

To Remove The Running Container By Force:

- *Run Command:* `docker rm -f container-name/container-id`

```
C:\Tests\Docker\Ultimate-Docker-Course-By-Mosh\Tutorial-01\Section-04>docker run -d --name c1 react-app
d8bbefc70097908abec4b8601a68b294668c5174f3020985ceceb6ba7d72dba3

C:\Tests\Docker\Ultimate-Docker-Course-By-Mosh\Tutorial-01\Section-04>docker rm c1
Error response from daemon: cannot remove container "/c1": container is running: stop the container before removing or force remove

C:\Tests\Docker\Ultimate-Docker-Course-By-Mosh\Tutorial-01\Section-04>docker rm -f c1
c1
```

Note (To Remember): To Show The Stopped Containers:

- *Run Command:* `docker ps -a`

Note (To Remember): Alpine Linux Doesn't have `useradd` OR `groupadd`, And For Adding User OR Group We Need `adduser` && `addgroup`

Note (To Remember): We Need To Define The User, Use It, Then Define The Working Directory.

Note (To Remember): When Using Shell Form For *CMD-Instruction* Of Docker File, It Will Use *Shell Process* OR *CMD Process* To Execute The CMD, So For Fasting Building We Need *Exec Form*.

Note (To Remember): The *Latest Image Tag* May Not Point To *Latest Image ID*, We Must Link Them *Explicitly*.

Note (To Remember): To Show The Logs Of Specific Container:

- Run Command: *docker logs container-id-or-some-of-it*

```
75ad55e11e3a728f573bc4453c30ba08ed29f28f3d2a46994b3b1a1361aea56
C:\Users\Tatweer>docker logs 75ad
> react-app@0.1.0 start /app
> react-scripts start

C:\Users\Tatweer>docker logs 75ad
> react-app@0.1.0 start /app
> react-scripts start

@wds@: Project is running at http://172.17.0.2/
@wds@: webpack output is served from
@wds@: Content not from webpack is served from /app/public
@wds@: 404s will fallback to /
Starting the development server...

C:\Users\Tatweer>docker logs -f 75ad
> react-app@0.1.0 start /app
> react-scripts start

@wds@: Project is running at http://172.17.0.2/
@wds@: webpack output is served from
@wds@: Content not from webpack is served from /app/public
@wds@: 404s will fallback to /
Starting the development server...

One of your dependencies, babel-preset-react-app, is importing the
"@babel/plugin-proposal-private-property-in-object" package without
declaring it in its dependencies. This is currently working because
"@babel/plugin-proposal-private-property-in-object" is already in your
node_modules folder for unrelated reasons, but it may break at any time.

babel-preset-react-app is part of the create-react-app project, which
is not maintained anymore. It is thus unlikely that this bug will
ever be fixed. Add "@babel/plugin-proposal-private-property-in-object" to
your devDependencies to work around this error. This will make this message
go away.
```

Activate Windows
Go to Settings to activate Windows.

Note 1 (To Remember): Docker Start-Command Used To Start Stopped Container (No New Process Will Be Created).

Note 2 (To Remember): Docker Run-Command Used To Run New Container (New Process Will Be Created).

Note: To Run The Shell Session From Git-Bash Shell We Use winpty:

winpty docker exec -it f14a30 sh

```
Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Docker/Ultime-Docker-Course-By-Mosh/Tutorial-01/Section-04 (main)
$ winpty docker exec -it f14a30 sh
/app $ echo data > data.txt
/app $ ls
Dockerfile      README.md      data.txt      node_modules  package-lock.json  package.json  public      src
/app $ exit

What's next:
Try Docker Debug for seamless, persistent debugging tools in any container or image - docker debug f14a30
Learn more at https://docs.docker.com/go/debug-cli/

Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Docker/Ultime-Docker-Course-By-Mosh/Tutorial-01/Section-04 (main)
$ winpty docker exec -it e3617 sh
/app $ ls -la | grep -i data
/app $ exit

What's next:
Try Docker Debug for seamless, persistent debugging tools in any container or image - docker debug e3617
Learn more at https://docs.docker.com/go/debug-cli/
```

Note: When We Delete The Container, Also The *FileSystem Of Container* Will Be Deleted, So We Must Store The Data Of Containers Inside Volumes.

