

Accede al documento original de Wuolah



**WUOLAH**

# Docker.pdf



Aalbii



Acceso a datos



1º Desarrollo de Aplicaciones Multiplataforma



Estudios España

**Google Gemini:**  
Plan Pro a 0€ durante 1 año.  
**Tu ventaja por ser estudiante.**

Oferta válida hasta el 9 de diciembre de 2025

[Consigue la oferta](#)

Después 21,99€/mes



Domina cualquier tema con el Aprendizaje Guiado.

Puedes explicarme como se crea un eclipse lunar completo y sus fases?

¡Claro vamos paso a paso para que lo entiendas a la perfección! 🔍🔍🔍..



# Google Gemini: Plan Pro a 0€ durante 1 año. **Tu ventaja por ser estudiante.**



**Oferta válida hasta el 9 de diciembre de 2025** **Consigue la oferta** Despu s 21,99€/mes

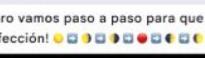
**Consigue la oferta**

Después 21,99€/mes

Domina cualquier tema con el Aprendizaje Guiado.

Puedes explicarme como se crea un eclipse lunar completo y sus fases?

¡Claro vamos paso a paso para que lo entiendas a la perfección! ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺ ☺



2:44

+

☰

Aprendizaje Guiado

x

# DOCKER

# ¡QUÉ ES?

- Docker es una plataforma que permite **crear, implementar y ejecutar** aplicaciones en contenedores
  - Los contenedores son entornos ligeros y portátiles que contienen todo lo necesario para ejecutar una aplicación, incluyendo dependencias, configuraciones y bibliotecas.
  - Docker utiliza el mismo núcleo del sistema operativo del host, esto lo hace más eficiente y rápido.

# DIFERENCIAS ENTRE DOCKER Y MÁQUINAS VIRTUALES

|                 | DOCKER  | MAQUINAS VIRTUALES                                 |
|-----------------|---|--|
| PESO            | Ligero (Mb)                                       | Pesado (Gb)  |
| INICIO          | Rápido (seg)                                      | Lento (min)  |
| AISLAMIENTO     | Comparte el núcleo del sistema operativo del host | Cada MV tiene su propio sistema operativo completo |
| USO DE RECURSOS | Eficiente, usa menos memoria y CPU                | Requiere más recursos debido al sistema operativo  |

# COMPONENTES PRINCIPALES

# DOCKER ENGINE

- Es el núcleo de docker, encargado de gestionar contenedores e imágenes.
  - Utiliza tecnologías de Linux como “namespace” y “cgroups” para garantizar el aislamiento de procesos y recursos

# IMÁGENES

- Son plantillas inmutables para crear contenedores
  - Contienen todo lo necesario para ejecutar una aplicación, como el sistema operativo base, bibliotecas y configuraciones
  - EJEMPLO: “postgres:latest” es una imagen oficial para PostgreSQL

## CONTENEDORES

- Son instancias de las imágenes en ejecución
- Cada contenedor es independiente y puede ejecutarse en paralelo con otros

## DOCKERFILE

- Es un archivo de configuración que define cómo se debe construir una imagen personalizada
- Contiene instrucciones como “FROM”, “RUN” , “COPY” ...

## DOCKER HUB

- Repositorio público donde se pueden encontrar y compartir imágenes de Docker
- URL: <https://hub.docker.com/>

# COMANDOS BÁSICOS

## GESTIÓN DE IMAGEN

- Descargar una imagen

```
docker pull <imagen>
```

```
docker pull postgres:latest
```

## GESTIÓN DE CONTENEDORES

- Crear y ejecutar un contenedor

```
docker run -d --name <nombre> -p <puerto_host>:<puerto_contenedor> <imagen>
```

```
docker run -d --name postgres-container -p 5432:5432 -e POSTGRES_PASSWORD=admin postgres
```

- Ver contenedores en ejecución
- Detener un contenedor
- Eliminar un contenedor

`docker ps`

`docker stop <id>`

`docker rm <id>`