

## Docker Installation

```
[sriha@MacBook-Air ~ % docker --version
Docker version 20.10.17, build 100c701
sriha@MacBook-Air ~ %
```

## Docker Login

```
[sriha@MacBook-Air ~ % docker login
Authenticating with existing credentials...
Login Succeeded

Logging in with your password grants your terminal complete access to your account.
For better security, log in with a limited-privilege personal access token. Learn more at https://docs.docker.com/go/access-tokens/
sriha@MacBook-Air ~ %
```

## Docker Desktop Setup

The screenshot shows the Docker Desktop application window. The left sidebar contains navigation options: Containers, Images, Volumes, Dev Environments (marked BETA), and Extensions (marked BETA). The main area is titled 'Containers' and shows a list of 12 containers. The status bar at the bottom indicates RAM usage (1.57GB), CPU usage (2.97%), and connection to the Hub.

	NAME	IMAGE	STATUS	PORT(S)	STARTED	ACTIONS
<input type="checkbox"/>	adoring_cray 60feddb5462c	nginx:latest	Exited	-		
<input type="checkbox"/>	vibrant_cerf a60a6094fa07	nginx:latest	Exited	-		
<input type="checkbox"/>	quirky_jennings ef10eee74b87	alpine:latest	Exited (137)	-		
<input type="checkbox"/>	naughty_hugle 1e167470e9af	alpine:latest	Created (126)	-		
<input type="checkbox"/>	stupefied_jennings e43086b71c10	alpine:latest	Created (127)	-		
<input type="checkbox"/>	condescending_wilson 2d7d6a8c79bc	alpine:latest	Created (127)	-		
<input type="checkbox"/>	confident_galileo c20ae4bd5437	alpine:latest	Exited	-		
<input type="checkbox"/>	awesome_shtern e1ce81193dd8	ubuntu:latest	Exited	-		
<input type="checkbox"/>	heuristic_sutherland 2d737994d2db	ubuntu:latest	Exited	-		
<input type="checkbox"/>	wizardly_einstein 5b75f88180ed	ubuntu:latest	Created (127)	-		
<input type="checkbox"/>	silly_kilby d79236778cf2	hello-world:latest	Exited	-		
<input type="checkbox"/>	flamboyant_merkle					

## Docker run Ubuntu

```
[sriha@MacBook-Air ~ % docker run -i -t ubuntu
[root@2d77cd015bfb:/# ls
bin boot dev etc home lib media mnt opt proc root run sbin srv sys tmp usr var
```

## Docker run Alpine

```
[sriha@MacBook-Air ~ % docker run -i -t alpine
[/ # ls
bin dev etc home lib media mnt opt proc root run sbin srv sys tmp usr var
```

## Docker Build React-App

```
[sriha@MacBook-Air sa-frontend % docker build -f Dockerfile -t sriharshaperi/sentiment-analysis-frontend .  
[+] Building 0.2s (7/7) FINISHED  
=> [internal] load build definition from Dockerfile  
=> => transferring dockerfile: 85B  
=> [internal] load .dockerignore  
=> => transferring context: 63B  
=> [internal] load metadata for docker.io/library/nginx:latest  
=> [internal] load build context  
=> => transferring context: 1.88MB  
=> [1/2] FROM docker.io/library/nginx  
=> [2/2] COPY build /usr/share/nginx/html  
=> exporting to image  
=> => exporting layers  
=> => writing image sha256:7e3fc68d13629809e67362422130e252781e328bb5b1b6bdf19693e95e99f98a  
=> => naming to docker.io/sriharshaperi/sentiment-analysis-frontend
```

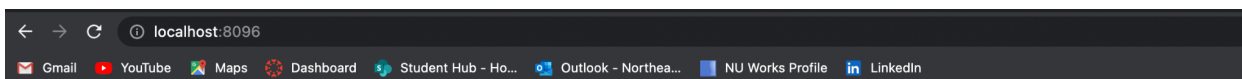
## Docker Push React-App

```
[sriha@MacBook-Air sa-frontend % docker push sriharshaperi/sentiment-analysis-frontend  
Using default tag: latest  
The push refers to repository [docker.io/sriharshaperi/sentiment-analysis-frontend]  
66b8a078f8c9: Pushed  
59cd114add79: Mounted from library/nginx  
060241690833: Mounted from library/nginx  
7a89740679a7: Mounted from library/nginx  
caf1f2684a98: Mounted from library/nginx  
d416aac930f7: Mounted from library/nginx  
c75eaa0eefd3: Mounted from library/nginx  
latest: digest: sha256:5f08184b87c626375d5dd91fb41ec674ffc1c4c3a2a65e527c1a8027313e5a89 size: 1780  
sriha@MacBook-Air sa-frontend %
```

