

- **Inventario de archivos generados**

Se crea un espacio en GitHub para este laboratorio: - <https://github.com/fjredondo/lab5-docker>.

En él se referencia el repositorio Dockerhub personal: - <https://hub.docker.com/repositories/fjredondoa>

- **Explorando las posibilidades de Docker**

Se realiza la instalación de Docker en local (con WSL) y comprueba que se crean dos distribuciones Linux en mi equipo: docker-desktop y docker-desktop-data.

Compruebo que la instalación se ha creado con visibilidad de docker tanto como desde los terminales de Windows en PowerShell, como desde un Shell en la instalación local de Ubuntu.

```
$ docker -v
Docker version 24.0.7, build afdd53b
```

Se crean las siguientes imágenes:

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
ubuntu	latest	174c8c134b2a	11 days ago	77.9MB
python	3.11-slim	dd150e5400f1	2 weeks ago	131MB
hello-world	latest	d2c94e258dcb	7 months ago	13.3kB
docker/getting-started	latest	3e4394f6b72f	12 months ago	47MB

- **Creando imágenes interactivamente con Docker**

Este apartado permite una gran flexibilidad para trabajar. Se accede al interior de los contenedores y se pueden personalizar. La opción `-rm` permite cerrar el contenedor al salir.

```
$ docker run --rm -it python:3.11-slim bash
root@35fcd2debd56:/#
```

La ejecución de un notebook con ipython en el equipo personal es una manera eficaz de desarrollar antes de subir a entornos productivos.

```
$ docker run --rm fjredondoa/lab5-docker:v1 ipython /app/pi.ipynb 10**5
> lab5-python-results-alumno24.out
```

Finalmente la imagen se ha subido al repositorio Dockerhub. Puede ser descargada con este comando:

```
$ docker push fjredondoa/lab5-docker:
```

- **Actividades extras**

Se ha elaborado una nueva imagen mediante un dockerfile que instala todo lo necesario para estimar el valor numérico de π a través de muestreo utilizando el algoritmo de Monte-Carlo. La imagen puede descargarse del repositorio de Dockerhub:

```
$ docker push fjredondoa/lab5-pi-docker:v1
```