To Create New Volume We Can Use: *docker volume create name-here*

```
Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Docker/Ultime-Docker-Course-By-Mosh/Tutorial-01/Section-04 (main)
$ docker volume --help

Usage:  docker volume COMMAND

Manage volumes

Commands:
  create      Create a volume
  inspect     Display detailed information on one or more volumes
  ls          List volumes
  prune       Remove unused local volumes
  rm          Remove one or more volumes

Run 'docker volume COMMAND --help' for more information on a command.

Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Docker/Ultime-Docker-Course-By-Mosh/Tutorial-01/Section-04 (main)
$ docker volume create app-data
app-data

Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Docker/Ultime-Docker-Course-By-Mosh/Tutorial-01/Section-04 (main)
$ ls -al
total 524158
drwxr-xr-x 1 Tatweer 197121      0 Jan 12 14:20 ./
drwxr-xr-x 1 Tatweer 197121      0 Jan 12 12:45 ../
-rw-r--r-- 1 Tatweer 197121    27 Apr 28 09:03 .dockerignore
-rw-r--r-- 1 Tatweer 197121    223 Jan  7 09:38 Dockerfile
-rw-r--r-- 1 Tatweer 197121   3402 Jan 12 13:58 README.md
drwxr-xr-x 1 Tatweer 197121      0 Jan  8 20:16 node_modules/
-rw-r--r-- 1 Tatweer 197121   931674 Jan  6 14:38 package-lock.json
-rw-r--r-- 1 Tatweer 197121    813 Mar  5 2021 package.json
drwxr-xr-x 1 Tatweer 197121      0 Jan  8 20:18 public/
-rw-r--r-- 1 Tatweer 197121 534929408 Jan 12 14:20 react-app.tar
-rw-r--r-- 1 Tatweer 197121   522007 Mar 16 2021 section4-react-app.zip
drwxr-xr-x 1 Tatweer 197121      0 Jan  8 20:18 src/
```

For Inspecting The Created Volume, We Can Run: *docker volume inspect volume-name-here*

```
Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Docker/Ultimate-Docker-Course-By-Mosh (main)
$ docker volume inspect app-data
[
  {
    "CreatedAt": "2025-05-27T07:29:19Z",
    "Driver": "local",
    "Labels": null,
    "Mountpoint": "/var/lib/docker/volumes/app-data/_data",
    "Name": "app-data",
    "Options": null,
    "Scope": "local"
  }
]
```

To Attach The Volume With Container:

- If The Volume Doesn't Exist, Then Docker Will Create It

```
Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Docker/Ultimate-Docker-Course-By-Mosh (main)
$ docker run -d --name jafar-loka-01 -v app-data:/app/data react-app
828d8b1882c99177e7c6eedc28fdf044bf1a4120dd74b8eb48c022eb1c5a706e

Tatweer@DESKTOP-9FV4RTN MINGW64 /c/Tests/Docker/Ultimate-Docker-Course-By-Mosh (main)
$ winpty docker exec -it jafar-loka-01 sh
/app $ ls
Dockerfile      README.md      data           node_modules   package-lock.json  package.json    public         src
/app $ |
```

If We Want To Create Any File Inside The *Data-Directory*, Then We Will Get Permission Denied:

Note: The Reason About That, Is That Docker Will Create Data-Directory With *Root-Permission If Not Exists*.

```
/app/data $ echo data > data.txt
sh: can't create data.txt: Permission denied
/app/data $ nano data.txt
sh: nano: not found
/app/data $ touch data.txt
touch: data.txt: Permission denied
/app/data $ cd ..
/app $ ls -la
total 956
drwxr-xr-x  1 app      app      4096 May 28 06:31 .
drwxr-xr-x  1 root    root     4096 May 28 06:31 ..
-rwxr-xr-x  1 root    root       20 Jan  6 12:16 .dockerignore
-rwxr-xr-x  1 root    root     223 Jan  7 06:38 Dockerfile
-rwxr-xr-x  1 root    root    3402 Jan 12 10:58 README.md
drwxr-xr-x  2 root    root     4096 May 27 07:29 data
drwxr-xr-x  1 app      app     4096 May 28 06:32 node_modules
-rwxr-xr-x  1 root    root  931674 Jan  6 11:38 package-lock.json
-rwxr-xr-x  1 root    root   813 Mar  5 2021 package.json
drwxr-xr-x  2 root    root     4096 Mar  9 2021 public
drwxr-xr-x  2 root    root     4096 Mar  9 2021 src
/app $ |
```

