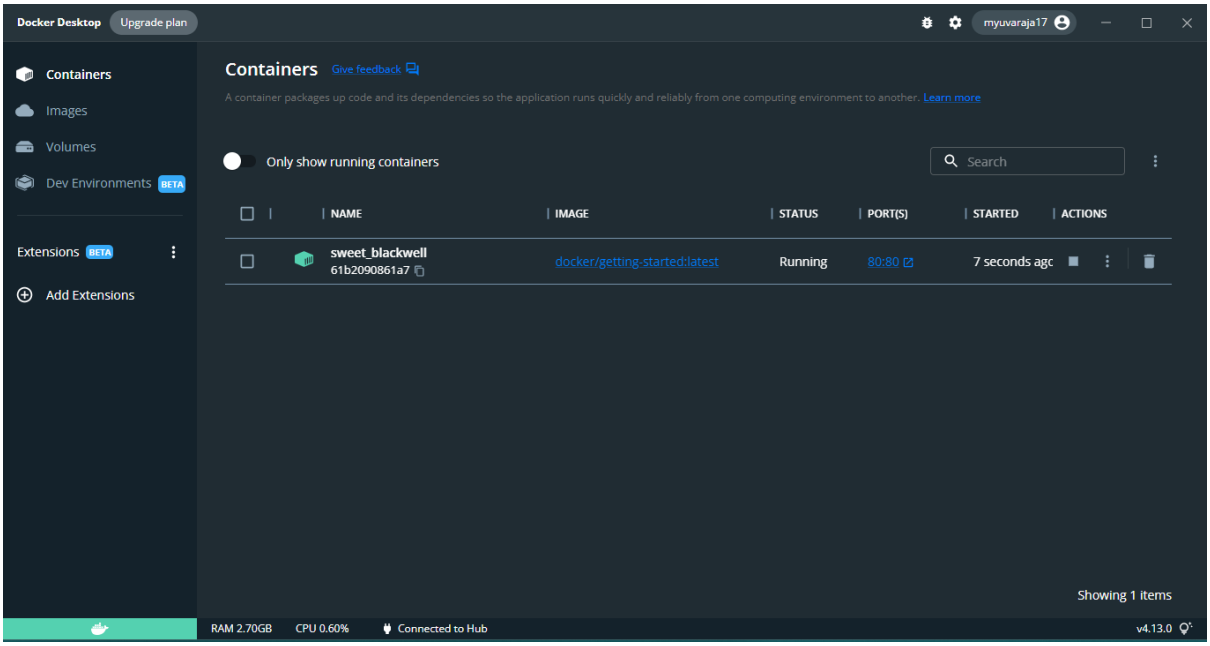
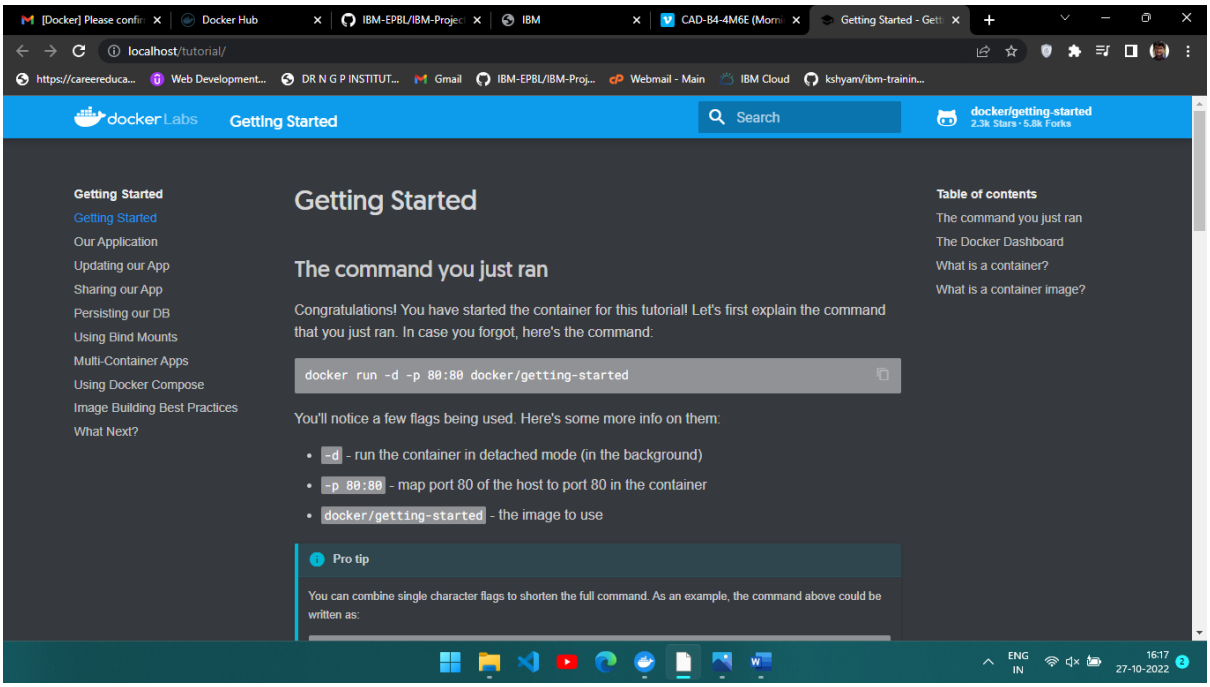
