**NAME : MAVIA ALAM KHAN (2303.KHI.DEG.017)**

**PARTNER: MUHAMMAD HUSSAM (2303.KHI.DEG.020)**

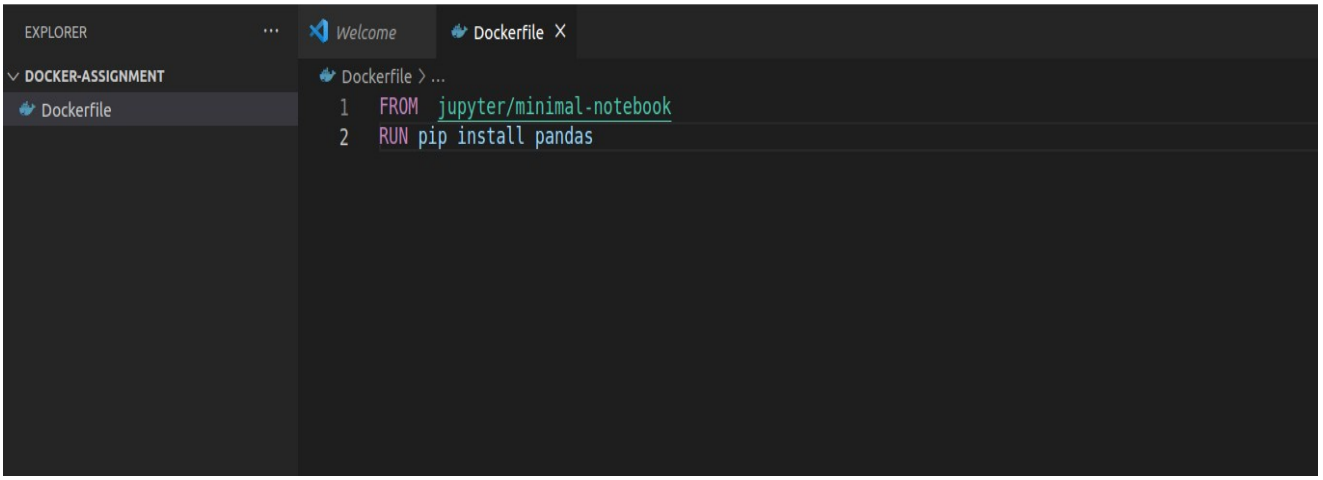
---

# ASSIGNMENT_DAY-4

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

## SOLUTION:

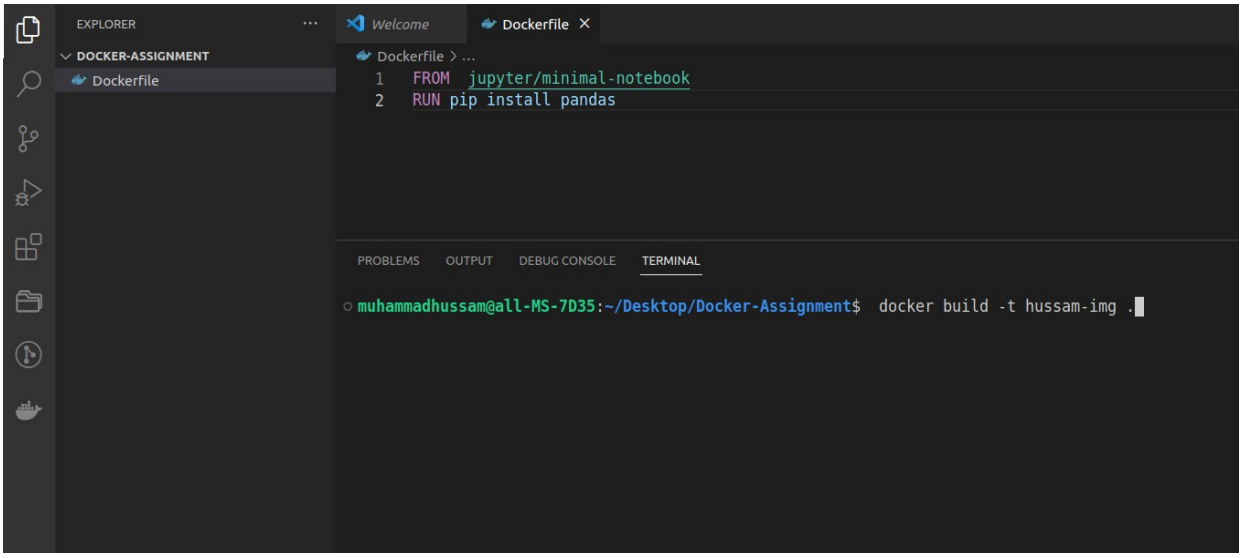
**STEP 1:**




First we have to create docker file which contains a set of instructions to create an image from a base image. Here we used **jupyter/minimal-**

**notebook** as base image,then run instruction is used to execute pip install pandas, This command installs the Pandas library into the container. Once the image is built, you can then use it to create a container that has Pandas pre-installed.

