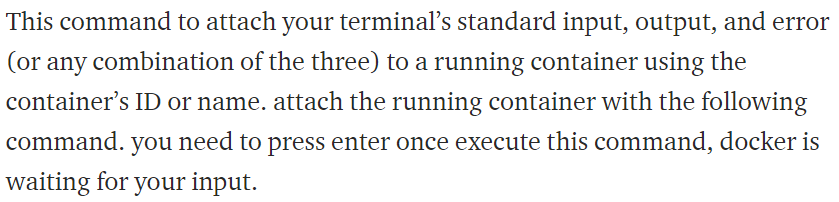
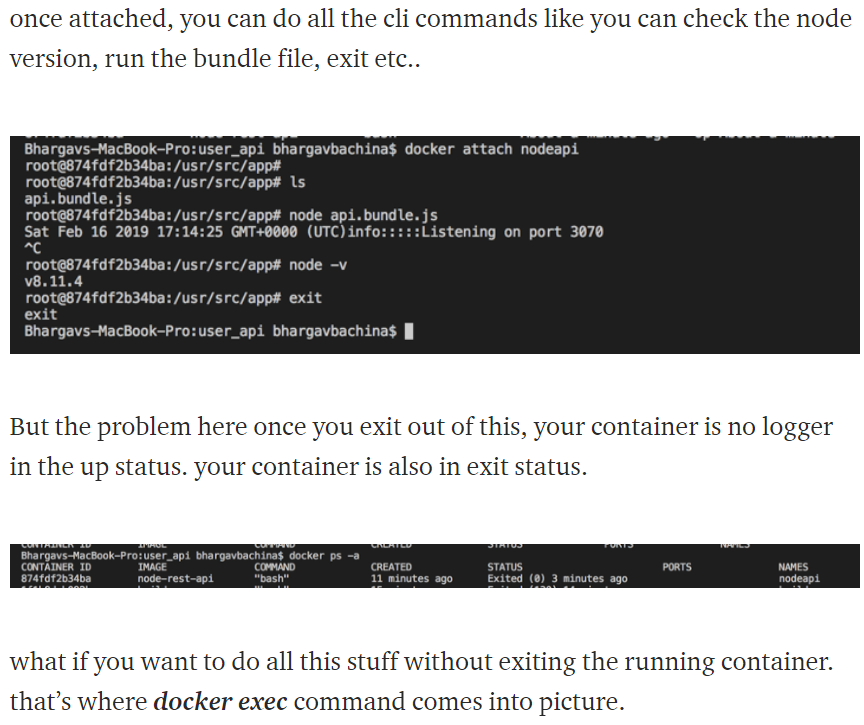
**Attach:** 

* **Docker attach <container ID>**



**Commit:**

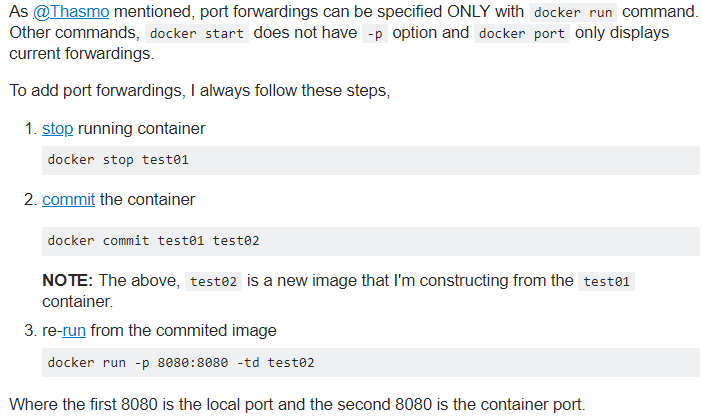
Docker commit command is used to commit a container to an image with the below command.

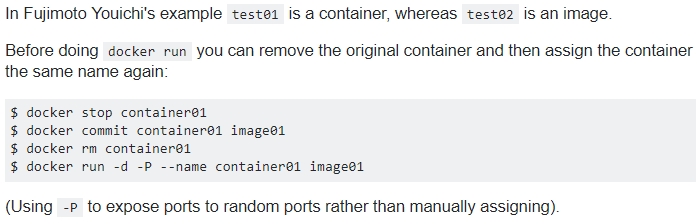
This will be used when we want to change something in running container.

* **Docker commit <container> <image>:<tag>**

**Changing the port of a current container:**

* We can’t change something in running container
* Below is how we can change the port of a current container without losing the data

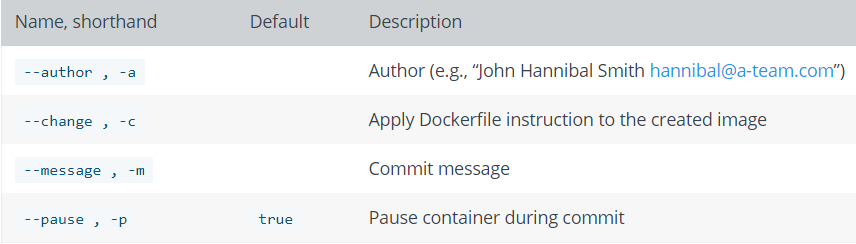




**With these methods, we are able to get the same data from old containers and also with the installations from container**

**The port which we are mapping should match with the port on container, example, we are running container tomcat with port 8080 on server.xml file. Then we can not map it with another port**

Below are the options we can use with commit command.



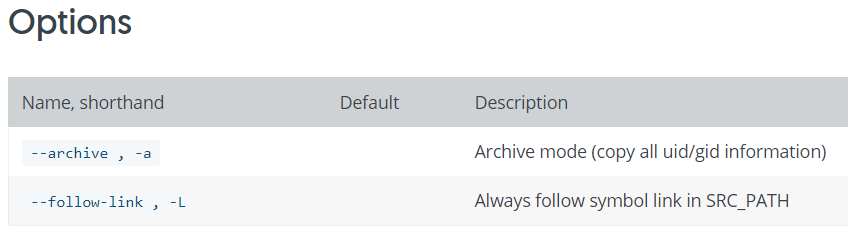
We can use the change option as below.

* **docker commit --change "ENV DEBUG true" c3f279d17e0a svendowideit/testimage:version3**
* **docker commit --change='CMD ["apachectl", "-DFOREGROUND"]' -c "EXPOSE 80" c3f279d17e0a svendowideit/testimage:version4**

**cp:**

copy command is used to copy something from/to the container as below.

* **Docker cp <container>:<path> <local path>**
* **Docker cp <local path> <container>:<path>**



**Tag:**

Docker tag is used to rename the image.

* **Sudo docker tag <old name> <new name>**
* **Sudo docker rmi <old name>**

**Rename:**

Rename command is used to rename the container.

We can name the container by using the - -name flag as below

* **Sudo docker run -d –name container1 -p 8080:8080 tomcat**

And rename the container with below command. We can rename it even if the container is running

* **Sudo docker rename <old name> <new name>**