Pushing Docker Images to Docker Hub

The given example gives an overview of creating a custom Docker image based on Ubuntu base image and push it to docker hub

1. Download and run Docker image

```
docker run -it --name ubuntu_container ubuntu
```

2. Create a new file inside container

```
touch my_file_bms.txt
```

3. Stop the running container

```
exit
```

4. Check the status of running containers

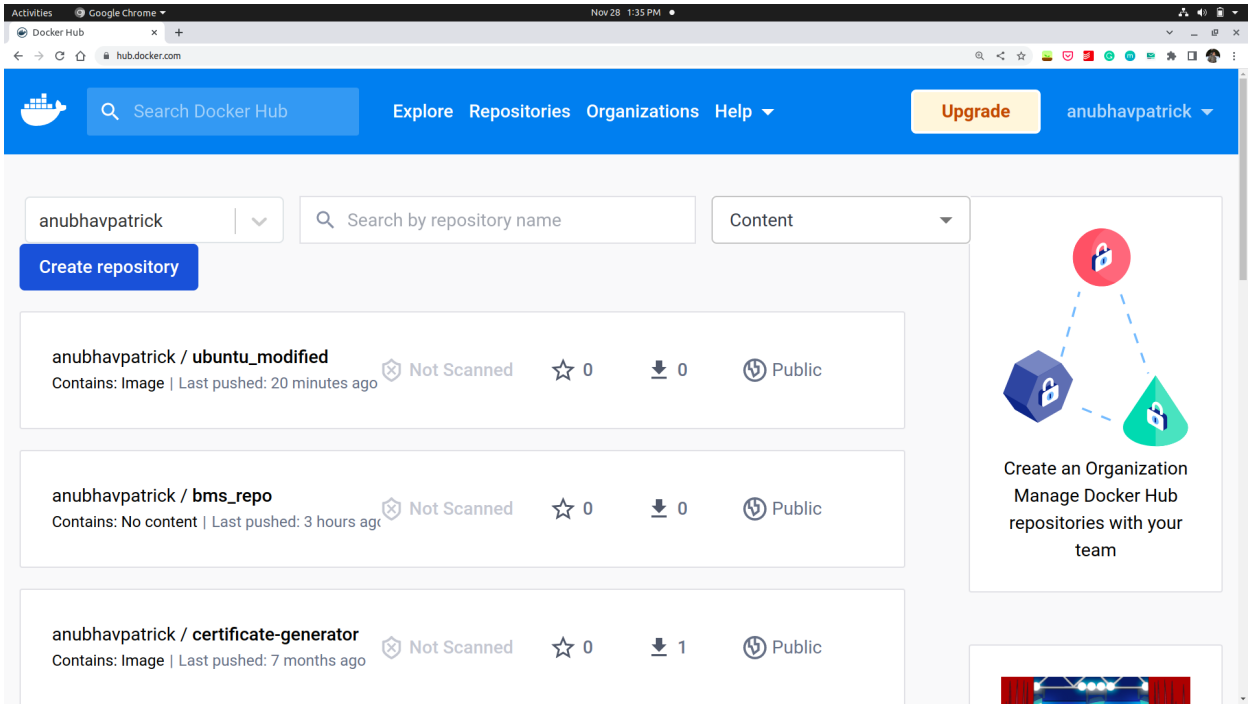
```
docker ps -a
```

```
(env) anubhav@anubhav-HP-Pavilion-x360-Convertible-14-cd0xxx:~$ docker ps -a
CONTAINER ID   IMAGE     COMMAND   CREATED       STATUS                     PORTS     NAMES
2718eff85985   ubuntu    "bash"    3 hours ago   Exited (0) 3 hours ago               ubuntu_container
```