## STEP 2:




We will use the command 'docker build' to build a Docker image with the name 'hussam-img' using the Dockerfile located in the current directory as:

**docker build -t hussam-img .**

After executing  above command , Docker will start building the image based on the instructions in the Dockerfile located in the current directory:

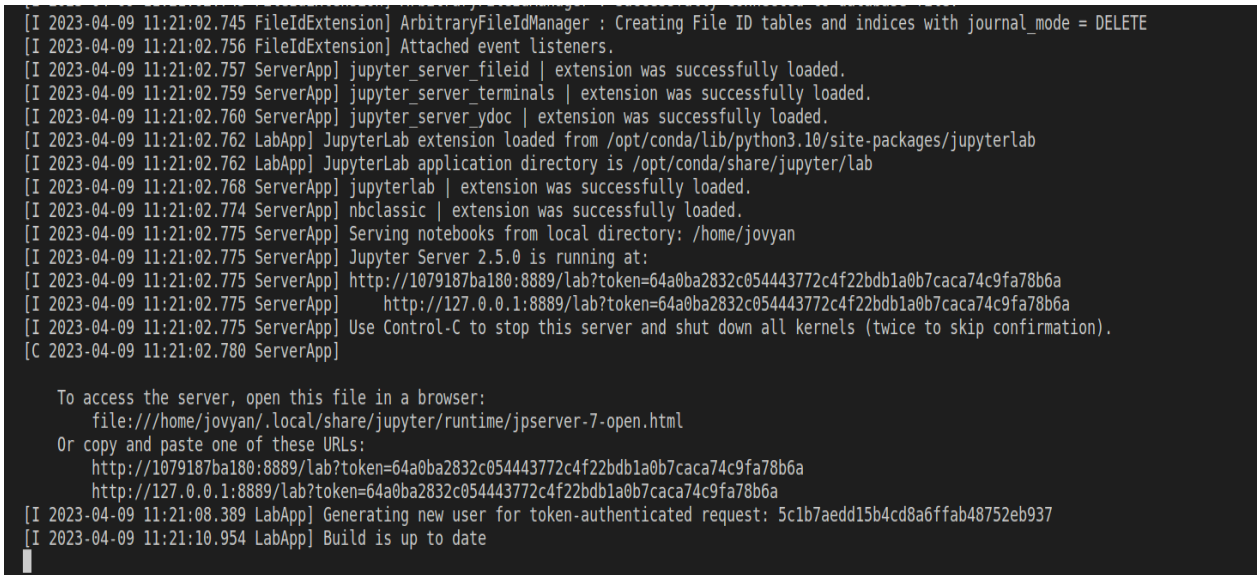
```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

muhammadhussam@all-MS-7D35:~/Desktop/Docker-Assignment$  docker build -t hussam-img .
[+] Building 0.1s (6/6) FINISHED
 => [internal] load build definition from Dockerfile
 => => transferring dockerfile: 90B
 => [internal] load .dockerignore
 => => transferring context: 2B
 => [internal] load metadata for docker.io/jupyter/minimal-notebook:latest
 => [1/2] FROM docker.io/jupyter/minimal-notebook
 => CACHED [2/2] RUN pip install pandas
 => exporting to image
 => => exporting layers
 => => writing image sha256:159375fed26831aae046a2f99a59655b501ec48ebc70c4677c2cbdfb67890e03
 => => naming to docker.io/library/hussam-img
muhammadhussam@all-MS-7D35:~/Desktop/Docker-Assignment$
```

# STEP 3:

Now to run a container based on the image we created above , we use the following command

