

# ACTIVIDAD EVALUABLE 3 - GIT Y DOCKER

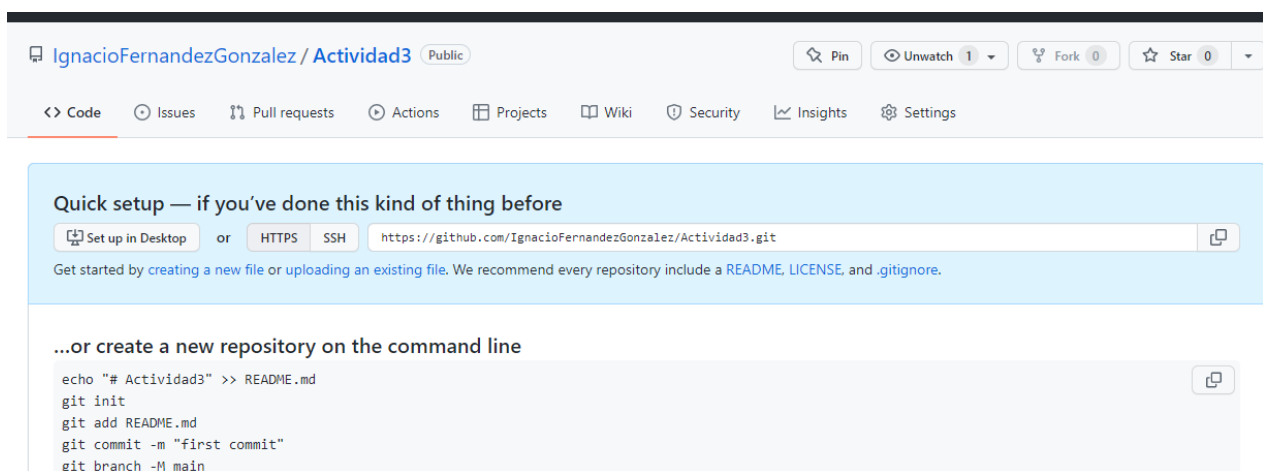
## ACTIVIDAD EVALUABLE 3 - GIT Y DOCKER

Ejercicio 1 - trabajo con imágenes

Servidor WEB

Servidor de base de datos

Para empezar a organizar la entrega de la tarea se va a crear un repositorio GitHub (  
[<https://github.com/IgnacioFernandezGonzalez/Actividad3>]:  
) para alojar las distintas soluciones de los ejercicios que componen la tarea.



A continuación he generado un directorio en mi disco local para alojar mis ficheros y poder trasladar los cambios a GitHub

## Ejercicio 1 - trabajo con imágenes

### Servidor WEB

Para empezar vamos a ejecutar un contenedor con la imagen de php:7.4-apache. Le vamos a dar de nombre web y vamos a hacer que sea accesible desde el navegador por el puerto 8080.

Lo primero que podemos hacer es comprobar si tenemos algún contenedor con ese nombre "web" ya que de ser así, no se nos permitiría crear otro, para ello usamos el comando

```
docker ps -a
```

Que nos muestra el listado de contenedores que tenemos, como se puede observar no tenemos ninguno con el nombre "web"

```
ignaciofernandez@clienteLinux:~$ docker ps -a
```

CONTAINER ID	IMAGE	STATUS	PORTS	COMMAND	CREATED	NAME
71a178510300	ejemplo1:v1	Exited (255) 2 days ago	80/tcp, 0.0.0.0:49153->8282/tcp, :::49153->8282/tcp	"/usr/sbin/apache2ct..."	2 days ago	ejemplo_v2
fec9da107175	ejemplo1:v1	Exited (255) 2 days ago	0.0.0.0:80->80/tcp, :::80->80/tcp	"/usr/sbin/apache2ct..."	2 days ago	ejemplo_v2
ced472a16180	ubuntu-python3	Exited (0) 2 days ago		"python3 hola.py"	2 days ago	nif
276eb359fc0e	ignaciofernandez/ubuntu-nodejs	Exited (0) 5 days ago		"bash"	5 days ago	kind
6780be5867e7	ubuntu	Exited (0) 5 days ago		"bash"	5 days ago	ama
96a211f1c9f7	hello-world	Exited (0) 5 days ago		"/hello"	5 days ago	agi

