UNIT 1.4 GRADED ASSIGNMENT
DOCKER

Group members

Ifra Saleem (2303.khi.deg.003)
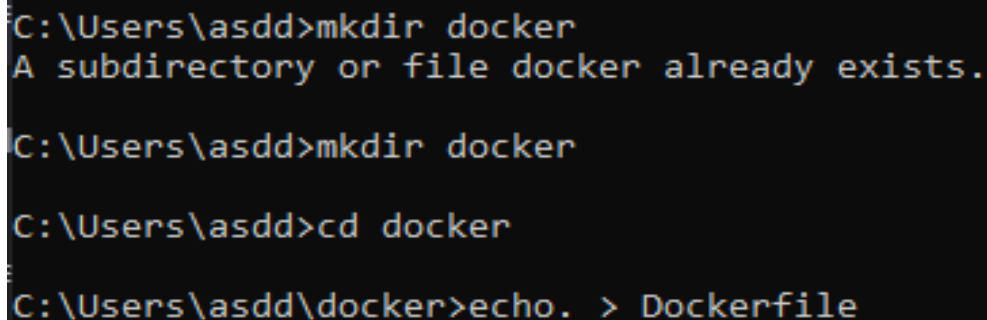Muhammad Khan (2303.khi.deg.027)

UNIT 1.4 GRADED ASSIGNMENT

Task:

1. Build an image based on Jupyter Notebook (jupyter/minimal-notebook) with Pandas installed (pip install pandas).
2. Create a container from this image and use the NOTEBOOK_ARGS=--port=8889 environment variable to change the port Jupyter is exposed on.
3. Verify you can access it on port 8889 and that Pandas is installed (type import pandas in a notebook).

Solution:

1. **Build an image based on Jupyter Notebook (jupyter/minimal-notebook) with Pandas installed (pip install pandas).**

To build an image based on jupyter notebook I created a new folder in my local directory and then in that folder I created a file named **Dockerfile** (A Dockerfile is a text document which contains all the information and commands which are required to create an image).


```
C:\Users\asdd>mkdir docker
A subdirectory or file docker already exists.

C:\Users\asdd>mkdir docker

C:\Users\asdd>cd docker

C:\Users\asdd\docker>echo. > Dockerfile
```

In the Dockerfile I wrote only two commands:

- **FROM jupyter/minimal-notebook**
- **RUN pip install pandas**

Using FROM keyword we can specify the name of the parent image which we are using to build our own image. The parent image is basically the base image and FROM keyword specifies the architecture that we want to use to build an image.

Then in the second line I am using RUN keyword which is used to run a command in the new image and here I am using RUN to install pandas library in the image.

Then I use the following command to build a docker image based on the Dockerfile.

docker build -t image-pandas .

- − **Docker build** is used to build an image.
- − **-t** is used to give the title to the image.
- − **Image-pandas** is the title of the image which we are building using the above command.
- − '**.**' represents the build context which is current directory where dockerfile is located.

2. **Create a container from this image and use the NOTEBOOK_ARGS=--port=8889 environment variable to change the port Jupyter is exposed on.**

After building the image I created a container from that image using the following command:

docker run -p 8889:8888 -e "NOTEBOOK_ARGS=--port=8889" image-pandas

- − docker run starts a docker container.
- − -p 8889:8888 maps the port 8888 in the container to port 8889 in the host machine.

- -e "NOTEBOOK_ARGS=--port=8889" sets the environment variable which indicates that the jupyter Notebook server will be accessed on port 8889.
- Image-pandas is the name of the image which we build earlier.

```
C:\Users\asdd\docker>docker run -p 8889:8888 -e "NOTEBOOK_ARGS=--port=8889" image-pandas
docker: Error response from daemon: open \\.\pipe\docker_engine_linux: The system cannot find the file specified.
See 'docker run --help'.

C:\Users\asdd\docker>docker run -p 8889:8888 -e "NOTEBOOK_ARGS=--port=8889" image-pandas
Entered start.sh with args: jupyter lab --port=8889
Executing the command: jupyter lab --port=8889
[I 2023-04-09 10:40:05.365 ServerApp] Package jupyterlab took 0.0001s to import
[I 2023-04-09 10:40:05.379 ServerApp] Package jupyter_server_fileid took 0.0122s to import
[I 2023-04-09 10:40:05.410 ServerApp] Package jupyter_server_terminals took 0.0287s to import
[I 2023-04-09 10:40:05.611 ServerApp] Package jupyter_server_ydoc took 0.1995s to import
[I 2023-04-09 10:40:05.613 ServerApp] Package nbclassic took 0.0001s to import
[W 2023-04-09 10:40:05.622 ServerApp] A `_jupyter_server_extension_points` function was not found in nbclassic. Instead
 a `_jupyter_server_extension_paths` function was found and will be used for now. This function name will be deprecated
in future releases of Jupyter Server.
[I 2023-04-09 10:40:05.624 ServerApp] Package notebook_shim took 0.0001s to import
[W 2023-04-09 10:40:05.624 ServerApp] A `_jupyter_server_extension_points` function was not found in notebook_shim. Ins
ead, a `_jupyter_server_extension_paths` function was found and will be used for now. This function name will be deprec
ted in future releases of Jupyter Server.
[I 2023-04-09 10:40:05.642 ServerApp] jupyter_server_fileid | extension was successfully linked.
[I 2023-04-09 10:40:05.657 ServerApp] jupyter_server_terminals | extension was successfully linked.
[I 2023-04-09 10:40:05.675 ServerApp] jupyter_server_ydoc | extension was successfully linked.
[I 2023-04-09 10:40:05.703 ServerApp] jupyterlab | extension was successfully linked.
[W 2023-04-09 10:40:05.715 NotebookApp] 'ip' has moved from NotebookApp to ServerApp. This config will be passed to Ser
erApp. Be sure to update your config before our next release.
[W 2023-04-09 10:40:05.716 NotebookApp] 'ip' has moved from NotebookApp to ServerApp. This config will be passed to Ser
erApp. Be sure to update your config before our next release.
[I 2023-04-09 10:40:05.724 ServerApp] nbclassic | extension was successfully linked.
```

3. **Verify you can access it on port 8889 and that Pandas is installed (type import pandas in a notebook).**

Where we start the container, we got a link through which we access jupyter notebook on the localhost on the port 8889 and here we created a new notebook and import pandas to verify that pandas library is installed or not. We didn't get an error so it means that pandas is installed.