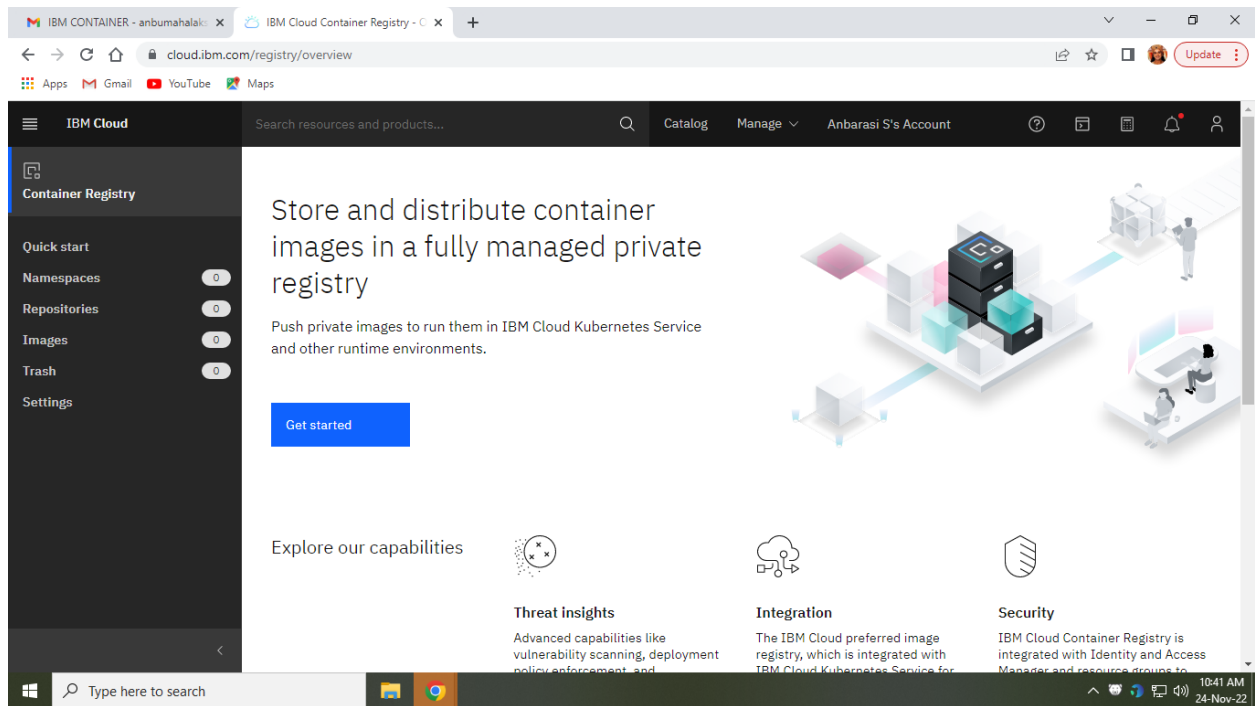


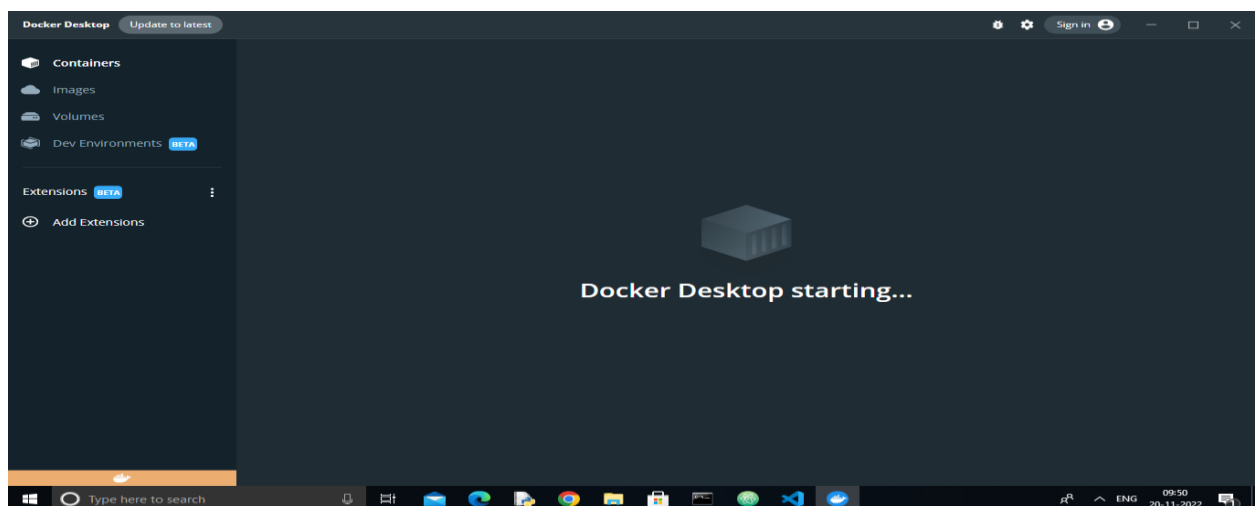
UPLOAD IMAGE TO IBM CONTAINER REGISTRY

Date	20/11/2022
Team ID	PNT2022TMID46354
Project Name	Smart fashion Recommender Application

IBM Container Registry :



Docker Running :



Container Registry :

The screenshot shows the Docker Desktop interface. On the left sidebar, the 'Containers' tab is selected. The main area displays a table with one container:

	NAME	IMAGE	STATUS	PORT(S)	STARTED	ACTIONS
<input type="checkbox"/>	stupefied_galois ac9ce08124cf	docker/getting-started:latest	Running	80:80	45 seconds ago	[Stop] [Refresh] [Delete]

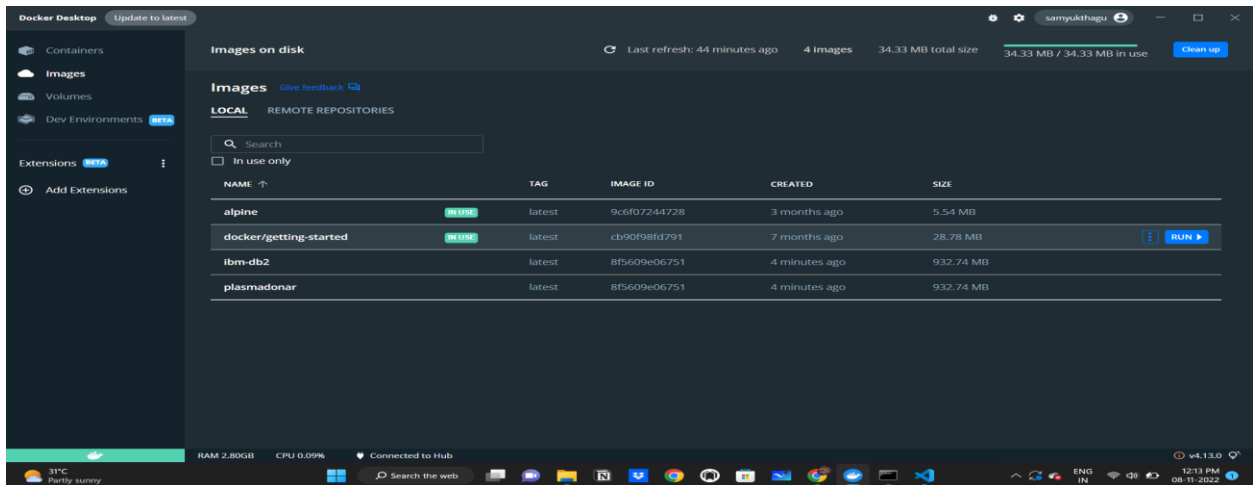
At the bottom, system statistics show RAM 1.73GB, CPU 0.00%, and a status of 'Not connected to Hub'. The taskbar at the bottom shows the date as 08-11-2022.

The screenshot shows the Docker Desktop interface with four containers listed:

