

# Docker - Ejercicios (2) - Enunciados

Esta lección es una introducción a la instalación de aplicaciones mediante **Docker**, utilizando las imágenes proporcionadas por Bitnami. Esta lección tiene un carácter "experimental". Los dos primeros ejercicios están planteados como actividades guiadas, por lo que no se ofrece una página de soluciones. Se pueden consultar las [soluciones de los demás ejercicios de Docker \(2\)](#), aunque se recomienda intentar realizarlos primero sin recurrir a estas soluciones.

---

## Docker (2) 1 - Instalar OpenCart

En este ejercicio instalaremos OpenCart en un contenedor Docker.

Los repositorios de las imágenes de Bitnami son [Bitnami MariaDB OpenCart](#), [Bitnami Docker OpenCart](#).

1. Averigüe la dirección ip de la máquina virtual.
2. Cree una nueva red para la aplicación y la base de datos:

```
sudo docker network create opencart-network
```

3. Cree un contenedor con la base de datos MariaDB en la nueva red:

```
sudo docker run -d --name=mariadb \  
    -e ALLOW_EMPTY_PASSWORD=yes -e MARIADB_USER=bn_opencart -e MARIADB_DATABASE=bitnami_opencart \  
    --net=opencart-network bitnami/mariadb
```

**Nota:** El nombre mariadb se puede cambiar, pero tenga en cuenta que las imágenes de bitnami suponen de forma predeterminada que ese es el nombre del contenedor de la base de datos. Si utiliza otro nombre de contenedor, deberá añadir la opción de configuración correspondiente al crear el contenedor de la aplicación.

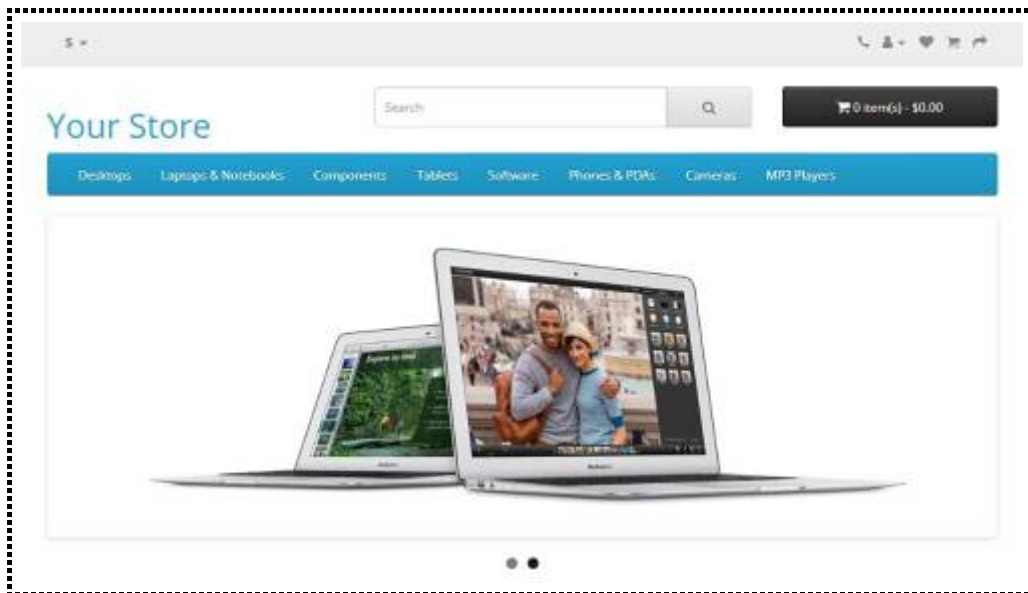
**Nota:** Algunas de las variables disponibles (y su valores predeterminados entre corchetes) son `MARIADB_EXTRA_FLAGS`, `MARIADB_ROOT_USER` [root], `MARIADB_ROOT_PASSWORD` [], `ALLOW_EMPTY_PASSWORD`, `MARIADB_USER`, `MARIADB_PASSWORD`, `MARIADB_DATABASE`.