Ya podemos crear el contenedor con la imagen php:7.4-apache

Para ello ejecutamos el siguiente comando.

```
docker run -it -p 8000:80 --name web -v
/ignaciofernandez/ejercicio1:/var/www/html/ -d php:7.4-apache
```

```
ignaciofernandez@clienteLinux:~/ejercicio1$ docker run -it -p 8000:80 --name web -v /
ignaciofernandez/ejercicio1:/var/www/html/ -d php:7.4-apache
da6a679000eefbc00868f4f2bc832f6bdba2df3dbd98ce815c68a6f4d35b1124
```

Ahora que tenemos el contenedor corriendo, vamos a ponernos en su shell, para poder acceder a sus directorios y crear los siguientes ficheros.

Para ello ejecutamos el siguiente comando

```
docker exec -it web bash
```

Obteniendo el siguiente resultado

```
ignaciofernandez@clienteLinux:~/ejercicio1$ docker exec -it web bash
root@da6a679000ee:/var/www/html#
```

Ahora podemos hacer el siguiente paso, que es crear el fichero `index.html`, para ello ayudándonos del editor nano (el cual hay que instalar en el contenedor), editaremos el fichero según el enunciado del ejercicio.

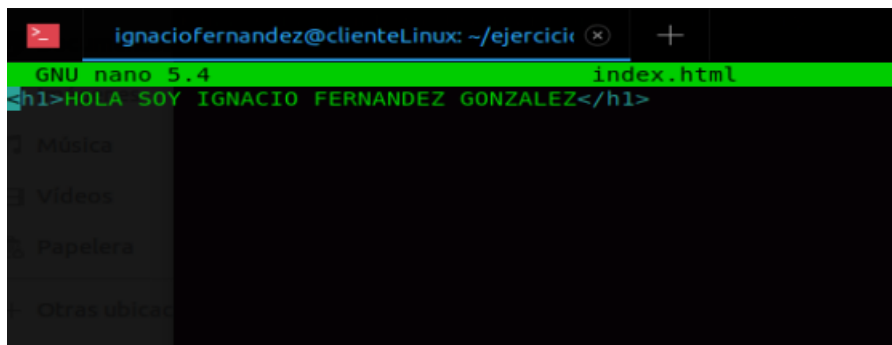
Primero actualizamos la lista de paquetes con:

```
apt-get update
```

Para a continuación instalar el editor de textos nano, con el siguiente comando

```
apt-get install nano
```

Ya podemos crear el fichero `index.html` con el código que queramos.



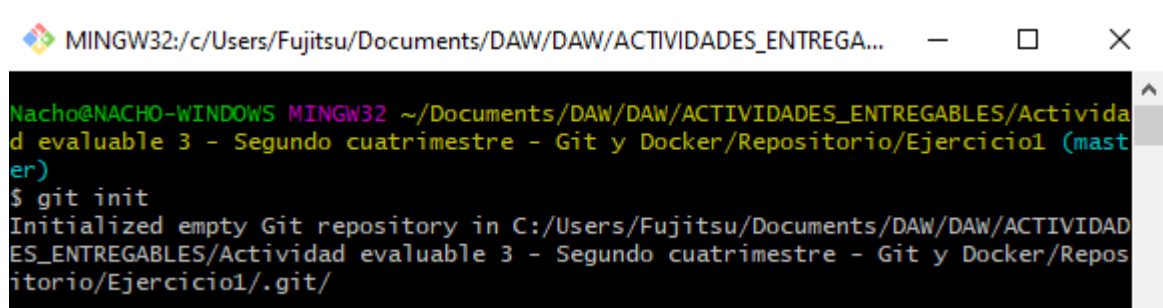
Ahora podemos visualizar este documento .html en el navegador y por el puerto 8000, como indicamos al correr el contenedor.



En este punto, y haciendo uso de git, vamos a crear una rama en nuestro repositorio para alojar el documento .md que estamos creando con la solución de este ejercicio.