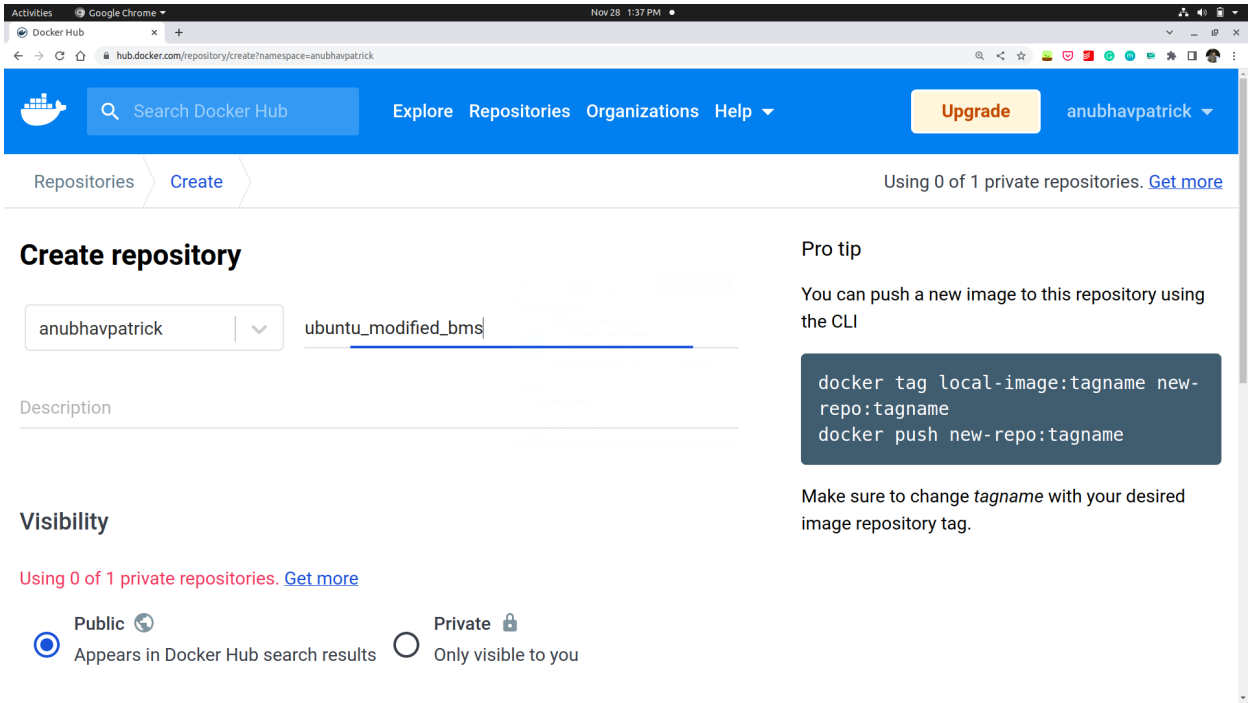
5. Convert the stopped container into image using *commit* command

```
docker commit ubuntu_container ubuntu_modified_image
```

6. Login to *hub.docker.com* and create click *Create repository*



7. Give some relevant name. For example, *ubuntu_modified_bms*



Click *Create* at the bottom

8. Rename your image on your local system as per the guidelines shown

```
docker tag local-image:tagname new-repo:tagname
```

```
docker tag ubuntu_modified_image:latest
anubhavpatrick/ubuntu_modified_bms:latest
```

Explanation:
Here *ubuntu_modified_image* is the name of the custom image we created. Latest is the default tag.
We are creating another name for that image that must match exactly the same name of the repository we created on Docker Hub. In our case it is *anubhavpatrick/ubuntu_modified_bms*. Latest is the tag associated with this image.

For more explanation:
If your docker username is satyaprakash then you have a docker repository titled ubuntu_modified_giindia then rename the image as:

```
docker tag ubuntu_modified_image:latest
satyapraksh/ubuntu_modified_giindia:latest
```

9. Push the updated image to Docker Hub

```
docker push anubhavpatrick/ubuntu_modified_bms:latest
```