4. Cree un contenedor con OpenCart en la misma red (sustituya AAA.BBB.CCC.DDD por la dirección ip de la máquina virtual):

```
sudo docker run -d --name=opencart \  
    -e ALLOW_EMPTY_PASSWORD=yes -e OPENCART_DATABASE_NAME=bitnami_opencart -e O \  
    PENCART_HOST=AAA.BBB.CCC.DDD \  
    -p 80:80 -p 443:443 --net=opencart-network bitnami/opencart
```

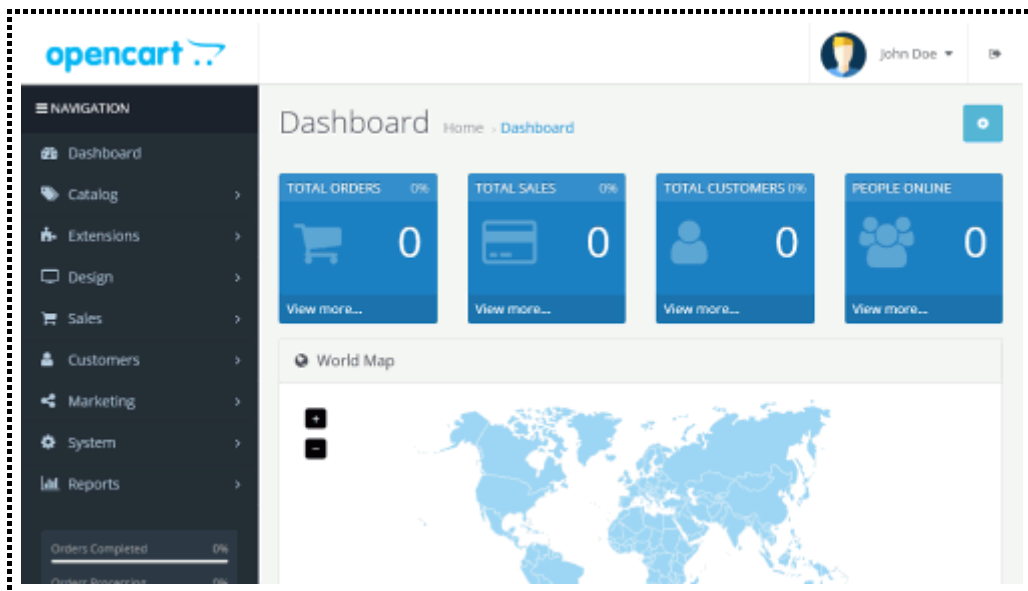
**Nota:** Las variables disponibles (y su valores predeterminados entre corchetes) son `OPENCART_USERNAME` [user], `OPENCART_PASSWORD` [bitnami1], `OPENCART_EMAIL` [user@example.com], `OPENCART_HOST`, `MARIADB_PASSWORD`, `MARIADB_HOST` [mariadb], `MARIADB_PORT_NUMBER` [3306].

5. Compruebe que puede acceder a OpenCart en el navegador escribiendo la dirección **http://AAA.BBB.CCC.DDD**, donde AAA.BBB.CCC.DDD es la dirección ip de la máquina virtual:



6. Compruebe que puede acceder como administrador a OpenCart en el navegador escribiendo la dirección **http://AAA.BBB.CCC.DDD/admin**, donde AAA.BBB.CCC.DDD es la dirección ip de la máquina virtual.

El usuario administrador predeterminado de OpenCart es **user** con contraseña **bitnami1**:



## Docker (2) 2 - Probar los comandos de Docker

En este ejercicio se trabaja con el OpenCart instalado en el ejercicio anterior.

Puede consultar la [Docker Cheat Sheet](#) para obtener información sobre los comandos básicos de Docker.

