Name – Mansi Saini
Roll.No – R171218123
DevOps Batch-2 (5th Semester)
Submitted to- Mr. Hitesh Kumar

CICD LAB- Experiment (Docker Image)

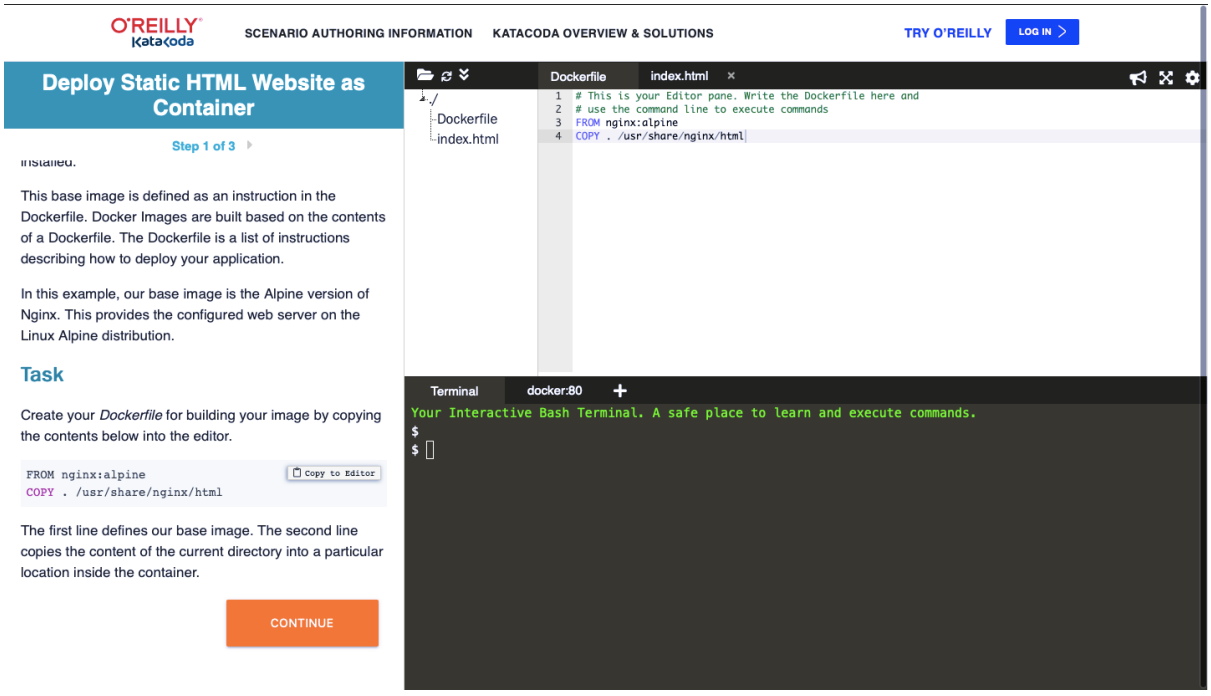
This base image is defined as an instruction in the Dockerfile. Docker Images are built based on the contents of a Dockerfile. The Dockerfile is a list of instructions describing how to deploy your application.

