

**[MPWAR] – Entorno Web**  
**Entregable 1**

1. [1 pto] Indica la diferencia entre el uso de la instrucción CMD y ENTRYPOINT (Dockerfile).
2. [1 pto] Indica la diferencia entre el uso de la instrucción ADD y COPY (Dockerfile).
3. [2,5 ptos] Crea un contenedor con las siguientes especificaciones:
  - a. Utilizar la imagen base NGINX haciendo uso de la versión 1.19.3
  - b. Al acceder a la URL localhost:8080/index.html aparecerá el mensaje HOMEWORK 1
  - c. Persistir el fichero index.html en un volumen llamado static\_content

**Nota: Adjuntar una breve explicación de todos los pasos que has seguido para la consecución de los objetivos marcados en el enunciado y todos los comandos/ficheros utilizados**

4. [2,5 ptos] Crea una imagen docker a partir de un Dockerfile. Esta aplicación expondrá un servicio en el puerto 8080 y se deberá hacer uso de la instrucción HEALTHCHECK para comprobar si la aplicación está ofreciendo el servicio o por si el contrario existe un problema.

El healthcheck deberá parametrizarse con la siguiente configuración:

- La prueba se realizará cada 45 segundos
- Por cada prueba realizada, se esperará que la aplicación responda en menos de 5 segundos. Si tras 5 segundos no se obtiene respuesta, se considera que la prueba habrá fallado
- Ajustar el tiempo de espera de la primera prueba (Ejemplo: Si la aplicación del contenedor tarda en iniciarse 10s, configurar el parámetro a 15s)
- El número de reintentos será 2. Si fallan dos pruebas consecutivas, el contenedor deberá cambiar al estado “unhealthy”)

Resultado de la configuración:

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
d3f5426525f7	go-healthcheck	"/go/src/github.com/..."	About a minute ago	Up About a minute (healthy)	0.0.0.0:8080->8080/tcp	go-healthcheck

**Nota: Podéis utilizar el mismo Dockerfile del ejercicio anterior**

5. [3 ptos] (Caso de uso) La compañía para la que trabajáis estudia la posibilidad de incorporar a nivel interno una herramienta para la monitorización de logs. Para ello, os han encomendado la tarea de realizar una **“Proof of Concept”** (PoC). Tras evaluar diferentes productos, habéis considerado que una buena opción es la utilización del producto Elastic stack, cumple con los requisitos y necesidades de la empresa.

Tras comentarlo con el CTO a última hora de la tarde, os ha solicitado que preparéis una presentación para mañana a primera hora. Dado el escaso margen para montar la demostración, la opción más ágil y rápida es utilizar una solución basada en contenedores donde levantaréis el motor de indexación (ElasticSearch) y la herramienta de visualización (Kibana).