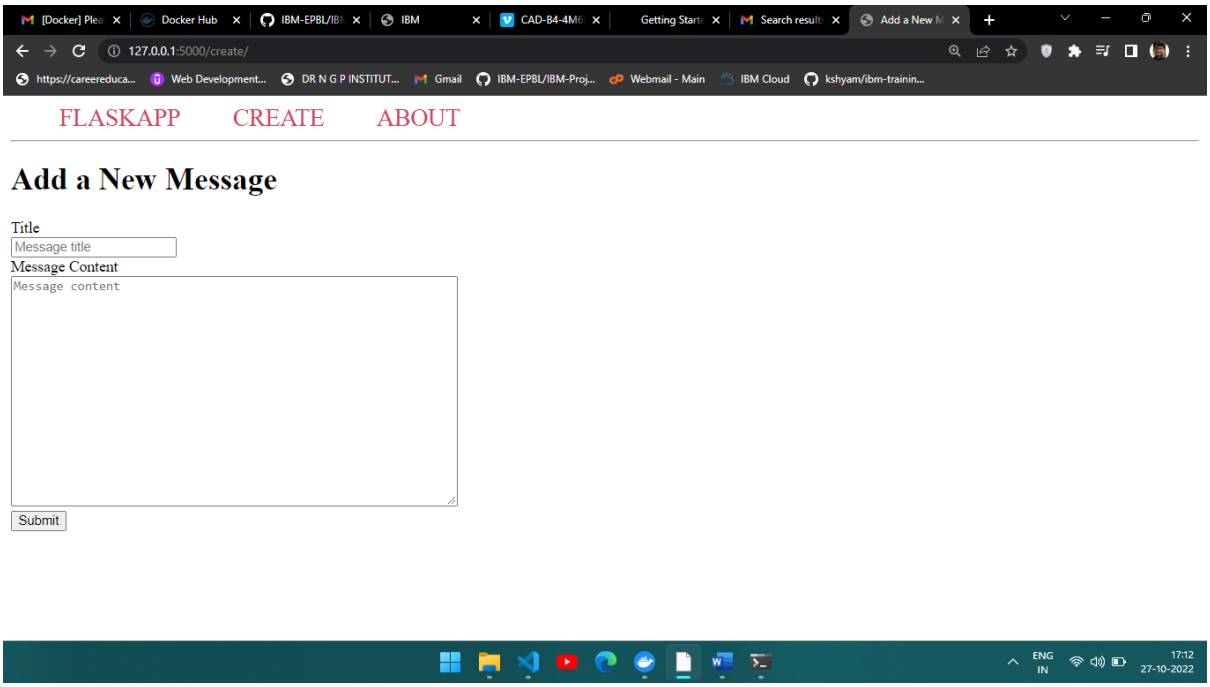
1. Pull a image from docker hub and run it in docker playground.