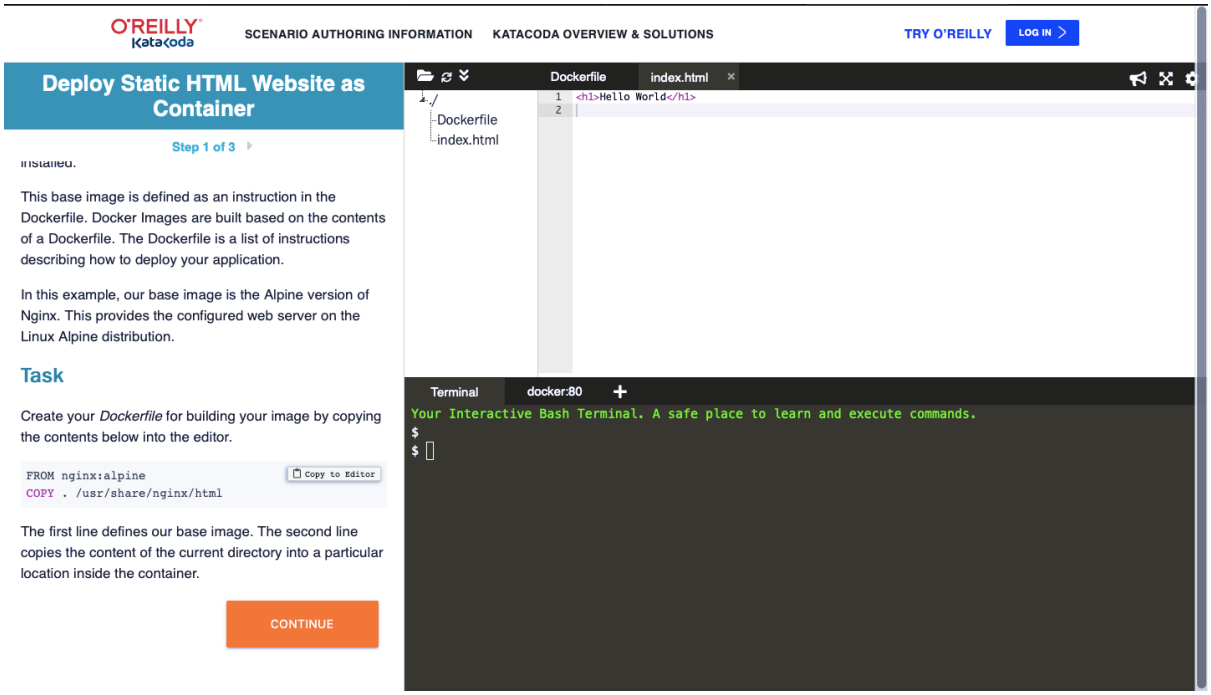
In this example, our base image is the Alpine version of Nginx. This provides the configured web server on the Linux Alpine distribution.

Task: Create your *Dockerfile* for building your image by copying the contents below into the editor.

```
FROM nginx:alpine
COPY . /usr/share/nginx/html
```

The first line defines our base image. The second line copies the content of the current directory into a particular location inside the container.





The Dockerfile is used by the Docker CLI *build* command. The *build* command executes each instruction within the Dockerfile. The result is a built Docker Image that can be launched and run your configured app.

The build command takes in some different parameters. The format is *docker build -t <build-directory>*. The *-t*parameter allows you to specify a friendly name for the image and a tag, commonly used as a version number. This allows you to track built images and be confident about which version is being started.

Task: Build our static HTML image using the build command below.

```
docker build -t webversion:v1 .
```

You can view a list of all the images on the host using `docker images`.

The built image will have the name *webserver-image* with a tag of *v1*.

```
$ docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
webversion          v1                  4bff7ea17e44        About a minute ago   22.3MB
nginx               alpine              98ab35023fd6        6 days ago          22.3MB
ubuntu              latest              16508e5c265d        2 years ago         84.1MB
redis               latest              4e8db158f18d        2 years ago         83.4MB
weaveworks/scope    1.9.1               4b07159e407b        2 years ago         68MB
alpine              latest              11cd0b38bc3c        2 years ago         4.41MB
$
```

Launch our newly built image providing the friendly name and tag. As it's a web server, bind port 80 to our host using the *-p* parameter.

```
docker run -d -p 80:80 webserver-image:v1
```