Rellena el siguiente fichero docker-compose para que podáis hacer la demostración al CTO.

```
version: '3.6'

services:

  elasticsearch:
    # Utilizar la imagen de elasticsearch v7.9.3
    ...

    # Asignar un nombre al contenedor
    ...

    # Define las siguientes variables de entorno:
    # discovery.type=single-node
    ...

    # Emplazar el contenedor a la red de elastic
    ...
```

```

# Mapea el Puerto externo 9200 al puerto interno del contenedor 9200
# Idem para el puerto 9300
...

kibana:
  # Utilizar la imagen kibana v7.9.3
  ...
  # Asignar un nombre al contenedor
  ...
  # Emplazar el contenedor a la red de elastic
  ...

  # Define las siguientes variables de entorno:
  # ELASTICSEARCH_HOST=elasticsearch
  # ELASTICSEARCH_PORT=9200
  ...

  # Mapear el puerto externo 5601 al puerto interno 5601
  ...

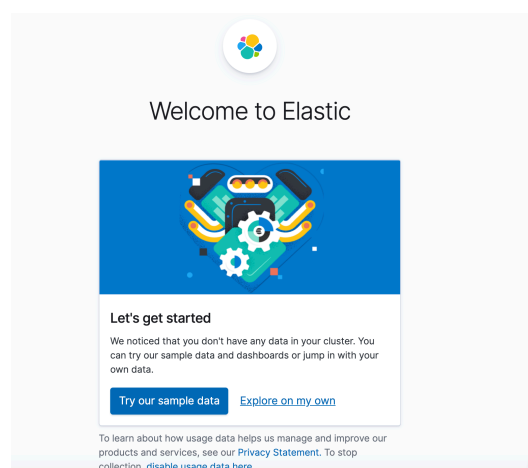
  # El contenedor Kibana debe esperar a la disponibilidad del servicio elasticsearch
  ...

# Definir la red elastic (bridge)
...

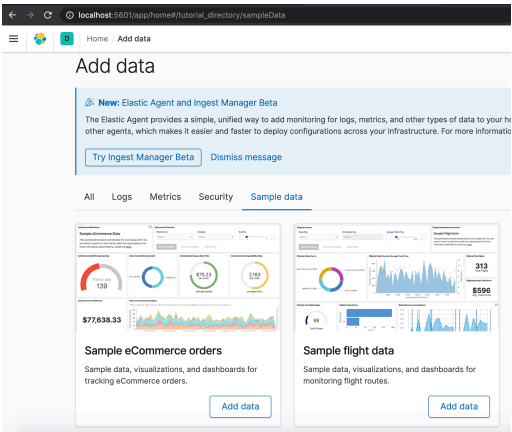
```

Para comprobar la correcta configuración deberemos acceder al portal de Kibana (<http://localhost:5601>):

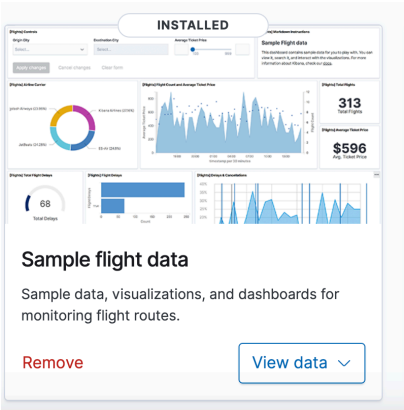
Escoger la opción “**Try our simple data**”:



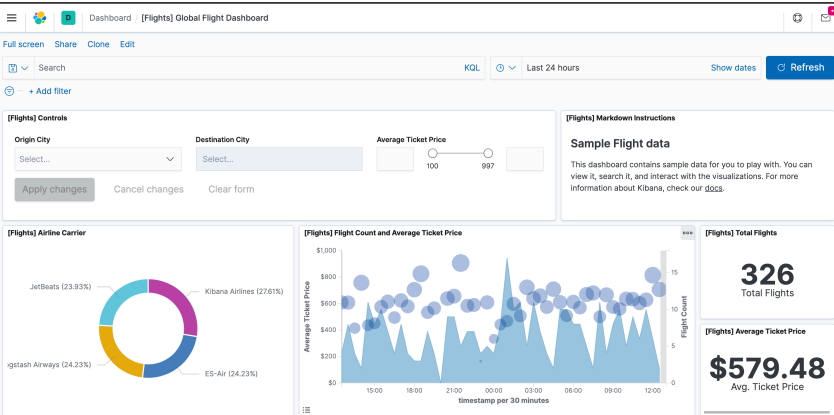
Seleccionar la opción **“Add data”**:



Y una vez se hayan cargado los datos, seleccionar **View data > Dashboard**:



Resultado:



**IMPORTANTE:** Crear un proyecto en vuestro repositorio de código (Github, Gitlab, ...) llamado **docker-exercises**. Dentro de ese proyecto, crear una carpeta para la primera entrega, por ejemplo **hw-01**.

Dentro de esa carpeta, crear un fichero por pregunta `answer_exercise_1.md`. Si necesitáis subir ficheros, crear una carpeta para ese ejercicio con todo el contenido. Si necesitáis hacer algún tipo de aclaración, hacerlo en el fichero `README.md`.

En el enlace de entrega del campus, entregar un nota (`notes_hw_1.txt`) con el enlace del repositorio y el código del commit.