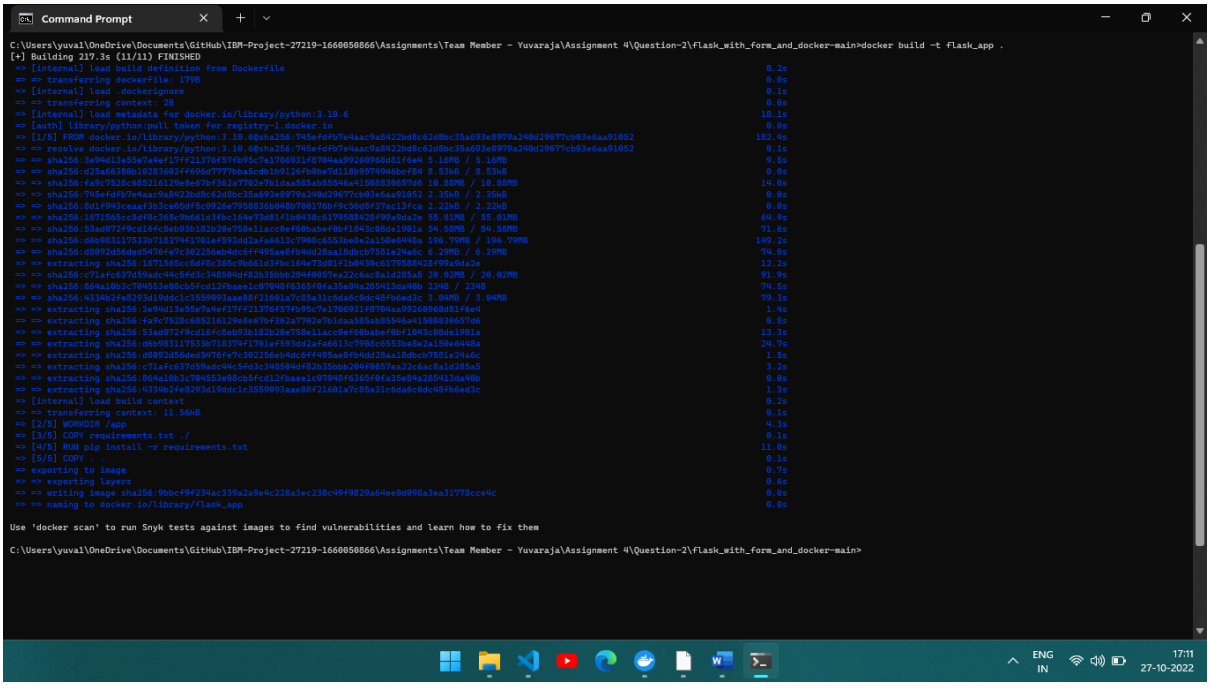


2.Create a docker file for the job portal application and deploy it in Docker desktop application.

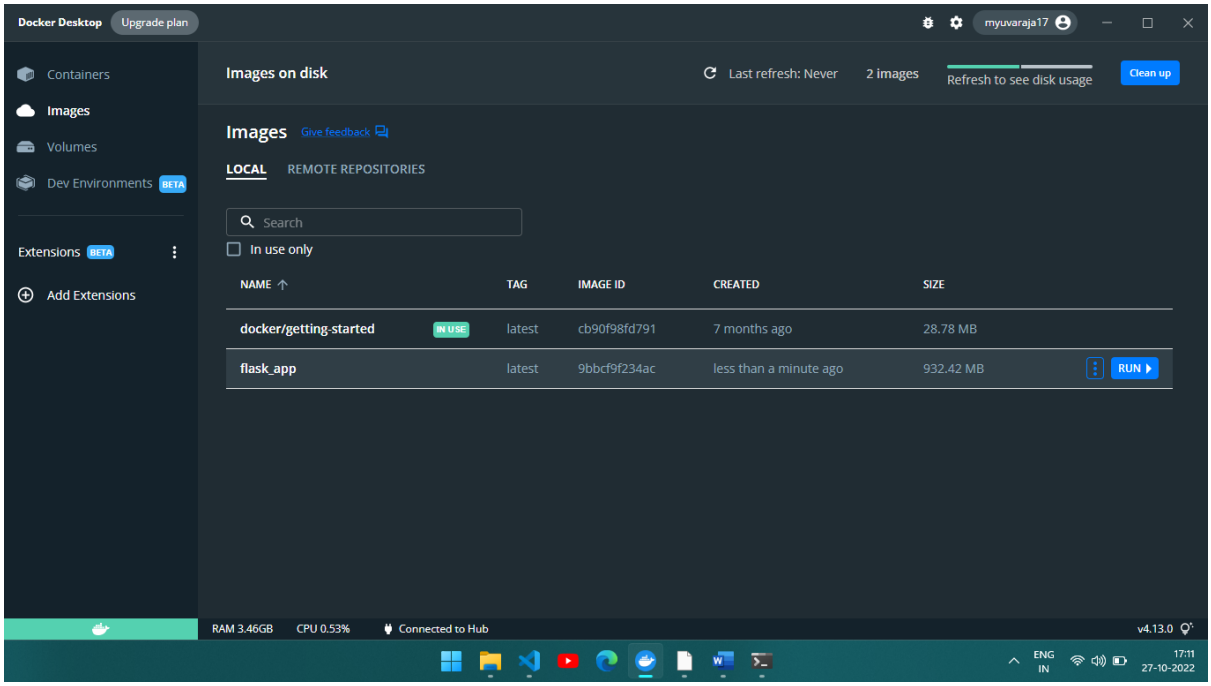
1. Application to be deployed in Docker.