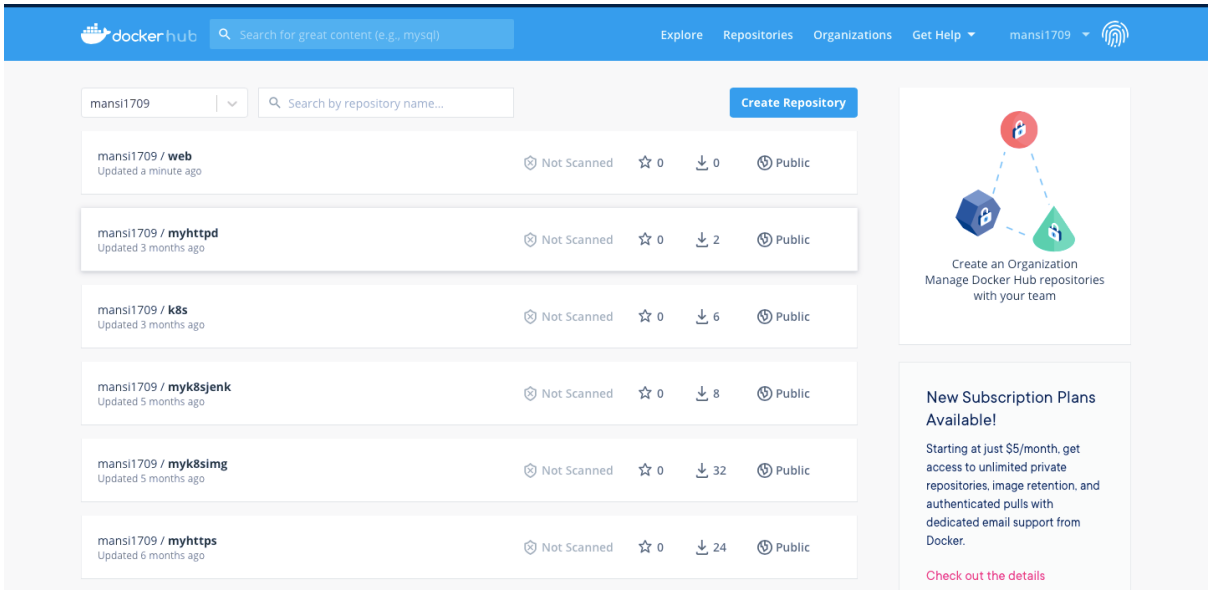
Once started, you'll be able to access the results of port 80 via `curl docker`

Then Provide the credential of dockerhub as mentioned below:-

```
$ docker run -d -p 80:80 webversion:v1
6196aca85e01614edf6119a579a6767484e0b9489ea97b1d9caff196181439f6
$ curl docker
<h1>Hello World</h1>
$ docker login --username mansi1709
Password:
Login Succeeded
$
```

Push to the dockerhub as mentioned below:-

```
$ docker tag webversion:v1 mansi1709/web
$ docker push mansi1709/web
The push refers to repository [docker.io/mansi1709/web]
7248816471b8: Pushed
468af79aab10: Mounted from library/nginx
fbf82c12d86e: Mounted from library/nginx
4dc20fbc0e8d: Mounted from library/nginx
b831cc3ae47e: Mounted from library/nginx
ace0eda3e3be: Mounted from library/nginx
latest: digest: sha256:ad7dd670ab680b8b9157527d12e3529aa795358c0b6ce8bb6fa4516936bca721 size: 1567
```



Download the docker image from remote location if its not available within the system as shown below:-

```
Your Interactive Bash Terminal. A safe place to learn and execute commands.
$
$ docker run -d -p 80:80 mansi1709/web
Unable to find image 'mansi1709/web:latest' locally
latest: Pulling from mansi1709/web
188c0c94c7c5: Already exists
617561f33ec6: Already exists
7d856acdaa9c: Already exists
a0d3c6e28e6d: Already exists
af69a9b963c8: Already exists
89cfa013d8d6: Pull complete
Digest: sha256:ad7dd670ab680b8b9157527d12e3529aa795358c0b6ce8bb6fa4516936bca721
Status: Downloaded newer image for mansi1709/web:latest
3af35b599827b519043f2d7c9ba5fe6b1ac9da0c3aa3c08177c3c9feebf062c0
$
```