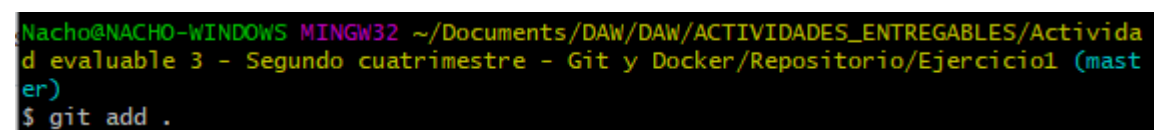
Iniciamos un repositorio en la carpeta de local donde estamos guardando los documentos, con el comando.

```
git init
```



Una vez iniciado el repositorio, añadimos los ficheros que tenemos actualmente a este repositorio con el comando

```
git add .
```



Si ejecutamos el comando

```
git status
```

observamos que tenemos cambios sin confirmar

```
$ git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   ACTIVIDAD EVALUABLE 3 - GIT Y DOCKER.assets/image-2022032419
4202659.png
    new file:   ACTIVIDAD EVALUABLE 3 - GIT Y DOCKER.assets/image-2022032419
4209941.png
    new file:   ACTIVIDAD EVALUABLE 3 - GIT Y DOCKER.assets/image-2022032419
4947547.png
    new file:   ACTIVIDAD EVALUABLE 3 - GIT Y DOCKER.assets/image-2022032419
5939526.png
    new file:   ACTIVIDAD EVALUABLE 3 - GIT Y DOCKER.assets/image-2022032420
0228407.png
    new file:   ACTIVIDAD EVALUABLE 3 - GIT Y DOCKER.assets/image-2022032420
5200677.png
    new file:   ACTIVIDAD EVALUABLE 3 - GIT Y DOCKER.assets/image-2022032420
5329652.png
    new file:   ACTIVIDAD EVALUABLE 3 - GIT Y DOCKER.md

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:   ACTIVIDAD EVALUABLE 3 - GIT Y DOCKER.md
```

El siguiente paso es realizar un commit, para confirmar estos cambios que damos por buenos.

```
git commit -m "index.htm creado"
```

```
$ git commit -m "index.html creado"
[master (root-commit) 8f7707b] index.html creado
 8 files changed, 78 insertions(+)
 create mode 100644 ACTIVIDAD EVALUABLE 3 - GIT Y DOCKER.assets/image-2022032419
4202659.png
 create mode 100644 ACTIVIDAD EVALUABLE 3 - GIT Y DOCKER.assets/image-2022032419
```

Podemos subir estos ficheros a github, con el comando

```
git push origin master
```

Ahora estamos trabajando en la rama master, pero en el ejercicio se nos dice que trabajemos en ramas diferentes para cada apartado, por tanto vamos a crear una rama "ejercicio1" y ahí trabajaremos en este ejercicio.

```
git branch ejercicio1
```

Y nos situaremos en ella con el siguiente comando

```
git checkout ejercicio1
```

Como se observa ya estamos en la rama "ejercicio1"

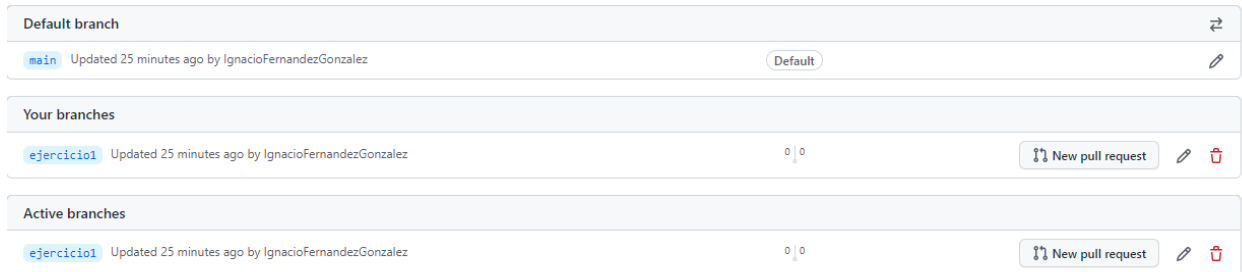
```
Nacho@NACHO-WINDOWS MINGW32 ~/Documents/DAW/DAW/ACTIVIDADES_ENTREGABLES/Actividad evaluable 3 - Segundo cuatrimestre - Git y Docker/Repositorio/Ejercicio1 (main)
$ git checkout ejercicio1
Switched to branch 'ejercicio1'

Nacho@NACHO-WINDOWS MINGW32 ~/Documents/DAW/DAW/ACTIVIDADES_ENTREGABLES/Actividad evaluable 3 - Segundo cuatrimestre - Git y Docker/Repositorio/Ejercicio1 (ejercicio1)
```