### Detener y poner en marcha

1. Modifique la tienda OpenCart eliminando alguna categoría del menú principal.
2. Detenga y reinicie el contenedor y compruebe que la tienda conserva los cambios. Los comandos que se deben utilizar son:

Para ver los contenedores existentes (en marcha o detenidos):

```
sudo docker ps -a
```

Para detener un contenedor:

```
sudo docker stop CONTENEDOR
```

Para poner en marcha un contenedor detenido:

```
sudo docker start CONTENEDOR
```

3. Detenga y borre el contenedor. Cree un nuevo contenedor y compruebe que los cambios se han perdido. Los comandos que se deben utilizar son:

Para borrar un contenedor (debe estar detenido primero):

```
sudo docker rm CONTENEDOR
```

**Nota:** La base de datos no se ha borrado, así que se podría cambiar el usuario y contraseña de la tienda para usar la antigua base de datos y recuperar la tienda antigua.

## Cambio de IP

Si cambiara la IP de la máquina virtual, OpenCart dejaría de funcionar correctamente porque en los archivos de configuración de OpenCart se guarda la URL de OpenCart. En clase esto ocurre de vez en cuando, ya que el router de aula está continuamente asignando direcciones.

1. Apague y reinicie la máquina virtual.
  - Si cambia la IP de la máquina virtual, compruebe que la tienda OpenCart ya no se ve correctamente
  - Si **no** cambia la IP de la máquina virtual, haga el ejercicio sin cambiar la IP, pero así sabrá cómo hacerlo si algún día cambia la IP.
2. Corrija los archivos de configuración de OpenCart con la nueva IP.

La forma más fácil es copiar los dos archivos de configuración de OpenCart del contenedor a la máquina virtual, modificarlos y volverlos a copiar a la imagen

```
sudo docker cp CONTENEDOR:ORIGEN DESTINO
```

Para averiguar el directorio del sistema de ficheros del contenedor en el que se encuentra opencart, puede abrir una sesión en la imagen y después buscarlo:

```
sudo docker exec -it CONTENEDOR /bin/bash
```

### ► Ayuda

3. Compruebe que la tienda OpenCart vuelve a verse correctamente.

---

## Docker (2) 3 - Instalar phpMyAdmin

En este ejercicio se pide instalar phpMyAdmin en un contenedor Docker.

El repositorio de las imágenes de Bitnami es [Bitnami phpMyAdmin OpenCart](#).

1. Cree un contenedor de phpMyadmin, al que se pueda acceder a través del puerto 8801.
2. Compruebe que puede ver la página inicial de conexión de phpMyAdmin en el navegador escribiendo la dirección **http://AAA.BBB.CCC.DDD:8801**, donde AAA.BBB.CCC.DDD es la dirección ip de la máquina virtual.

3. Averigüe la contraseña del usuario de OpenCart y entre con phpMyAdmin para ver la base de datos

---

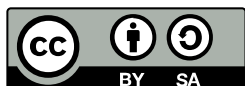
## Docker (2) 4 - Instalar dos aplicaciones (OPTATIVO)

En este ejercicio se pide instalar la misma aplicación dos veces (por ejemplo, dos instalaciones de WordPress), pero que utilicen el mismo contenedor con la base de datos, con el objetivo de hacerlos accesibles mediante las URLs **http://AAA.BBB.CCC.DDD:8802** y **http://AAA.BBB.CCC.DDD:8803**.

1. Cree una nueva red.
2. Cree un contenedor de MariaDB.
3. Cree un contenedor de phpMyAdmin al que se acceda por el puerto 8801.
4. Compruebe que puede acceder a phpMyAdmin desde el host.
5. Cree los usuarios wp1 y wp2, con base de datos única y con contraseña wp1 y wp2 y que puedan acceder desde cualquier nombre de host.
6. Cree un primer contenedor de WordPress que escuche en los puertos 8802 y 4432.
7. Compruebe que puede acceder a este WordPress desde el host.
8. Modifique el título del WordPress.
9. Cree un segundo contenedor de WordPress que escuche en los puertos 8803 y 4433.
10. Compruebe que puede acceder a este WordPress desde el host.
11. Modifique el tema o el título del WordPress para distinguirlo del anterior.

---

Última modificación de esta página: 4 de febrero de 2019



Esta página forma parte del curso [Aplicaciones web](#) por [Bartolomé Sintes Marco](#) que se distribuye bajo una [Licencia Creative Commons Reconocimiento-CompartirIgual 4.0 Internacional \(CC BY-SA 4.0\)](#).