## Docker Run React-App

```
[sriha@MacBook-Air sa-frontend % docker run -d -p 8096:80 sriharshaperi/sentiment-analysis-frontend  
0b4c63b14cb01f7dd519259cde587c8159098ad571661d219f89c23a1c9ce49b  
sriha@MacBook-Air sa-frontend %
```

## React App running on Port 8096 on docker image



# CSYE 7220 Sentiment Analyser

Type your sentence.

SEND

## Docker Build Python Flask App

```
sriha@MacBook-Air sa-logic % docker build -f Dockerfile -t sriharshaperi/sentiment-analysis-logic .
[+] Building 54.5s (10/10) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 279B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/library/python:3.8-slim
=> [auth] library/python:pull token for registry-1.docker.io
=> [internal] load build context
=> => transferring context: 1.43kB
=> [1/4] FROM docker.io/library/python:3.8-slim@sha256:c8d94ad93743b388e9b1e938d7fc6b6cb973b40ac6d0449c3718e418d27d324a
=> => resolve docker.io/library/python:3.8-slim@sha256:c8d94ad93743b388e9b1e938d7fc6b6cb973b40ac6d0449c3718e418d27d324a
=> => sha256:8752a7bdcccfcca675ed92a3a15ce279225db9ce63144bb24e26efc01952a77f 1.37kB / 1.37kB
=> => sha256:9d2fe4c7448da866c54aaf4b97725543472b3172391bfb4ff82f86ad1a1f2e0 7.55kB / 7.55kB
=> => sha256:61358e3667e2d11218ffd1160790a79eb2f7cd1fa39485e1de62d4f5e8eb72b0 858.91kB / 858.91kB
=> => sha256:3d74465b118267eb2fcd4cddfa56a1d4b965c6fc3348759424f27475b06d126d 11.34MB / 11.34MB
=> => sha256:0938e7741216c729b49eaf9bcaf2207e28c2945d17f4e3e18803a7c0e71a3175 233B / 233B
=> => sha256:c8d94ad93743b388e9b1e938d7fc6b6cb973b40ac6d0449c3718e418d27d324a 1.86kB / 1.86kB
=> => extracting sha256:61358e3667e2d11218ffd1160790a79eb2f7cd1fa39485e1de62d4f5e8eb72b0
=> => sha256:deb3b6a78ebce2d40c58db494c527145ff0bb69ec3f153b8f8e28fc1250d620b 2.96MB / 2.96MB
=> => extracting sha256:3d74465b118267eb2fcd4cddfa56a1d4b965c6fc3348759424f27475b06d126d
=> => extracting sha256:0938e7741216c729b49eaf9bcaf2207e28c2945d17f4e3e18803a7c0e71a3175
=> => extracting sha256:deb3b6a78ebce2d40c58db494c527145ff0bb69ec3f153b8f8e28fc1250d620b
=> [2/4] COPY sa /app
=> [3/4] WORKDIR /app
=> [4/4] RUN pip3 install -r requirements.txt && python3 -m textblob.download_corpora
=> exporting to image
=> => exporting layers
=> => writing image sha256:7bdc5962f496c4c1a3634bd0319a8827045b1a385e8497659b0ed54ce032d75f
=> => naming to docker.io/sriharshaperi/sentiment-analysis-logic
sriha@MacBook-Air sa-logic %
```

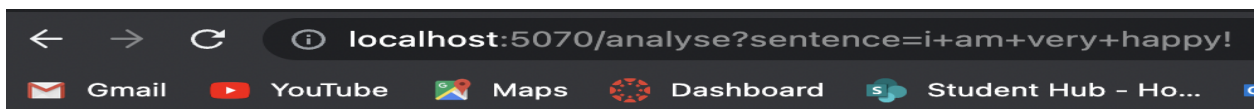
## Docker push Python Flask App

```
sriha@MacBook-Air sa-logic % docker push sriharshaperi/sentiment-analysis-logic
Using default tag: latest
The push refers to repository [docker.io/sriharshaperi/sentiment-analysis-logic]
feb93180903e: Pushed
5f70bf18a086: Layer already exists
8eff40da8ebd: Pushed
acb0741ebfed: Mounted from library/python
66bf7e1351e9: Mounted from library/python
db8d85528ba0: Mounted from library/python
3b937c2aa23f: Mounted from library/python
c75eaa0eefd3: Mounted from sriharshaperi/sentiment-analysis-frontend
latest: digest: sha256:80a6bbafe04035a314992097b73cb2a416793d49645ebd039c26b85993121c2e size: 1994
sriha@MacBook-Air sa-logic %
```