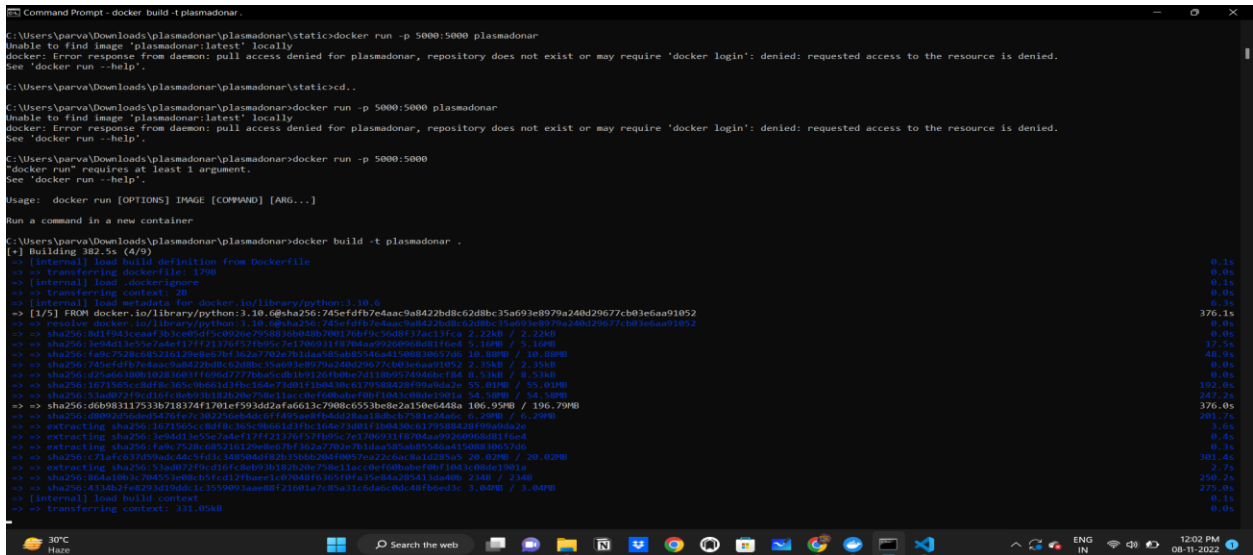
	NAME	IMAGE	STATUS	PORT(S)	STARTED	ACTIONS
<input type="checkbox"/>	stupefied_galois ac9ce08124cf	docker/getting-started:latest	Running	80:80	10 minutes ago	[Stop] [Refresh] [Delete]
<input type="checkbox"/>	stoic_merkle fc08f7378b47	docker/getting-started:latest	Running		4 minutes ago	[Stop] [Refresh] [Delete]
<input type="checkbox"/>	hardcore_mendeleev 812c40ec07e9	alpine:latest	Exited			[Start] [Refresh] [Delete]
<input type="checkbox"/>	sweet_vaughan ad4efcfa0056	docker/getting-started:latest	Created (128)	80:80		[Start] [Refresh] [Delete]

System statistics at the bottom show RAM 1.77GB, CPU 0.10%, and 'Not connected to Hub'. The taskbar shows the date as 08-11-2022.

Upload images :



