Docker creating the volumes, backup and restoration

Volumes are useful for backups, restores, and migrations.

Step involved in this process are:

Create a volume

1. Sudo docker create volume ${volume_name}
2. If any specific data are need to copy into the created volume.
   Sudo docker cp ${specific file or as preferred}
3. Create a container with volumes attached
   Sudo docker run -v ${volume_name} –name ${Container_name} ${image_name} /bin/bash

5 We need to backup the containers volume in the docker and in the local system in zip format.
(So that we can verify the data in the local machine and original data is safe in docker volumes.)
Sudo docker run –rm –volumes-form ${container_name} -v $(pwd):/backup ${image_name} tar cvf /backup/backup.tar /${volume_name}.

Creating the volume for data:

$ vim attach.sh (open the file to write a bash script)

#!/bin/bash

Docker run -v task8:/data –name task8 ubuntu /bin/bash

#backup of volume and creating our local window

Docker run –rm –volumes-from task8 -v $(pwd):/backup ubuntu tar cvf /backup/backup.tar /data

```
#!/bin/bash
docker run -v task8:/data --name task8 ubuntu /bin/bash

#backup of volume created into local window

docker run --rm --volumes-from task8 -v $(pwd):/backup ubuntu tar cvf /backup/backup.tar /data
~
```

$ bash ./attach.sh (command to run bash script)

```
bhas@bhas-virtual-machine:~$ sudo bash ./attach.sh
tar: Removing leading `/' from member names
/data/
bhas@bhas-virtual-machine:~$ ls
attach.sh   data     Documents  Music     Public      script.sh  srcript.sh  Templates
backup.tar  Desktop  Downloads  Pictures  script1.sh  snap       taskvol     Videos
bhas@bhas-virtual-machine:~$ sudo docker volume ls
DRIVER    VOLUME NAME
local     task8
bhas@bhas-virtual-machine:~$ sudo docker ps -a
CONTAINER ID   IMAGE     COMMAND       CREATED              STATUS                          PORTS     NAMES
4fe74d494a20   ubuntu    "/bin/bash"   About a minute ago   Exited (0) About a minute ago             task8
```

Docker volumes restore in to new container

1. Create a container with the same volume which is in the backup with new image (container).

Sudo docker run -v ${backup_volume_name} –name ${container_name} /bin/bash

2. So that the same volume is opened in these newly created containers.
3. After the execution again save the volume data into the existing volume file and in the backup file (in local machine).
   Sudo docker run –rm –volumes-from ${container-name} -v $(pwd):/backup ${image_name} bash -c "cd /${backup_volume_name} && tar xvf /backup/backup.tar – strip 1"

==========

Restoring the volume for data:

$ *vim restore.sh*

#!/bin/bash

docker run -v task8:/data –name task10 ubuntu /bin/bash

#restoration of existing volume in new container

Docker run –rm –volumes-from task10 -v $(pwd):/backup ubuntu bash -c "cd /backup && tar cvf /backup/backup.tar –strip 1"

```
#!/bin/bash
docker run -v taak8:/data --name task10 ubuntu /bin/bash

#restoration of existing volume in new container

docker run --rm --volumes-from task10 -v $(pwd):/backup ubuntu bash -c "cd /backup && tar xvf /backup/backup.tar --strip 1"
```

$ sudo bash ./restore.sh (command to run bash script)

```
bhas@bhas-virtual-machine:~$ sudo bash ./restore.sh
bhas@bhas-virtual-machine:~$ sudo docker volume ls
DRIVER    VOLUME NAME
local     e2fd3b61e3fb813838884493150a13c282d536489d481655c0bbf124d63ae754
local     taak8
local     task8
bhas@bhas-virtual-machine:~$ ls
attach.sh   data      Documents  Music     Public      script1.sh  snap        taskvol    Videos
backup.tar  Desktop   Downloads  Pictures  restore.sh  script.sh   srcript.sh  Templates
```