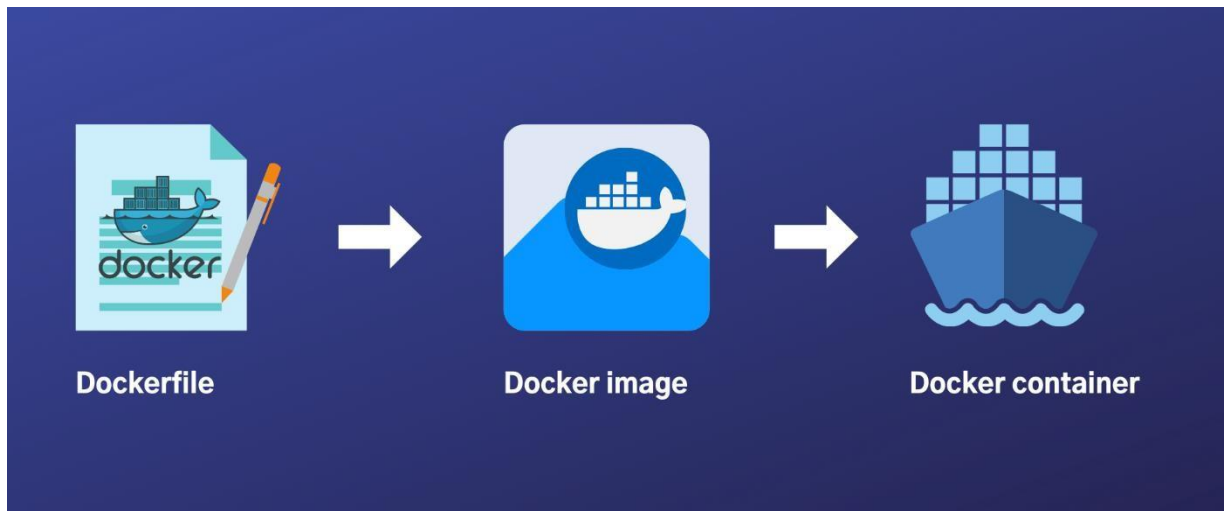


Creating Image to IBM Container Registry :

Date	19 NOVEMBER 2022
Team ID	PNT2022TMID38243
Project Name	Personal Expense Tracker Application



flask-docker

- app.py
- Dockerfile
- requirements.txt
- **venv**

How to modify app.py

```
from flask import Flask
app = Flask(__name__)

@app.route('/')
def hello_geek():
    return '<h1>Hello from Flask & Docker</h2>'

if __name__ == "__main__":
    app.run(debug=True)
```

Now, if we run **python app.py** on the command line to test our Flask app, we should get results similar to the ones shown below:

```
* Serving Flask app 'app' (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: on
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
* Restarting with stat
* Debugger is active!
* Debugger PIN: 316-584-348
```

How to modify the Dockerfile:

```
# syntax=docker/dockerfile:1

FROM python:3.8-slim-buster

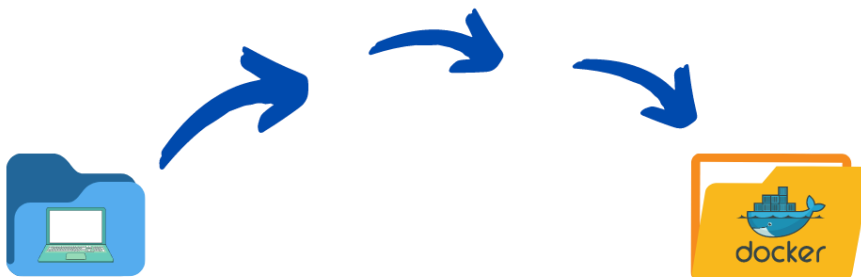
WORKDIR /python-docker

COPY requirements.txt requirements.txt
RUN pip3 install -r requirements.txt

COPY . .

CMD [ "python3", "-m", "flask", "run", "--host=0.0.0.0"]
```

```
# syntax=docker/dockerfile:1
FROM python:3.8-slim-buster
WORKDIR /python-docker
COPY requirements.txt requirements.txt
RUN pip3 install -r requirements.txt
COPY . .
```



Directory on local machine

Directory contained within our
container image

```
CMD [ "python3", "-m", "flask", "run", "--host=0.0.0.0"] if
__name__ == "__main__":
    app.run(debug=True) docker
build --tag python-docker . docker
images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
python-docker	latest	cd52b70b361a	About an hour ago	912MB
headless-cms	latest	e8b253e230ee	43 hours ago	937MB
scrappy	latest	3e7ac0d44890	7 weeks ago	904MB
python	3.9.2	587b1bc803b3	7 months ago	885MB

docker run

```
docker run -d -p 5000:5000 python-docker
```

This time, we'll see the following output if we run it in detached mode and visit localhost at port 5000:



```
docker ps
```

The output is

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS
a173935297cd	python-docker	"python3 -m flask ru..."	5 minutes ago	Up 5 minut

To stop the currently running container, we execute this command:

```
docker stop <container-name>
```

```
docker container prune
```

It removes unused resources, freeing up space and keeping your system clean.

And that's it!