Ahora podemos subir estos ficheros de la rama a git hub, para ello, ejecutamos el comando

```
git push origin ejercicio1
```

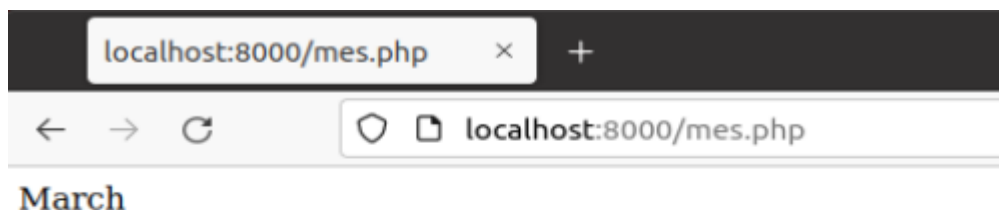
Como vemos en github, tenemos las ramas subidas (main y ejercicio1)



En este mismo apartado se nos manda incluir tambien en la misma ruta un fichero mes.php que muestre el mes actual. Para ello crearemos en el directorio /var/www/html de nuestro contenedor el fichero mes.php con el siguiente contenido.

```
<?php
setlocale(LC_TIME, 'esp');
echo date("F");
?>
```

Nos muestra lo siguiente (debería de mostrarlo en castellano, pero en mi navegador no es así, y el código es el correcto.)



En este punto vamos a comprobar el tamaño del contenedor web después de crear los ficheros index.html y mes.php, lo haremos con el comando

```
docker ps --size
```

Obteniendo como resultado 19.5MB

```
ignaciofernandez@clienteLinux:~/ejercicio1$ docker ps --size
CONTAINER ID   IMAGE             COMMAND                  CREATED        STATUS        PORTS
da6a67900ee    php:7.4-apache   "docker-php-entrypoi..." 23 hours ago   Up 23 hours   0.0.0.0:8000->80/tcp, :::8000->80/tcp
web            19.5MB (virtual 472MB)
```

Por último borraremos el contenedor.

Primero tenemos que para el contenedor con el comando

```
docker stop web
```

Y ahora podremos eliminarlo con el comando

```
docker rm web
```

```
ignaciofernandez@clienteLinux:~/ejercicio1$ docker rm web
web
```

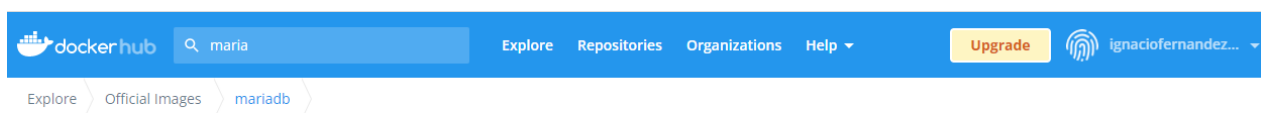
Como vemos ahora, el contenedor "web", no existe

```
ignaciofernandez@clienteLinux:~/ejercicio1$ docker ps -a
CONTAINER ID   IMAGE             COMMAND                  CREATED        STATUS        PORTS
71a178510300   ejemplo1:v1      "/usr/sbin/apache2ct..." 3 day ago     Exited (255) 2 days ago   80/tcp, 0.0.0.0:49153->8282/tcp, :::49153->8282/tcp
plo_v2        fec9da107175     ejemplo1:v1            "/usr/sbin/apache2ct..." 3 day ago     Exited (255) 2 days ago   0.0.0.0:80->80/tcp, :::80->80/tcp
ced472a16180   ubuntu-python3   "python3 hola.py"         3 day ago     Exited (0) 3 days ago
y_agnesi      276eb359fc0e     ignaciofernandez/ubuntu-nodejs "bash"         6 day ago     Exited (0) 6 days ago
_cray         6780be5867e7     ubuntu                 "bash"         6 day ago     Exited (0) 6 days ago
ing_cannon    96a211f1c9f7     hello-world            "/hello"        6 day ago     Exited (0) 6 days ago
ated_aryabhata
ianaciofernandez@clienteLinux:~/ejercicio1$
```

## Servidor de base de datos

Ahora vamos a correr un contenedor que ejecute una instancia de la imagen mariadb.