Running on terminal to push the images :



```
Select Command Prompt - docker build -t plasmadonar .

Usage: docker run [OPTIONS] IMAGE [COMMAND] [ARG...]

Run a command in a new container

C:\Users\parva\Downloads\plasmadonar>docker build -t plasmadonar .
[+] Building 573.2s (8/9)
  => [internal] load build definition from Dockerfile 0.53s
  => transferring dockerfile: 179B 0.4s
  => [internal] load .dockerignore 0.1s
  => transferring context: 2B 0.0s
  => [internal] load metadata for docker.io/library/python:3.10.6 0.53s
  => [1/4] FROM docker.io/library/python:3.10.6@sha256:729ef47b7e4aac9a8422b08c62486c35a693e8979a28d129677c801e6a091052 953.7s
  => resolve docker.io/library/python:3.10.6@sha256:729ef47b7e4aac9a8422b08c62486c35a693e8979a28d129677c801e6a091052 0.0s
  => sha256:821f943e9a4f3b3ce95df3c093a67050836040b7081708f9c56b8f37ac13fca 2.22kB / 2.22kB 0.0s
  => sha256:3a04d12e55e7ade174f22178f371809c7e478092118704a992606da0116e4 5.16kB / 5.16kB 12.5s
  => sha256:f49c7526c485216129e0e67bf362a7702c7b18a5955a65546a4150843065700 10.80kB / 10.80kB 20.9s
  => sha256:745efdfb7e4aac9a8422b08c62486c35a693e8979a28d129677c801e6a091052 2.35kB / 2.35kB 0.0s
  => sha256:4250c100810283004f4f06d7779b0c4d109124f080a724180957004de4f04 8.53kB / 8.53kB 0.0s
  => sha256:1471565c80f8c365c9a6d1d31bc1106c75081f104386c179588420f09ad4de 55.81kB / 55.81kB 192.0s
  => sha256:53a0972f9cd16f8b03c182b28c75b01ac0ef50b0baf0bf1043c08de1901a 54.50kB / 54.50kB 267.2s
  => sha256:d809311753b712374f1701ef093d2afae013c7908c4553be0e2a150e6448a 196.79kB / 196.79kB 532.7s
  => sha256:d809311753b712374f1701ef093d2afae013c7908c4553be0e2a150e6448a 401.7s
  => extracting sha256:1871565c80f8c365c9a6d1d31bc1106c75081f104386c179588420f09ad4de 3.6s
  => extracting sha256:3e6d13e5e7ade174f22178f371809c7e478092118704a992606da0116e4 0.4s
  => extracting sha256:f49c7526c485216129e0e67bf362a7702c7b18a5955a65546a4150843065700 0.3s
  => sha256:c71ef05709ad4de5f43c148904d482653a6c204f0057a22c0ac3a1d285a5 20.62kB / 20.62kB 301.4s
  => extracting sha256:53a0972f9cd16f8b03c182b28c75b01ac0ef50b0baf0bf1043c08de1901a 2.7s
  => sha256:8a41063c704553a08c5fcd1709e1c07040f6305f0f435e8a285412da00 234B / 234B 250.2s
  => sha256:41342f6823af1043c1550813a08f21091a7c83a31f6ade04c48f0e03c 3.40kB / 3.40kB 275.0s
  => extracting sha256:d809311753b712374f1701ef093d2afae013c7908c4553be0e2a150e6448a 12.0s
  => extracting sha256:d809311753b712374f1701ef093d2afae013c7908c4553be0e2a150e6448a 0.0s
  => extracting sha256:1871565c80f8c365c9a6d1d31bc1106c75081f104386c179588420f09ad4de 0.9s
  => extracting sha256:8a41063c704553a08c5fcd1709e1c07040f6305f0f435e8a285412da00 0.0s