**docker run -p 8889:8889 -e NOTEBOOK_ARGS=--port=8889  hussam-img**

```
muhammadhussam@all-MS-7D35:~/Desktop/Docker-Assignment$ docker run -p 8889:8889 -e NOTEBOOK_ARGS=--port=8889  hussam-img
Entered start.sh with args: jupyter lab --port=8889
Executing the command: jupyter lab --port=8889
[I 2023-04-09 11:21:02.422 ServerApp] Package jupyterlab took 0.0000s to import
[I 2023-04-09 11:21:02.424 ServerApp] Package jupyter_server_fileid took 0.0020s to import
[I 2023-04-09 11:21:02.428 ServerApp] Package jupyter_server_terminals took 0.0039s to import
[I 2023-04-09 11:21:02.452 ServerApp] Package jupyter_server_ydoc took 0.0233s to import
[I 2023-04-09 11:21:02.453 ServerApp] Package nbclassic took 0.0000s to import
[W 2023-04-09 11:21:02.455 ServerApp] A `_jupyter_server_extension_points` function was not found in nbclassic. Instead, a `_jupyter_server_ext
ension_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 11:21:02.455 ServerApp] Package notebook_shim took 0.0000s to import
[W 2023-04-09 11:21:02.456 ServerApp] A `_jupyter_server_extension_points` function was not found in notebook_shim. 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 11:21:02.459 ServerApp] jupyter_server_fileid | extension was successfully linked.
[I 2023-04-09 11:21:02.462 ServerApp] jupyter_server_terminals | extension was successfully linked.
[I 2023-04-09 11:21:02.466 ServerApp] jupyter_server_ydoc | extension was successfully linked.
[I 2023-04-09 11:21:02.474 ServerApp] jupyterlab | extension was successfully linked.
[W 2023-04-09 11:21:02.479 NotebookApp] 'ip' has moved from NotebookApp to ServerApp. This config will be passed to ServerApp. Be sure to updat
e your config before our next release.
[W 2023-04-09 11:21:02.479 NotebookApp] 'ip' has moved from NotebookApp to ServerApp. This config will be passed to ServerApp. Be sure to updat
e your config before our next release.
[I 2023-04-09 11:21:02.483 ServerApp] nbclassic | extension was successfully linked.
[I 2023-04-09 11:21:02.486 ServerApp] Writing Jupyter server cookie secret to /home/jovyan/.local/share/jupyter/runtime/jupyter_cookie_secret
[I 2023-04-09 11:21:02.653 ServerApp] notebook_shim | extension was successfully linked.
[I 2023-04-09 11:21:02.744 ServerApp] notebook_shim | extension was successfully loaded.
[I 2023-04-09 11:21:02.745 FileIdExtension] Configured File ID manager: ArbitraryFileIdManager
[I 2023-04-09 11:21:02.745 FileIdExtension] ArbitraryFileIdManager : Configured root dir: /home/jovyan
[I 2023-04-09 11:21:02.745 FileIdExtension] ArbitraryFileIdManager : Configured database path: /home/jovyan/.local/share/jupyter/file_id_manage
r.db
[I 2023-04-09 11:21:02.745 FileIdExtension] ArbitraryFileIdManager : Successfully connected to database file.
[I 2023-04-09 11:21:02.745 FileIdExtension] ArbitraryFileIdManager : Creating File ID tables and indices with journal_mode = DELETE
```




This command starts a new Docker container based on the **hussam-img** image. The container is configured to map port 8889 on the host machine to port 8889 inside the container, allowing access to the Jupyter Notebook server from the host machine's web browser. An environment variable called **NOTEBOOK_ARGS** is set to **--port=8889** to specify that the Jupyter Notebook should listen on port 8889 . After the command runs successfully, you should be able to access the Jupyter Notebook interface by opening a web browser and navigating to:

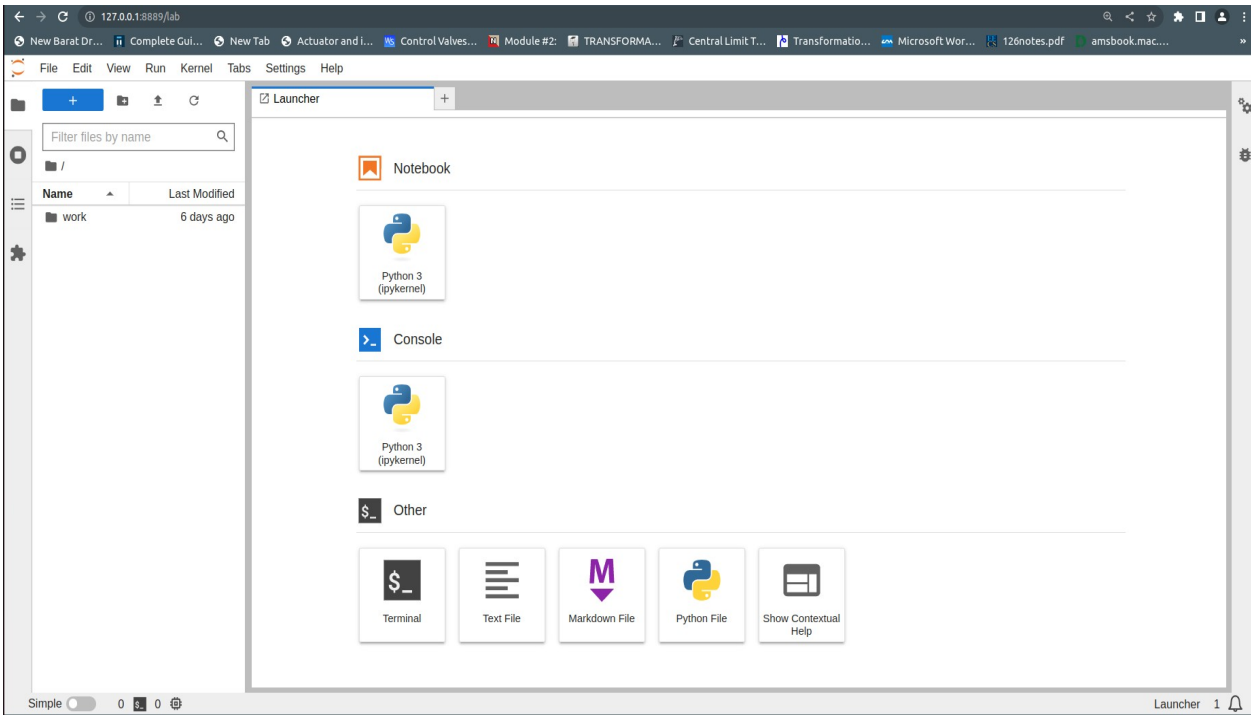
**http://127.0.0.1:8889/lab?token=64a0ba2832c054443772c4f22bdb1a0b7caca74c9fa78b6a**

