Suppose you have a Docker container running a database that stores important data. You want to ensure that the data is persistent and can survive if the container is stopped or deleted. You decide to use a Docker volume to store the data outside of the container.

Solution:

1. Create a Docker volume by running the command **docker volume create <volume\_name>**. Replace **<volume\_name>** with a name for the new volume that you want to create.
2. Stop the running container by running the command **docker stop <container\_name\_or\_id>**. Replace **<container\_name\_or\_id>** with the name or ID of the running container.
3. Attach the new volume to the container by running the command **docker run -v <volume\_name>:<container\_mount\_point> <image\_name>**. Replace **<volume\_name>** with the name of the new volume that you created in step 1, **<container\_mount\_point>** with the path inside the container where you want to mount the volume, and **<image\_name>** with the name of the image that you want to run.
4. Start the container by running the command **docker start <container\_name\_or\_id>**. Replace **<container\_name\_or\_id>** with the name or ID of the container that you want to start.
5. Verify that the data is stored in the volume by running the command **docker volume inspect <volume\_name>**. You should see the metadata for the new volume, including the mount point and other details. You can also verify that the data is persistent by stopping and starting the container again and verifying that the data is still present.