**Dockerzied Flask application & upload on DockerHub and Heroku:**

* **Platform:**

Windows 10 Home (64-bit)

Docker toolbox (Version)

* **Step:**

1. Create a Flask application with requirements.txt and Dockerfile.
2. Build docker image to verify if the image is working correctly or not. Following are the commands I used to build and test the image. Make sure that you are in the same directory where your project files (python source files, requriements.txt and Dockerfile) are placed. Following are the commands:
   1. docker pull python
   2. docker image built -t weather\_app .
   3. docker container run -d -p 3000:5000 --name weaApp weather\_app

From the above command, your newly built image will be run as “daemon” and you can check on your localhost:3000. But, in case of Docker Toolbox, you can’t check on localhost:3000. It is because Docker toolbox uses VM and everything is to be accessed via this VM’s IP.

You can get Docker VM’s IP using:

Docker-machine ip

Now, if above command returns IP as “192.168.99.100” then, you can use below address to access your Flask app.

<http://192.168.99.100:3000/>

1. Uploading on docker hub is quite easy and it requires some basic steps as follows:
   1. docker login (Login to your docker hub account)
   2. docker tag weather\_app <username>/<image\_name>:<tag>

(With the above command, you can tag your pre-built image, in our case the image we just built. Tagging convention is according to docker hub which is mentioned in their Wiki)

* 1. docker push <username>/<image\_name>