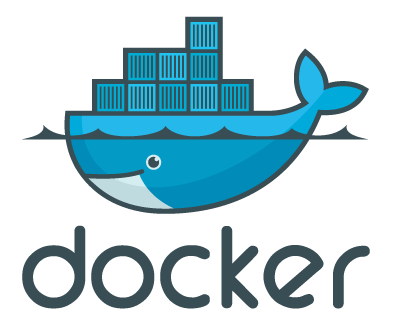
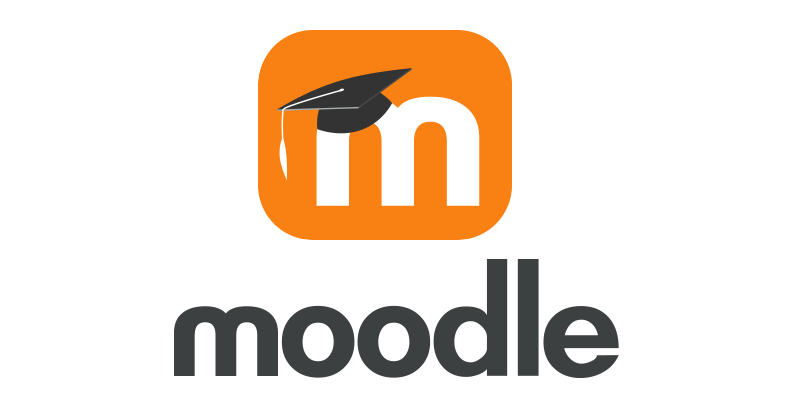
**MEMORIA PRÁCTICA PARA INSTALAR DOCKER DE MOODLE**

****

**NOMBRE:** Javier Villarreal Hinojo

**MÓDULO:** Despliegue de Aplicaciones Web

**DESPLEGAR UN CONTENEDOR CON MOODLE**

Requisitos:

* Tener docker instalado
* Tener docker compose instalado.

En primer lugar para realizar la instalación de Moodle necesitaremos un servidor PhP, y requiere a su misma vez, que este tenga componentes adicionales instalados.

Moodle a su misma vez, requerirá una base de datos ya que almacena información, normalmente, para este se utiliza mariaDB o mySQL.

**OBTENCIÓN DE LA IMAGEN:**

Una vez nos aseguremos de que hemos cumplido los requisitos, el primer paso sería obtener la imagen de Bitname docker Moodle. para ello usaremos el siguiente comando:

* **docker pull bitnami/moodle:latest**

Si por el contrario en vez de querer la última versión, se desea alguna en específica, se pondrá en la etiqueta “tag” de la siguiente forma:

* **docker pull bitnami/moodle:[TAG]**

También puede crear la imagen clonando el repositorio, cambiando al directorio que contiene el Dockerfile y ejecutando el comando “docker build”. Recuerda reemplazar los marcadores de posición APP, VERSION y OPERATING-SYSTEMpath en el siguiente comando de ejemplo con los valores correctos.

* **git clone https://github.com/bitnami/containers.git**
* **cd bitnami/APP/VERSION/OPERATING-SYSTEM**
* **docker build -t bitnami/APP:latest .**

**CÓMO HACER USO DE ESTA IMÁGEN:**

Una vez llegados a este punto en el cual ya hemos obtenido la imagen, esta imagen requerirá de una base de datos para almacenar información en ella, generalmente usaremos MariaDB o MySQL pero esto no tiene por que ser las únicas bases de datos válidas para la instalación.

Mediante docker compose, que previamente hemos instalado, ejecutaremos la aplicación mediante la cuenta principal del repositorio, el cual contiene un docker-compose.yml, y lo realizaremos de la siguiente forma:

* **curl -sSL https://raw.githubusercontent.com/bitnami/containers/main/bitnami/moodle/docker-compose.yml > docker-compose.yml**
* **docker-compose up -d**

De esta forma mediante docker-compose ya tendríamos la aplicación. Mediante estos últimos comandos y usando docker compose, nos ahorraremos tener que realizar muchísimos pasos de forma manual, ya que a través de este, de forma automática crearemos una red, un volumen para la persistencia de MariaDB, crearía un contenedor y a su misma vez un volumen necesario para la persistencia de moodle y lanzaría el contenedor.

**PERSISTIENDO EN SU APLICACIÓN:**

Si elimina el contenedor, se perderán todos sus datos y la próxima vez que ejecute la imagen, la base de datos se reiniciará.

Para que esto cambie, deberemos montar un directorio en /bitnami/moodle, de forma que si este esta vacío se inicializará en la primera ejecución. Además se debe montar un volumen de persistencia de los datos de MariaDB.

Para evitar la eliminación accidental de volúmenes, puede montar directorios de host como volúmenes de datos, lo cual se realizaría mediante docker compose, ya que lo tenemos, de la siguiente forma:

* Editamos el archivo docker-compose.yml, aplicando un pequeño cambio, de forma que quedará de la siguiente manera:



**¿POR QUÉ USAR IMÁGENES BITNAMI?¿BENEFICIOS?**

* Bitnami sigue siendo una de las fuentes más avanzadas debido a que de todos sus servicios e imágenes sacan versiones actualizadas de forma recurrente utilizando sus sistemas automatizados.
* Con las imágenes de Bitnami, las últimas funciones y correcciones de errores están disponibles lo antes posible.
* Los contenedores de Bitnami, las máquinas virtuales y las imágenes en la nube utilizan los mismos componentes y el mismo enfoque de configuración, de esta forma de facilita el cambio de formato.
* Todas las imágenes que proporcionan se basan en minideb , una imagen de contenedor minimalista basada en Debian que le brinda una imagen de contenedor base pequeña y la familiaridad de una distribución líder de Linux.
* Todas las imágenes de Bitnami disponibles en Docker Hub están firmadas con Docker Content Trust (DCT). Puede utilizar DOCKER\_CONTENT\_TRUST=1para verificar la integridad de las imágenes.
* Las imágenes de contenedores de Bitnami se publican regularmente con los últimos paquetes de distribución disponibles.