## Docker Run Python Flask App

```
sriha@MacBook-Air sa-logic % docker run -d -p 5070:5020 sriharshaperi/sentiment-analysis-logic
e3c7915133ca44f91020871ed3acc54d4b3e51e5668a7267227aba24f4336ffa
sriha@MacBook-Air sa-logic %
```

## Running Python Flask App on port 5070 on docker image



# 1.0

# Docker Build Spring Boot App

```
sriha@MacBook-Air sentiment-analysis-webapp % docker build -f Docker -t sriharshaperi/sa-springboot-webapp .
[+] Building 1.4s (8/8) FINISHED
=> [internal] load build definition from Docker
=> => transferring dockerfile: 527B
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [internal] load metadata for docker.io/library/openjdk:17
=> [auth] library/openjdk:pull token for registry-1.docker.io
=> [internal] load build context
=> => transferring context: 27.70MB
=> [1/2] FROM docker.io/library/openjdk:17@sha256:528707081fdb9562eb819128a9f85ae7fe000e2fbaeaf9f87662e7b3f38cb7d8
=> CACHED [2/2] ADD target/sentiment-analysis-api-0.0.1-SNAPSHOT.jar /
=> exporting to image
=> => exporting layers
=> => writing image sha256:774e6643dbf5935821a357a9d7c2ba43473926bc07d294b0339a69d0458bd92f
=> => naming to docker.io/sriharshaperi/sa-springboot-webapp
sriha@MacBook-Air sentiment-analysis-webapp %
```

## Docker Push Spring Boot App

```
[sriha@MacBook-Air sentiment-analysis-webapp % pwd
/Users/sriha/Documents/workspace-spring-tool-suite-4-4.16.0.RELEASE/sentiment-analysis-webapp
[sriha@MacBook-Air sentiment-analysis-webapp % docker push sriharshaperi/sa-springboot-webapp
Using default tag: latest
The push refers to repository [docker.io/sriharshaperi/sa-springboot-webapp]
e9189e187390: Mounted from sriharshaperi/web-app
e017d39c755a: Layer already exists
03ee828ef0e4: Layer already exists
b0d4c4485e7e: Layer already exists
latest: digest: sha256:461bee1d9eb24a0431686b1b3adb179fd5154f13499f21767bee701aa8c4fe10 size: 1166
sriha@MacBook-Air sentiment-analysis-webapp %
```

## Docker Run Spring Boot App

```
sriha@MacBook-Air sentiment-analysis-webapp % docker run -p 8093:3869 sriharshaperi/sa-springboot-webapp
```

```
(v2.7.4)

:: Spring Boot ::

2022-10-10 14:19:02.414 INFO 1 --- main] com.springsa.DemoApplication : Starting DemoApplication v0.0.1-SNAPSHOT using Java 17.0.2 on b752a7ae7ecb with PID 1
api-0.0.1-SNAPSHOT.jar started by root in /)
2022-10-10 14:19:02.416 INFO 1 --- main] com.springsa.DemoApplication : No active profile set, falling back to 1 default profile: "default"
2022-10-10 14:19:03.127 INFO 1 --- main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port(s): 3869 (http)
2022-10-10 14:19:03.132 INFO 1 --- main] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2022-10-10 14:19:03.132 INFO 1 --- main] org.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/9.0.65]
2022-10-10 14:19:03.378 INFO 1 --- main] o.a.w.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext
2022-10-10 14:19:03.378 INFO 1 --- main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 729 ms
2022-10-10 14:19:03.829 INFO 1 --- main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port(s): 3869 (http) with context path ''
2022-10-10 14:19:03.835 INFO 1 --- main] com.springsa.DemoApplication : Started DemoApplication in 1.642 seconds (JVM running for 1.941)
```

## Spring Boot App running on Port 8097 on docker image

The screenshot shows the Chrome DevTools Network tab. The top bar indicates a POST request to `http://localhost:8097/sentiment`. The 'Body' tab is selected, showing the request body as a JSON object: `{\"sentence\": \"I am very happy!\"}`. The response status is 200 OK, and the response body is a JSON object: `{\"sentence\": \"I am very happy!\", \"polarity\": 1.0}`.