## STEP 4 (verification if pandas installed in it or not):

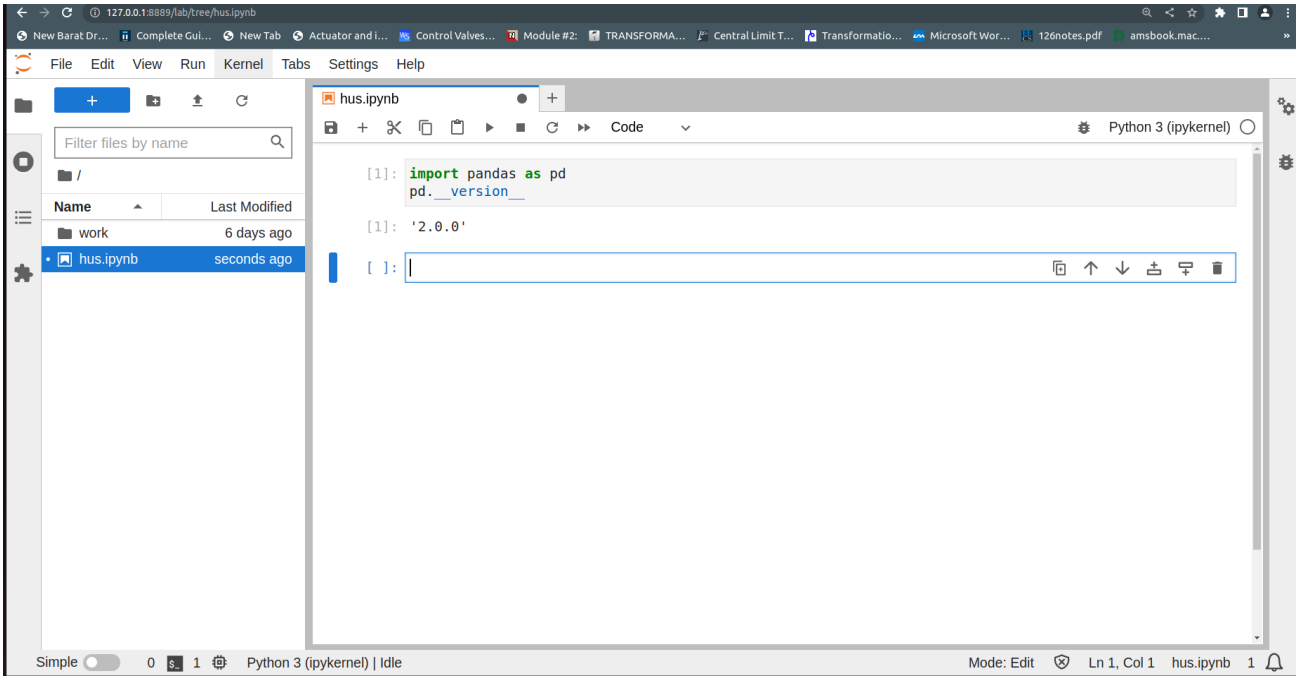
After clicking to the link

http://127.0.0.1:8889/lab?token=64a0ba2832c054443772c4f22bdb1a0b7caca74c9fa78b6a

We will be navigating  to page as show in image below:




Now to verify if pandas in installed we will type **import pandas as pd** and check its version using **pd.__version__** in the python3(ipykernel) file as :




As shown in the image above, the pandas command is running successfully.