Como vemos esta imagen esta en GitHub, y es oficial



Lo primero que haremos, será dentro de nuestra rama para solucionar este ejercicio descargarnos la imagen de mariadb

```
docker pull mariadb
```

Una vez descargada podremos correr el contenedor y ponerle los datos de usuarios y contraseñas que se nos solicitan en el ejercicio, lo haremos con el comando

```
docker run --detach --name bbdd --env MARIADB_USER=invitado --env  
MARIADB_PASSWORD=invitado --env MARIADB_ROOT_PASSWORD=root --env  
MARIADB_DATABASE=prueba mariadb:latest
```

```
ignaciofernandez@clienteLinux:~/ejercicio1$ docker run --detach --name bbdd --env MAR  
IADB_USER=invitado --env MARIADB_PASSWORD=invitado --env MARIADB_ROOT_PASSWORD=root --  
-env MARIADB_DATABASE=PRUEBA mariadb:latest  
9ff396221405fd44ceb30b35b295046ceb8f5abad7c74f9af23c21929f400e93
```

Vamos a conectarnos a la base de datos para comprobar que tenemos acceso con el usuario creado y donde se ve que esta creada la base de datos prueba

Para la conexión a la base de datos primero nos pondremos en el shell del contenedor, con el comando

```
docker exec -it bbdd bash
```

Y a continuación nos conectaremos a la base de datos con el usuario creado y su contraseña, a través del siguiente comando

```
mysql --user=invitado --password=invitado
```

```
ignaciofernandez@clienteLinux:~/ejercicio1$ docker exec -it bbdd bash  
root@6elf088c1b10:/# mysql --user=invitado --password=invitado  
Welcome to the MariaDB monitor.  Commands end with ; or \g.  
Your MariaDB connection id is 3  
Server version: 10.7.3-MariaDB-1:10.7.3+maria~focal mariadb.org binary distribution  
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

Una vez conectados podemos ver si se nos ha creado la base de datos "prueba", con el comando

```
show databases;
```

Y como podemos comprobar, el resultado es satisfactorio



Para eliminar el contenedor usaremos el comando

```
docker rm bbdd
```

Al no haber parado el contenedor nos da un error y no nos deja borrarlo

```
ignaciofernandez@clienteLinux:~/ejercicio1$ docker rm bbdd  
Error response from daemon: You cannot remove a running container 6elf088c1b1058cbd2a  
53e9b3c65d7b218fbc2c80ef9b3c633de13548665f56d. Stop the container before attempting r  
emoval or force remove  
ignaciofernandez@clienteLinux:~/ejercicio1$
```

Debemos de parar el contenedor con el siguiente comando, antes de eliminarlo

```
docker stop bbdd
```

Ahora ya podemos eliminar el contenedor. Como se ve en la imagen, ya está eliminado

```

ignaciofernandez@clienteLinux:~/ejercicio1$ docker stop bbdd
bbdd
ignaciofernandez@clienteLinux:~/ejercicio1$ docker rm bbdd
bbdd
ignaciofernandez@clienteLinux:~/ejercicio1$ docker ps -a
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS      PORTS                               NAMES
71a178510300   ejemplo1:v1                          "/usr/sbin/apache2ct..." 3 days ago    Exited (255) 3 days ago             80/tcp, 0.0.0.0:49153->8282/tcp, :::49153->8282/tcp   ejemplo_v2
fec9da107175   ejemplo1:v1                          "/usr/sbin/apache2ct..." 3 days ago    Exited (255) 3 days ago             0.0.0.0:80->80/tcp, :::80->80/tcp                   ejm1
ced472a16180   ubuntu-python3                      "python3 hola.py"         3 days ago    Exited (0) 3 days ago                                     nift
y_agnesi      24-17-2022-19-2022
276eb359fc0e   ignaciofernandez/ubuntu-nodejs      "bash"                   6 days ago    Exited (0) 6 days ago                                     kind
cray          linux ~$
6780be5867e7   ubuntu                              "bash"                   6 days ago    Exited (0) 6 days ago                                     amaz
ing_cannon
96a211f1c9f7   hello-world                         "/hello"                  6 days ago    Exited (0) 6 days ago                                     agit
ated_aryabhata
ignaciofernandez@clienteLinux:~/ejercicio1$

```