

Containerize The App

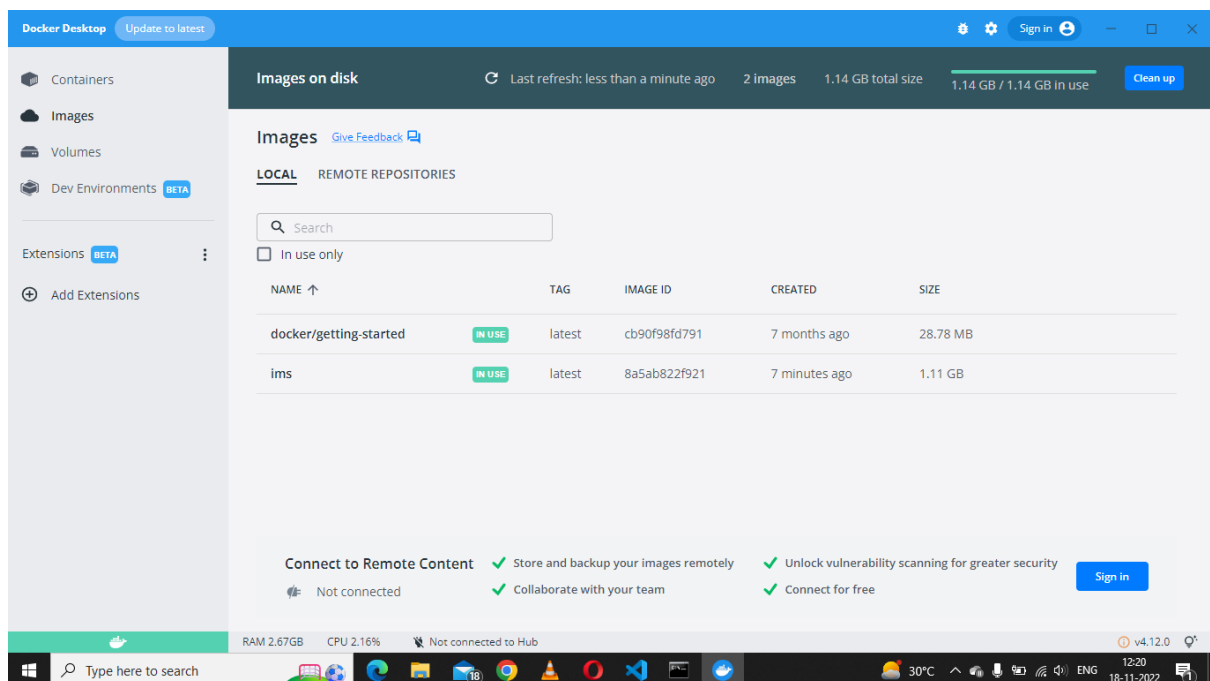
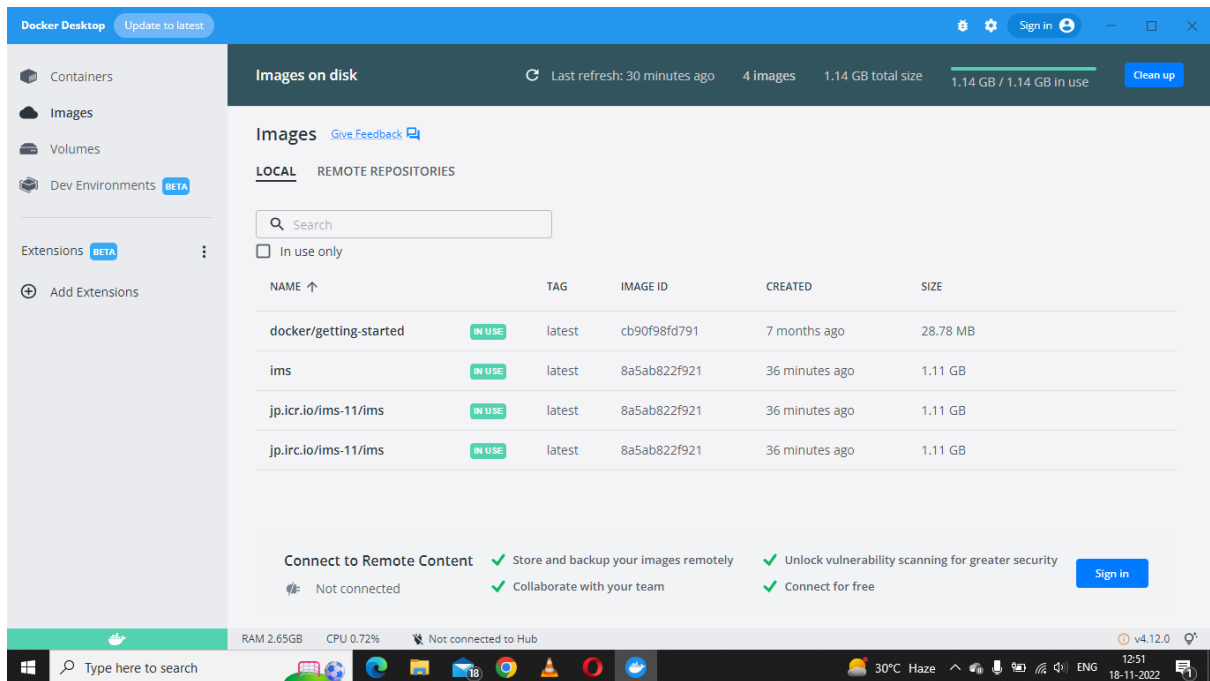
```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19044.2006]
(c) Microsoft Corporation. All rights reserved.

D:\RedStore>docker build -t ims .
[+] Building 1139.6s (11/11) FINISHED
=> [Internal] load build definition from Dockerfile 1.5s
=> => transferring dockerfile: 328 0.3s
=> [Internal] load .dockerignore 1.7s
=> => transferring context: 28 0.1s
=> [Internal] load metadata for docker.io/library/python:3.10.6 6.1s
=> [Internal] load build context 1.2s
=> => transferring context: 2.44kB 0.3s
=> [1/6] FROM docker.io/library/python:3.10.6@sha256:745efd7b7e4aac9a8422bd8c62d8bc35a693e8079a240d29677cb03e6aa91052 572.6s
=> resolve docker.io/library/python:3.10.6@sha256:745efd7b7e4aac9a8422bd8c62d8bc35a693e8079a240d29677cb03e6aa91052 0.5s
=> sha256:745efd7b7e4aac9a8422bd8c62d8bc35a693e8079a240d29677cb03e6aa91052 2.35kB / 2.35kB 0.0s
=> sha256:8d1f943ceaf3b3ce05df5c0926e7958836b048b700170bf9c56d8f37ac13fca 2.22kB / 2.22kB 0.0s
=> sha256:d25ae6380b10283603ff096d777bba5c0b1b9126fb0b0e7d118b9574940bcf84 8.53kB / 8.53kB 0.0s
=> sha256:1671565cc0d48c365c9b661d3fbc1c6e72d01f1b0430c6179580420f99a9d42e 55.01MB / 55.01MB 163.4s
=> sha256:3e94d13e55e7a4ef17ff21376f57fb95c7e1706931f8704aa99260968d81f6e4 5.16MB / 5.16MB 15.6s
=> sha256:fa9c7528c685216129e8e67bf362a7702e7b1daa585ab85546a41508830657d6 10.88MB / 10.88MB 39.2s
=> sha256:53ad072f9cd16fc8eb93b182b20e758e11acc0ef60babe70bf1043c08de1901a 54.58MB / 54.58MB 166.7s
