Running Lessons With Docker

Correlation-One uses `docker` to run Jupyter Notebook training cases. This allows us to eliminate portability issues between different operating systems.

This document describes how to use `docker` to run training cases.

Supported systems include macOS, Windows (7/8/10), and Linux.

Installing Docker

The installation process differs by operating system. We defer to the official `docker` documentation for the installation instructions:

- Linux: `docker-engine`
- Windows 7/8 and Windows 10 Home (below version 2004): `docker-toolbox`
- macOS (before El Capitan 10.11): `docker-toolbox`
- Windows 10 Home (version 2004 or higher): `docker-desktop`
- Windows 10 Pro/Enterprise/Education (Build 16299 or later): `docker-desktop`
- macOS (from El Capitan 10.11 on): `docker-desktop`

Note: If you are using Windows, you will need to enable virtualization on your machine. This process is machine specific, we defer to this guide for further guidance, but still you may need to search how to do it for your specific machine.

Running the Docker Image

Once `docker` is installed, please follow the below instructions on how to run the relevant `docker` image for ***your*** operating system.

Windows 10 Home (version 2004 or higher)/Pro/Enterprise/Education (Build 16299 or later)

Open the `Command Prompt` as an administrator. In order to do so, please find `CMD` in the start menu, right-click it, and select "Run as Administrator".

Enter the following command to pull and run the `docker` image:

```
docker run -p 8888:8888 -p 8050:8050 -u root -e GRANT_SUDO=yes -v ^
%cd%:/home/jovyan/work jupyter/scipy-notebook:17aba6048f44 ^
start-notebook.sh --NotebookApp.token='' --NotebookApp.password=''
```

Case-writers: Please run the command above from the directory **containing** the `starter-kit/` directory otherwise tests will not be executed.

Students: Ensure that the command is run from a directory containing the cases used in your lesson.

The prompt should pull the `docker` image and run a notebook server. Follow the instructions in the *accessing the notebook server* section.

macOS (from El Capitan 10.11 on) and Linux

On macOS, launch the Terminal application. On Linux, open your preferred terminal.

Enter the following command to pull and run the `docker` image:

```
docker run -p 8888:8888 -p 8050:8050 -u root -e GRANT_SUDO=yes -v \
`pwd`:/home/jovyan/work jupyter/scipy-notebook:17aba6048f44 \
start-notebook.sh --NotebookApp.token='' --NotebookApp.password=''
```

Case-writers: Please run the command above from the directory **containing** the `starter-kit/` directory otherwise tests will not be executed.

Students: Ensure that the command is run from a directory containing the cases used in your lesson.

The prompt should pull the `docker` image and run a notebook server. Follow the instructions in the *accessing the notebook server section.*

Windows 7/8/10 Home (below version 2004) and macOS (before El Capitan 10.11)

Open the `Docker Quickstart Terminal` application.

Enter the following command to know the ip address on which your `default docker-machine` is running:

```
docker-machine ip default
```

The prompt should display the address of your `default docker-machine`. You will use this address to replace the `<docker-machine ip address>` when accessing the notebook server.

Then enter the following command to pull and run the `docker` image:

```
docker run -p 8888:8888 -p 8050:8050 -u root -e GRANT_SUDO=yes -v \
`pwd`:/home/jovyan/work jupyter/scipy-notebook:17aba6048f44 \
start-notebook.sh --NotebookApp.token='' --NotebookApp.password=''
```

Case-writers: Please run the command above from the directory **containing** the `starter-kit/` directory otherwise tests will not be executed.

Students: Ensure that the command is run from a directory containing the cases used in your lesson.

The prompt should pull the `docker` image and run a notebook server. Follow the instructions in the *accessing the notebook server section.*

Accessing the notebook server

After running the `docker run` command with the instructions according to your operating system, you will be able to access the notebook server. You should have been prompted with a message containing the following:

```
[I 22:13:17.840 NotebookApp] Serving notebooks from local directory: /
    home/jovyan
[I 22:13:17.840 NotebookApp] The Jupyter Notebook is running at:
[I 22:13:17.841 NotebookApp] http://(cbc0758b5012 or 127.0.0.1):8888/
[I 22:13:17.841 NotebookApp] Use Control-C to stop this server and shut
    down all kernels (twice to skip confirmation).
```

Depending on the version of `docker` and operating system you are using the URL you need to access the notebook server will change:

For Windows 10 Home (version 2004 or higher)/Pro/Enterprise/Education (Build 16299 or later) and macOS (from El Capitan 10.11 on) and Linux to access the notebook server go to the following URL: http://127.0.0.1:8888

For Windows 7/8/10 Home (below version 2004) and macOS (before El Capitan 10.11) users the notebook server URL should be `http://<docker-manchine ip address>:8888` where `<docker-manchine ip address>` is the ip address you got when running `docker-manchine ip default`

Managing Environments

Case writers: Please ignore this section.

Once a `docker` process is running, you may follow the instructions listed in the `managing-environments.pdf` document.

Managing Docker Processes

Running `docker` images persist in the background once opened. Eventually, you should remove these running processes.

List the running `docker` processes by using `docker ps`. Sample output:

```
docker ps
CONTAINER ...  NAMES
aacda4d33aef   elastic_zhukovsky
```

The output above shows a table of running processes. Most of the fields are omitted for clarity. The `NAMES` field is used to reference the process. Names are randomly generated.

To stop a process, use the `docker stop` command with the name of the process.

```
docker stop <name-of-process>
# e.g., docker stop elastic_zhukovsky
```