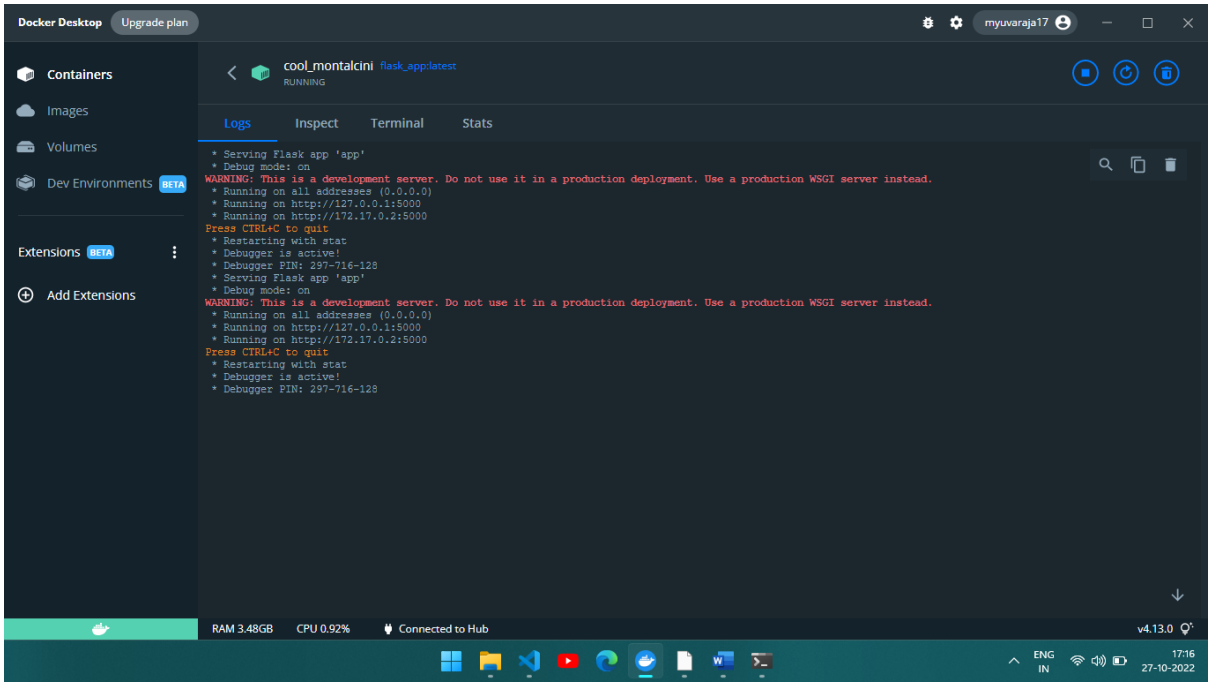
II. Building the application : docker build -t flask_app .



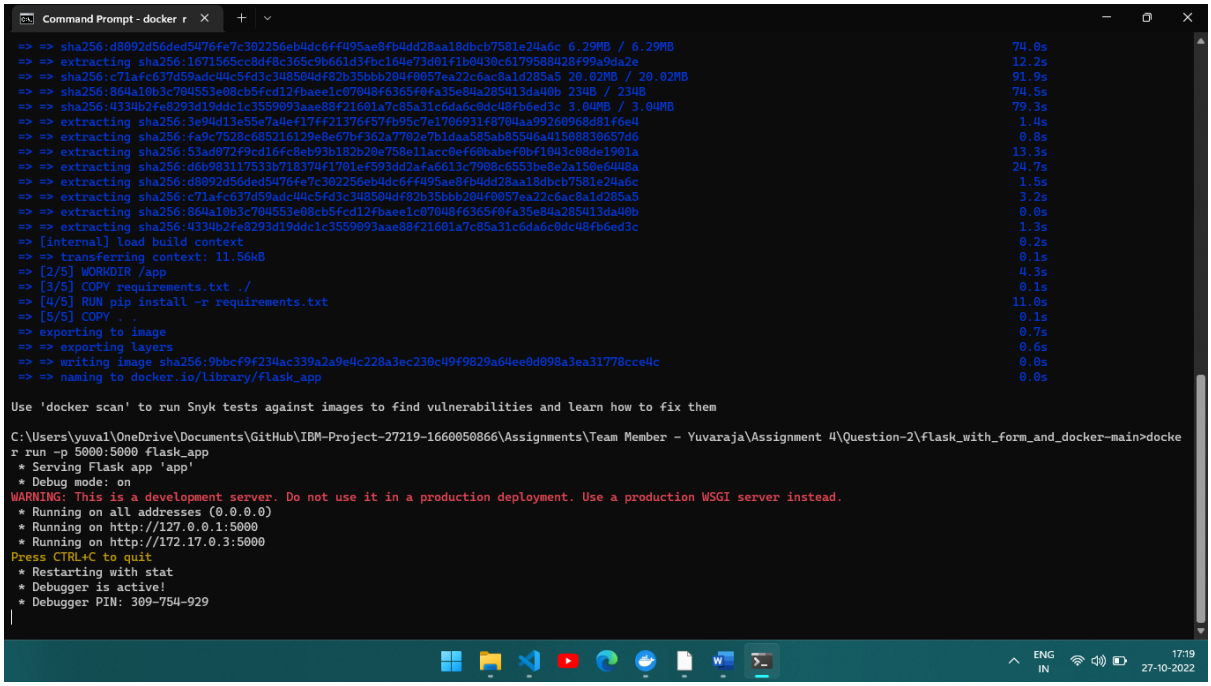
- Image of the deployed application.