=> sha256:d0b983117533b718374f1701ef593dd2afa6613c7908c6553be8e2a150e6448a 196.79MB / 196.79MB 373.8s
=> sha256:d8092d56ded5476fe7c302256eb4dc6ff495ae8fb4dd28aa18dbcb7581e24a6c 6.29MB / 6.29MB 192.2s
=> extracting sha256:1671565cc0d48c365c9b661d3fbc1c6e72d01f1b0430c6179580420f99a9d42e 56.8s
=> sha256:c71afc637d59adc44c5fd3c348504df82b35bbb204f0057ea22c6ac8a1d285a5 20.02MB / 20.02MB 238.1s
=> sha256:864a10b3c704553e08cb5fcd12fbaee1c07048f6365f0a35e84a285a13da40b 234B / 234B 164.3s
=> sha256:4334b2fe8293d19ddc1c3559093aae88f21601a7c85a31c6da6c0d48fb6ed3c 3.04MB / 3.04MB 208.9s
=> extracting sha256:3e94d13e55e7a4ef17ff21376f57fb95c7e1706931f8704aa99260968d81f6e4 4.6s
=> extracting sha256:fa9c7528c685216129e8e67bf362a7702e7b1daa585ab85546a41508830657d6 5.0s
=> extracting sha256:53ad072f9cd16fc8eb93b182b20e758e11acc0ef60babe70bf1043c08de1901a 53.2s
=> extracting sha256:d0b983117533b718374f1701ef593dd2afa6613c7908c6553be8e2a150e6448a 124.1s
=> extracting sha256:d8092d56ded5476fe7c302256eb4dc6ff495ae8fb4dd28aa18dbcb7581e24a6c 6.3s
=> extracting sha256:c71afc637d59adc44c5fd3c348504df82b35bbb204f0057ea22c6ac8a1d285a5 23.4s
=> extracting sha256:864a10b3c704553e08cb5fcd12fbaee1c07048f6365f0a35e84a285a13da40b 0.1s
=> extracting sha256:4334b2fe8293d19ddc1c3559093aae88f21601a7c85a31c6da6c0d48fb6ed3c 6.2s
[2/6] WORKDIR /app 3.1s
=> [3/6] COPY requirements.txt ./ 3.0s
=> [4/6] RUN pip install -r requirements.txt 83.7s
=> [5/6] RUN pip install ibm_db 434.9s
=> [6/6] COPY . . 4.7s
=> exporting to image 17.8s
=> exporting layers 16.3s