  => extracting sha256:41342f6823af1043c1550813a08f21091a7c83a31f6ade04c48f0e03c 0.4s
  => [internal] load build context 0.1s
  => transferring context: 311.0kB 0.0s
  => [2/5] WORKDIR /app 0.1s
  => [3/5] COPY requirements.txt ./ 0.1s
  => [4/5] pip install -r requirements.txt 12.1s
  => [5/5] COPY . . 0.2s
```

```
Microsoft Windows [Version 10.0.22000.1098]
(c) Microsoft Corporation. All rights reserved.

C:\Users\parva>docker run -d -p 80:80 docker/getting-started
Unable to find image 'docker/getting-started:latest' locally
latest: Pulling from docker/getting-started
ff989388f0da: Pull complete
0b62ba5fcd: Pull complete
4b630e65cb3b: Pull complete
801ed9e2b976: Pull complete
bc19f3e0e01: Pull complete
4071be97c256: Pull complete
70b306fa5b: Pull complete
0c9232f52506: Pull complete
Digest: sha256:b558b874169471bd4e6bdeac8c303b271a7ee8553ba7481b73b2bf597aae
Status: Downloaded newer image for docker/getting-started:latest
ac9ce08124cf2d8e43bb14de75cc479902b99ad5c4d680bc9051371b65f26b

C:\Users\parva>docker container run alpine echo "Hello world"
Unable to find image 'alpine:latest' locally
docker: Error response from daemon: pull access denied for alpine, repository does not exist or may require 'docker login': requested access to the resource is denied.
See 'docker run --help'.

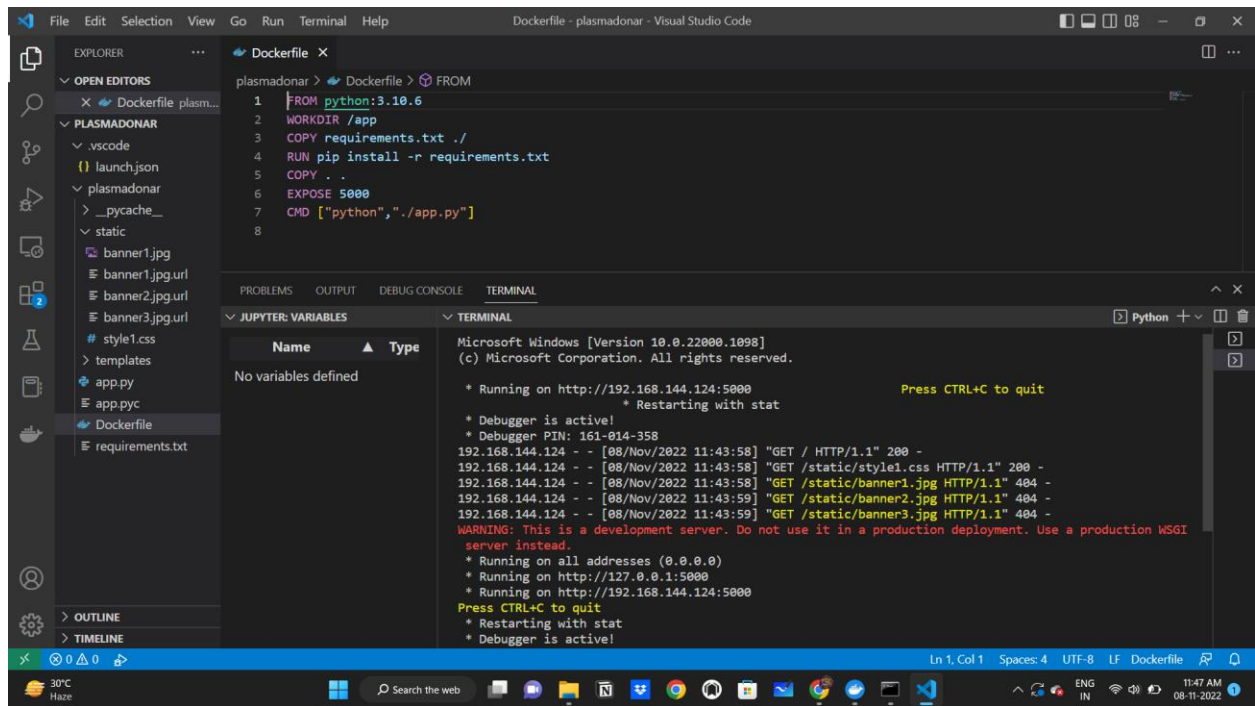
C:\Users\parva>docker container run alpine echo "Hello world"
Unable to find image 'alpine:latest' locally
latest: Pulling from library/alpine
21ec9ee27d5: Pull complete
Digest: sha256:bc41182d7ef5ffc53a40b04de725193bc10142a1243f395ee852a8d9730fc2ad
Status: Downloaded newer image for alpine:latest
Hello world

C:\Users\parva>cd..

C:\Users>docker run -d -p 80:80 docker/getting-started
Unable to find image 'docker/getting-started:latest' locally
latest: Pulling from docker/getting-started
ff989388f0da: Pull complete
0b62ba5fcd: Pull complete
4b630e65cb3b: Pull complete
801ed9e2b976: Pull complete
bc19f3e0e01: Pull complete
4071be97c256: Pull complete
70b306fa5b: Pull complete
0c9232f52506: Pull complete
Digest: sha256:b558b874169471bd4e6bdeac8c303b271a7ee8553ba7481b73b2bf597aae
Status: Downloaded newer image for docker/getting-started:latest
ac9ce08124cf2d8e43bb14de75cc479902b99ad5c4d680bc9051371b65f26b

C:\Users>docker container run alpine echo "Hello world"
Unable to find image 'alpine:latest' locally
docker: Error response from daemon: pull access denied for alpine, repository does not exist or may require 'docker login': denied; requested access to the resource is denied.
See 'docker run --help'.
```

