bytes | 173.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/jonatanyugero/forty.git
    * [new tag]          v2.0 -> v2.0
```

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 cambios.

5.1-Creamos nueva rama en local

```
$git checkout -b feature-links-articulo
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git checkout -b feature-links-articulo
Switched to a new branch 'feature-links-articulo'
```

5.2-Abrimos `index.html` y añadimos el enlace a la nueva página `articulo.html` en el menú de navegación

```
<ul class="links">
  <li><a href="index.html">Home</a></li>
  <li><a href="landing.html">Landing</a></li>
  <li><a href="generic.html">Generic</a></li>
  <li><a href="elements.html">Elements</a></li>
  <li><a href="about.html">About</a></li>
  <li><a href="articulo.html">Articulo</a></li>
</ul>
```

5.3-Confirmamos los cambios

```
$git add index.html
git commit -m "Añadimos enlace a articulo.html en el menú"
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (feature-links-articulo)
$ git add index.html
git commit -m "Añadimos enlace a articulo.html en el menú"
On branch feature-links-articulo
nothing to commit, working tree clean
```

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

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

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (feature-links-articulo)
$ git log --graph --oneline --all --decorate
* 259c0fb (HEAD -> feature-links-articulo, tag: v2.0, origin/main, origin/HEAD, main) Merge remote-tracking branch 'origin/feature-articulo'
|
| * 7d8d755 (origin/feature-articulo) Create articulo.html
| * 4bebbbd (origin/feature-head, feature-head) Actualizamos head y eliminamos comentarios en index.html
|/
* 999ec48 (tag: v1.0, hotfix) Merge branch 'feature-content'
|
| * 7db9df3 (feature-content) Añadimos página about.html
| * 8cbc1fa Modificar font-size a 1.5em en main.css
| * 390446a Modificamos título en generic.html
|/
* de736de Ignorar README.txt, LICENSE.txt y passwords.txt
* 6d4d50d Commit inicial con todos los los archivos del proyecto
```

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

7.1-Volvemos a main

```
$git checkout main
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (feature-links-articulo)
$ git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
```

7.2-Fusionamos la rama de los enlaces

```
$git merge feature-links-articulo
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git merge feature-links-articulo
Already up to date.
```

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.

8.1-Comprobamos las ramas que hay en remoto


```
$git branch -r
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git branch -r
  origin/HEAD -> origin/main
  origin/feature-articulo
  origin/feature-head
  origin/main
```

8.2-Sube los cambios de main a remoto

```
$git push origin main
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git push origin main
Everything up-to-date
```

8.3-Borramos las ramas "feature" que ya están fusionadas. Primero la local que acabamos de fusionar:

```
$git branch -d feature-links-articulo
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git branch -d feature-links-articulo
Deleted branch feature-links-articulo (was 259c0fb).
```

8.4-Ahora las remotas, que ya no son necesarias

```
$git push origin --delete feature-head
git push origin --delete feature-articulo
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git push origin --delete feature-head
git push origin --delete feature-articulo
To https://github.com/jonatanyugero/forty.git
  - [deleted]          feature-head
To https://github.com/jonatanyugero/forty.git
  - [deleted]          feature-articulo
```

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

```
$git diff HEAD~1 HEAD -- index.html
```

```
Jonatan@DESKTOP-N2N3I74 MINGW64 ~/Downloads/forty (main)
$ git diff HEAD~1 HEAD -- index.html
```

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