=> writing image sha256:8a5ab822f92129f08dadd56501aa451836322ff4af409b10d7723ce92d1f1eae 0.2s
=> naming to docker.io/library/ims 0.2s
```

```
C:\Windows\System32\cmd.exe - docker run -p 5000:5000 ims
=> => extracting sha256:3e94d13e55e7a4ef17ff21376f57fb95c7e1706931f8704aa99260968d81f6e4 4.6s
=> => extracting sha256:fa9c7528c685216129e8e67bf362a7702e7b1daa585ab85546a41508830657d6 5.0s
=> => extracting sha256:53ad072f9cd16fc8eb93b182b20e758e11acc0ef60babe70bf1043c08de1901a 53.2s
=> => extracting sha256:d0b983117533b718374f1701ef593dd2afa6613c7908c6553be8e2a150e6448a 124.1s
=> => extracting sha256:d8092d56ded5476fe7c302256eb4dc6ff495ae8fb4dd28aa18dbcb7581e24a6c 6.3s
=> => extracting sha256:c71afc637d59adc44c5fd3c348504df82b35bbb204f0057ea22c6ac8a1d285a5 23.4s
=> => extracting sha256:864a10b3c704553e08cb5fcd12fbaee1c07048f6365f0a35e84a285a13da40b 0.1s
=> => extracting sha256:4334b2fe8293d19ddc1c3559093aae88f21601a7c85a31c6da6c0d48fb6ed3c 6.2s
[2/6] WORKDIR /app 3.1s
=> [3/6] COPY requirements.txt ./ 3.0s
=> [4/6] RUN pip install -r requirements.txt 83.7s
=> [5/6] RUN pip install ibm_db 434.9s
=> [6/6] COPY . . 4.7s
=> exporting to image 17.8s
=> exporting layers 16.3s
=> writing image sha256:8a5ab822f92129f08dadd56501aa451836322ff4af409b10d7723ce92d1f1eae 0.2s
=> naming to docker.io/library/ims 0.2s

Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them

D:\RedStore>docker run -p 5000:5000 ims
* Serving Flask app 'app'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 956-863-287
```



Team ID: PNT2022TMID39828

Project: inventory Management System for Retailers.