To Solve The Above Problem, We Need To Create The Data-Directory From Docker-File.

```
FROM node:14.16.0-alpine3.13
RUN addgroup app && adduser -S -G app app
USER app
WORKDIR /app
COPY package*.json .
RUN npm i
RUN ["mkdir", "data"]
COPY . .
ENV API_URL=https://jafar.loka.com/
EXPOSE 3000
# Exec Form
CMD ["npm", "start"]
```

```
C:\Tests\Docker\Ultimate-Docker-Course-By-Mosh\Tutorial-01\Section-04>docker exec -it jafar-loka-01 sh
/app $ cd data
/app/data $ ls
/app/data $ echo data > data.txt
/app/data $ ls
data.txt
/app/data $ cat data.txt
data
/app/data $
```

Note: The Benefit Of Using This Way To Create Volumes, Is That: We Can Share Volumes Between Multiple Containers.

Note (From Me): We Can Use It As Simple Shared Memory Between *Microservices Applications*.

To Copy File From Container To Current Directory (From Container To Host)

Run Command: *docker cp container-id:/path/to/file .*

```
C:\Tests\Docker\Ultimate-Docker-Course-By-Mosh\Tutorial-01\Section-04>docker cp 65e:/app/log.txt .
Successfully copied 2.05kB to C:\Tests\Docker\Ultimate-Docker-Course-By-Mosh\Tutorial-01\Section-04\.
```

```
C:\Tests\Docker\Ultimate-Docker-Course-By-Mosh\Tutorial-01\Section-04>ls
'ls' is not recognized as an internal or external command,
operable program or batch file.
```

```
C:\Tests\Docker\Ultimate-Docker-Course-By-Mosh\Tutorial-01\Section-04>dir
Volume in drive C has no label.
Volume Serial Number is 8A24-F060

Directory of C:\Tests\Docker\Ultimate-Docker-Course-By-Mosh\Tutorial-01\Section-04

05/28/2025  09:52 AM    <DIR>          .
05/28/2025  09:52 AM    <DIR>          ..
04/28/2025  09:03 AM                27  .dockerignore
05/28/2025  09:43 AM               246  Dockerfile
05/28/2025  09:51 AM                6  log.txt
01/08/2025  08:16 PM    <DIR>          node_modules
01/06/2025  02:38 PM          931,674  package-lock.json
03/05/2021  10:00 PM               813  package.json
01/08/2025  08:18 PM    <DIR>          public
01/12/2025  02:20 PM        534,929,408  react-app.tar
01/12/2025  01:58 PM           3,402  README.md
03/16/2021  07:35 PM          522,007  section4-react-app.zip
01/08/2025  08:18 PM    <DIR>          src
               8 File(s)      536,387,583 bytes
               5 Dir(s)      292,482,519,040 bytes free
```

Also, We Can Copy The Files From Host To Container Using The Same Command:

```
C:\Tests\Docker\Ultimate-Docker-Course-By-Mosh\Tutorial-01\Section-04>docker cp secret.txt 65e:/app
Successfully copied 2.05kB to 65e:/app
```

```
C:\Tests\Docker\Ultimate-Docker-Course-By-Mosh\Tutorial-01\Section-04>docker exec -it 65e sh
/app $ ls
Dockerfile      data            node_modules    package.json     secret.txt
README.md       log.txt         package-lock.json public            src
/app $ ls -la
total 964
drwxr-xr-x  1 app      app           4096 May 28 06:56 .
drwxr-xr-x  1 root    root          4096 May 28 06:56 ..
-rwxr-xr-x  1 root    root           27 Apr 28 06:03 .dockerignore
-rwxr-xr-x  1 root    root          246 May 28 06:39 Dockerfile
-rwxr-xr-x  1 root    root         3402 Jan 12 10:58 README.md
drwxr-xr-x  2 app      app           4096 May 28 06:46 data
-rw-r--r--  1 app      app            6 May 28 06:51 log.txt
drwxr-xr-x  1 app      app           4096 May 28 06:46 node_modules
-rwxr-xr-x  1 root    root        931674 Jan  6 11:38 package-lock.json
-rwxr-xr-x  1 root    root           813 Mar  5 2021 package.json
drwxr-xr-x  2 root    root           4096 Mar  9 2021 public
-rwxr-xr-x  1 root    root            5 May 28 06:56 secret.txt
drwxr-xr-x  2 root    root           4096 Mar  9 2021 src
/app $
```

