

**UNIVERSITY OF PETROLEUM & ENERGY STUDIES**

**Dehradun**

[**Application Containerization Lab**](https://learn.upes.ac.in/webapps/blackboard/execute/launcher?type=Course&id=_49490_1&url=)

**Experiment-9**

**Name: Abhishek Sharma**

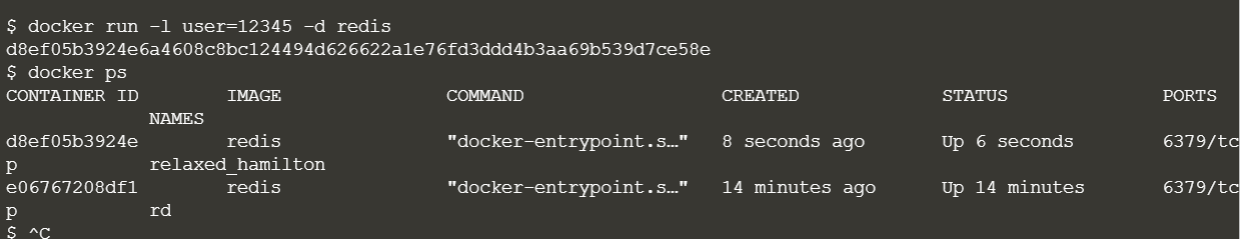
**Course: B. TECH CSE DevOps (2018-22)**

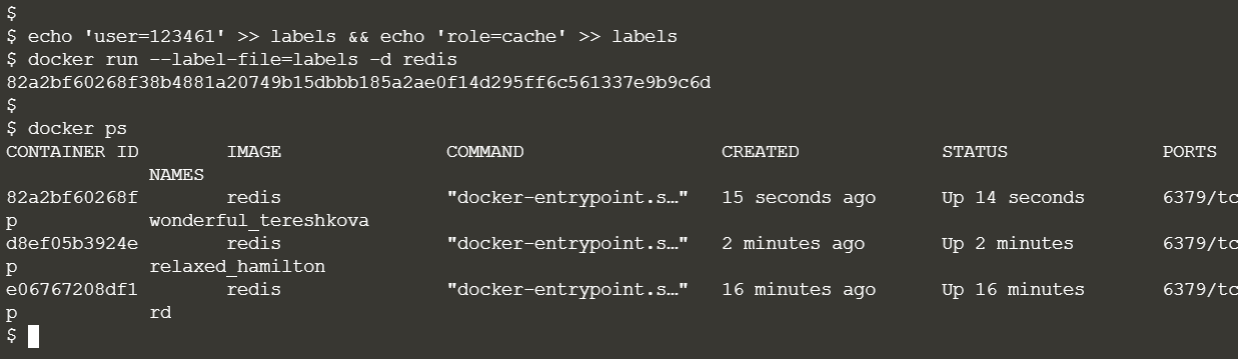
**Roll no.: R171218005**

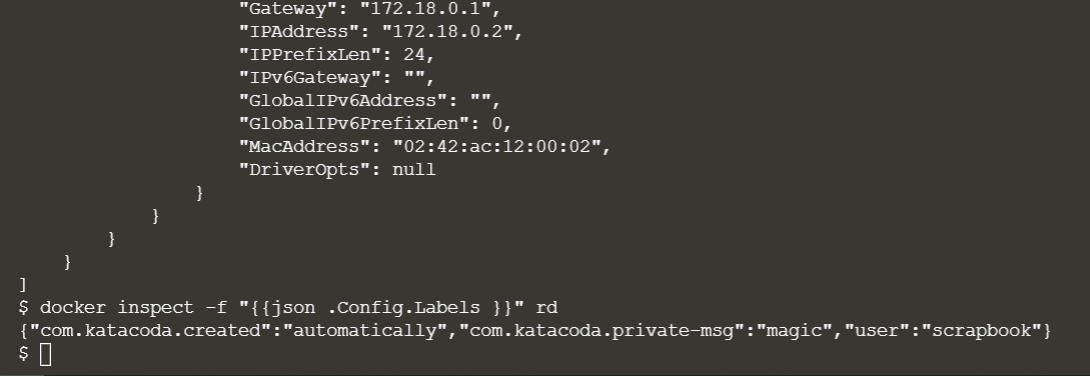
**Sapid: 500067644**

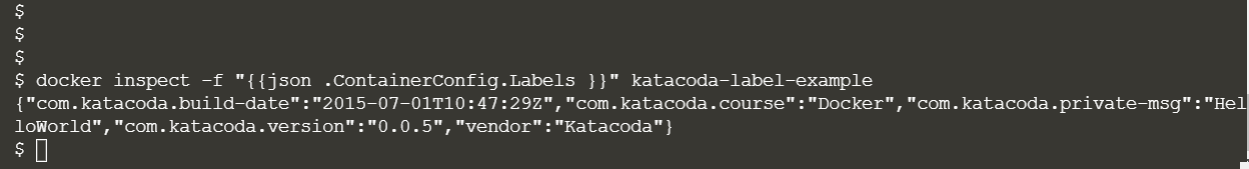
**Aim:** Docker Metadata and labels.

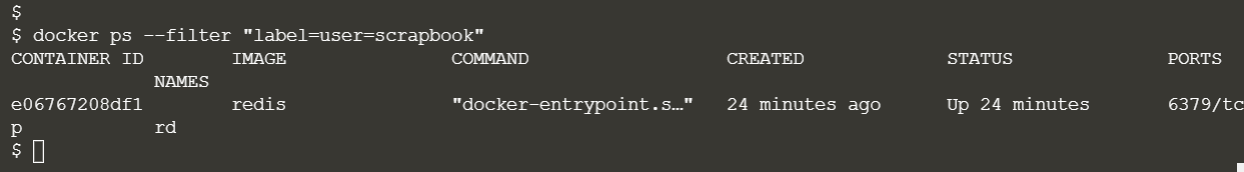
**Procedure:**

**1)** To add a label on a container user option -l while launching container. For example, docker run -l user=12345 -d redis.  


**2)** We can also add labels from external file. For this, use option --label-flie to provide file containing labels.  


**3)** To see all the labels on a running container, use command **docker inspect**. But docker inspect will show other information also. Using the -f option you can filter the JSON response to just the Labels section we're interested in.  


**4)** We can also see labels on images. For this command is **docker inspect -f "{{json .ContainerConfig.Labels }}" katacoda-label-example**.  


**5)** The command **docker ps** allows you to specify a filter based on a label name and value. For example, the query below will return all the containers which have a user label key with the value katacoda.  


**6)** The same filter approach can be applied to images based on the labels used when the image was built.  