Coding :



The screenshot shows the Visual Studio Code interface with a Dockerfile open in the editor. The Dockerfile contains the following instructions:

```
1 FROM python:3.10.6
2 WORKDIR /app
3 COPY requirements.txt ./
4 RUN pip install -r requirements.txt
5 COPY . .
6 EXPOSE 5000
7 CMD ["python","./app.py"]
8
```

The terminal window at the bottom displays the output of the Docker build and run process. It shows the container starting on http://192.168.144.124:5000 and serving static files. The output includes the following messages:

```
Microsoft Windows [Version 10.0.22000.1098]
(c) Microsoft Corporation. All rights reserved.

* Running on http://192.168.144.124:5000
* Restarting with stat
* Debugger is active!
* Debugger PIN: 161-014-358

192.168.144.124 - - [08/Nov/2022 11:43:58] "GET / HTTP/1.1" 200 -
192.168.144.124 - - [08/Nov/2022 11:43:58] "GET /static/style1.css HTTP/1.1" 200 -
192.168.144.124 - - [08/Nov/2022 11:43:58] "GET /static/banner1.jpg HTTP/1.1" 404 -
192.168.144.124 - - [08/Nov/2022 11:43:59] "GET /static/banner2.jpg HTTP/1.1" 404 -
192.168.144.124 - - [08/Nov/2022 11:43:59] "GET /static/banner3.jpg HTTP/1.1" 404 -
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://192.168.144.124:5000
Press CTRL+C to quit
* Restarting with stat
* Debugger is active!
```

The status bar at the bottom indicates the current file is Dockerfile, using UTF-8 encoding, LF line endings, and is located at line 1, column 1.