For Linux/Mac OS we can clear our working space using:

```
~/Dropbox/Courses/Docker/Demo  
> ✓ docker image rm $(docker image ls -q)
```

```
C:\Users\Tatweer>docker image ls -q  
0bc9d28e6541  
5b31ac1e45b0  
56ec78ea88b7  
d21aa96c9304  
6c6326135b66  
1b97297862b7  
850d1f628e28  
a60dd8dae2ad  
e9adc5c075a8  
38406042cf08  
25903461e65c  
71161e05b9bb  
c69fa2e9cbf5  
2e96e5913fc0  
3861cfcd7c04  
556098075b3d  
e6f181688397  
99f89471f470
```

The same for containers:

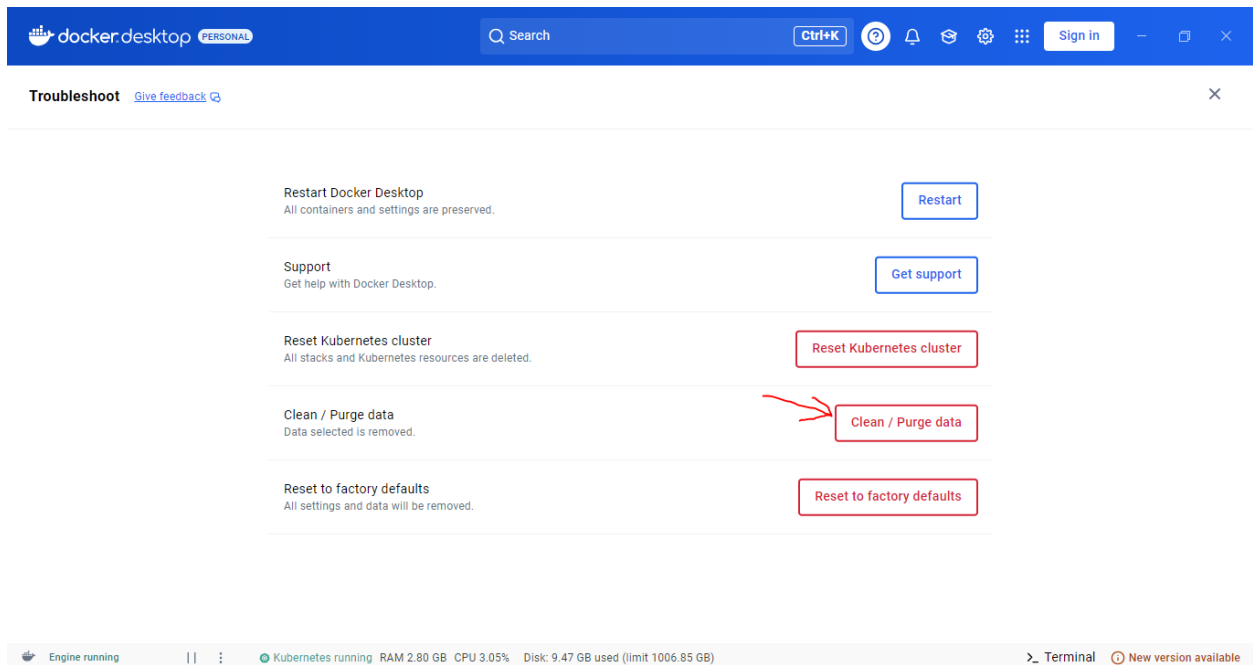
Note: Here we set `-a` for stopped container also

```
~/Dropbox/Courses/Docker/Demo  
> ✓ docker container rm $(docker container ls -aq)
```

If we have running containers, then we need to supply `-f`:

```
~/Dropbox/Courses/Docker/Demo  
1 docker container rm -f (docker container ls -aq)
```

For cleaning all data from Troubleshooting Of Docker Desktop:



To indicate the start of .yaml OR .yml Files, we set: ---

The *standard name for docker compose file*, is: ***docker-compose.yml***

In this way, we represented array in *yaml-files*:

```
tags:
  - software
  - devops
```

In this way, we represented objects in *yaml-files*:

```
author:
  first_name: Mosh
  last_name: Hamedani
```
