

Modify a running container

There are several ways to modify a running container, to start with suppose we want to make some change to the running container. One way is to get the shell access of the running container and do the changes and exit to save the changes.

```
$ docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
42b099b1674e	nginx	"nginx -g 'daemon of..."	7 minutes ago	Up 7 minutes	0.0.0.0:8081->80/tcp

Use the command `docker exec -it <container name> /bin/bash` to get a bash shell in the container

Generically, use `docker exec -it <container name> <command>` to execute whatever command you specify in the container.

```
$ docker exec -it <container ID> /bin/bash
```

Note that in docker we can use first few letters of a container ID to refer a container and not necessarily type the whole container ID.

42b0 is the first few letter of the container id see above. You can use first few letters of the container id ex. 42 in our case 42b099b1674e. Container ID's are unique and different for all containers.

```
root@42b099b1674e:/#
```

(notice that the shell has changed and you are now logged into the container shell)

```
root@42b099b1674e:/# apt-get update
root@42b099b1674e:/# apt-get install vim
```

```
$ vim /usr/share/nginx/html/index.html
```

Change the line 14 from

Welcome to nginx!

To

Welcome to Docker Labs!

:wq (to save and exit the vim editor)

Now, go to your browser and reload the page, and you will be able to see the changes you made.