The Pushed application is running in docker



Run using cmd


Command Prompt - docker r
=> => sha256:d8092d56ded5476fe7c302256eb4dc6ff495ae8fb4dd28aa18dbcb7581e24a6c 6.29MB / 6.29MB 74.0s
=> => extracting sha256:1671565cc8df8c365c9b661d3fbc164e73d01f1b0430c6179588428f99a9da2e 12.2s
=> => sha256:c71afc637d59adc44c5fd3c34B504df82b35bbb204f0057ea22c6ac8a1d285a5 20.02MB / 20.02MB 91.9s
=> => sha256:864a10b3c704553e08cb5fcd12fbaee1c07048f6365f0fa35e84a285413da40b 234B / 234B 74.5s
=> => sha256:4334b2fe8293d19ddc1c3559093aae88f21601a7c85a31c6da6c0dc48fb6ed3c 3.04MB / 3.04MB 79.3s
=> => extracting sha256:3e94d13e55e7a4ef17ff21376f57fb95c7e1706931f8704aa99260968d81f6e4 1.4s
=> => extracting sha256:fa9c7528c685216129e8e67bf362a7702e7b1daa585ab85546a41508830657d6 0.8s
=> => extracting sha256:53ad072f9cd16fc8eb93b182b20e758e11acc0ef60babef0bf1043c08de1901a 13.3s
=> => extracting sha256:d6b983117533b718374f1701ef593dd2afa6613c7908c6553be8e2a150e6448a 24.7s
=> => extracting sha256:d8092d56ded5476fe7c302256eb4dc6ff495ae8fb4dd28aa18dbcb7581e24a6c 1.5s
=> => extracting sha256:c71afc637d59adc44c5fd3c34B504df82b35bbb204f0057ea22c6ac8a1d285a5 3.2s
=> => extracting sha256:864a10b3c704553e08cb5fcd12fbaee1c07048f6365f0fa35e84a285413da40b 0.0s
=> => extracting sha256:4334b2fe8293d19ddc1c3559093aae88f21601a7c85a31c6da6c0dc48fb6ed3c 1.3s
=> [internal] load build context 0.2s
=> => transferring context: 11.56kB 0.1s
=> [2/5] WORKDIR /app 4.3s
=> [3/5] COPY requirements.txt ./ 0.1s
=> [4/5] RUN pip install -r requirements.txt 11.0s
=> [5/5] COPY . . 0.1s
=> exporting to image 0.7s
=> => exporting layers 0.6s
=> => writing image sha256:9bbcf9f234ac339a2a9e4c228a3ec230c49f9829a64ee0d098a3ea31778cce4c 0.0s
=> => naming to docker.io/library/flask_app 0.0s
Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them
C:\Users\yuva1\OneDrive\Documents\GitHub\IBM-Project-27219-1660050866\Assignments\Team Member - Yuvaraja\Assignment 4\Question-2\flask_with_form_and_docker-main>docke
r run -p 5000:5000 flask_app
* 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 all addresses (0.0.0.0)
* Running on http://127.0.0.1:5000
* Running on http://172.17.0.3:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
* Debugger PIN: 309-754-929
ENG
IN
17:19
27-10-2022

3.