

# Docker Main Command Tutorial

郭益華

# 目錄

1. [簡介說明](#)
2. [下載 image](#)
3. [執行多個container](#)
4. [查看container運行狀態的資訊](#)

# 1. 簡介說明

# 說明

- 以docker常用重要指令為主
- 以 dockerhub 的 redis image為例，進行docker指令實際操作

用到的指令:

1. `docker pull <images name>`: 下載 images
2. `docker run <images name>`: 啟動新的 container & 直接 pull images 後啟動 container
3. `docker run -d`: 啟動另一個新的容器(d就是分離的意思)
4. `docker run -p<localPort>:<containerPort> <image>` : 綁定 image 端口
5. `docker run -p<localPort>:<containerPort> <named> <image>`: 綁定 image 端口並為 container 命名
6. `docker start <id>`: 指定啟動停止的 container
7. `docker stop <id>`: 指定停止運行中的 container
8. `docker ps`: 列出當前運行中的 container
9. `docker ps -a`: 列出當前所有存在的 container (不論是否運行中)
10. `docker logs <Container ID or Names>`: 查看 container 的狀態資訊
11. `docker exec -it <Container ID>`: 與 container 終端機互動

## 2. 下載image

# 至 dockerhub 搜尋 redis

The screenshot shows the Docker Hub search results for 'redis'. The search bar at the top is highlighted with a red box. The results list the 'redis' Docker Official Image, which is also highlighted with a red box. The image details include its name, 'Docker Official Image' status, download count (1B+), star count (10K+), and update time (19 hours ago). It also provides a description of Redis as an open source key-value store and lists supported architectures: Linux, unknown, Windows, ARM 64, 386, mips64le, PowerPC 64 LE, IBM Z, unknown, x86-64, and ARM. To the right of the image details, the pull count (25,952,282) and a line graph showing pull activity over time are visible. A 'Learn more' link is also present.

docker hub

redis

Explore Repositories Organizations Help

Upgrade

jerry7776112

Filters

1 - 25 of 10,000 results for redis.

Best Match

Products

- ☐ Images
- ☐ Extensions
- ☐ Plugins

Trusted Content

- ☐ Docker Official Image

redis Docker Official Image · 1B+ · 10K+ · Updated 19 hours ago

Redis is an open source key-value store that functions as a data structure server.

Linux unknown Windows ARM 64 386 mips64le PowerPC 64 LE IBM Z unknown x86-64 ARM

Pulls: 25,952,282 Last week

Learn more

會看到 Docker Official Image  
即為 Docker 官方所提供的 Redis 映像檔，點選即可



# redis

Docker Official Image • 1B+ • 10K+

Redis is an open source key-value store that functions as a data structure server.

```
docker pull redis
```



Overview

Tags

## Quick reference

- Maintained by:  
[the Docker Community](#)
- Where to get help:

### Recent Tags

latest bookworm 7.2.1-bookworm 7.2.1  
7.2-bookworm 7.2 7.0.13-bookworm 7.0.13  
7.0-bookworm 7.0

此為下載Redis映像檔的指令

# 往下拉可以看到其他版本的Redis映像檔

## Supported tags and respective Dockerfile links

- `7.2.1`, `7.2`, `7`, `latest`, `7.2.1-bookworm`, `7.2-bookworm`, `7-bookworm`, `bookworm`
- `7.2.1-alpine`, `7.2-alpine`, `7-alpine`, `alpine`, `7.2.1-alpine3.18`, `7.2-alpine3.18`, `7-alpine3.18`, `alpine3.18`
- `7.0.13`, `7.0`, `7.0.13-bookworm`, `7.0-bookworm`
- `7.0.13-alpine`, `7.0-alpine`, `7.0.13-alpine3.18`, `7.0-alpine3.18`
- `6.2.13`, `6.2`, `6`, `6.2.13-bookworm`, `6.2-bookworm`, `6-bookworm`
- `6.2.13-alpine`, `6.2-alpine`, `6-alpine`, `6.2.13-alpine3.18`, `6.2-alpine3.18`, `6-alpine3.18`
- `6.0.20`, `6.0`, `6.0.20-bookworm`, `6.0-bookworm`
- `6.0.20-alpine`, `6.0-alpine`, `6.0.20-alpine3.18`, `6.0-alpine3.18`



# 下載 Redis 的 image 至 Local端

```
PS C:\Users\jerry> docker pull redis
Using default tag: latest
latest: Pulling from library/redis
a378f10b3218: Pull complete
3c75410c1f8b: Pull complete
667874757cc1: Pull complete
7150b93d249d: Pull complete
ed7c83735e28: Pull complete
4f4fb700ef54: Pull complete
3fa899a007ab: Pull complete
Digest: sha256:55f70c57c74b25d00c38ff60726da1caee17202774af2de365c36628d9b3c7e1
Status: Downloaded newer image for redis:latest
docker.io/library/redis:latest
PS C:\Users\jerry>
```

latest為最新版本

Command:  
docker pull <image name>

# 下載完畢查看當前所存在的image

可看到已經有 redis 的 image 了

Command:  
docker images

```
PS C:\Users\jerry> docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
jupyter/pyspark-notebook	latest	b4062e157cea	3 weeks ago	4.91GB
my-flask-app	latest	3af02193fbc7	3 weeks ago	1.09GB
redis	latest	5b0542ad1e77	5 weeks ago	138MB
sqlc/sqlc	latest	e5c3b6c57c08	5 weeks ago	98.2MB
confluentinc/cp-zookeeper	latest	04bd40128a4e	7 weeks ago	849MB
postgres	12-alpine	57609cf5d1df	2 months ago	230MB
postgres	14-alpine	955d825eef13	2 months ago	235MB
postgres	9.6	027ccf656dc1	20 months ago	200MB
confluentinc/cp-kafka	5.3.1	3ce94ccaeaea	24 months ago	590MB

# 將images啟動並運行於container中

Command:  
docker run <image name>

```
PS C:\Users\jerry> docker run redis
1:C 12 Oct 2023 01:57:44.379 # WARNING Memory overcommit must be enabled! Without
ay fail under low memory condition. Being disabled, it can also cause failures wi
/github.com/jemalloc/jemalloc/issues/1328. To fix this issue add 'vm.overcommit_m
reboot or run the command 'sysctl vm.overcommit_memory=1' for this to take effec
1:C 12 Oct 2023 01:57:44.379 * oO00oO00oO00o Redis is starting oO00oO00oO00o
1:C 12 Oct 2023 01:57:44.379 * Redis version=7.2.1, bits=64, commit=00000000, mod
1:C 12 Oct 2023 01:57:44.379 # Warning: no config file specified, using the defau
file use redis-server /path/to/redis.conf
1:M 12 Oct 2023 01:57:44.380 * monotonic clock: POSIX clock_gettime
1:M 12 Oct 2023 01:57:44.380 * Running mode=standalone, port=6379.
1:M 12 Oct 2023 01:57:44.380 * Server initialized
1:M 12 Oct 2023 01:57:44.380 * Ready to accept connections tcp
```

# 查看當前運行中的 container

Command:  
docker ps

```
PS C:\Users\jerry> docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
b375b30b6b83	redis	"docker-entrypoint.s..."	About a minute ago	Up About a minute	6379/tcp	cranky_tharp

```
PS C:\Users\jerry>
```

可看到 redis 已啟動

# 停止當前運行的container

Command:  
docker stop <container ID or name>

```
PS C:\Users\jerry> docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS        NAMES
b375b30b6b83   redis     "docker-entrypoint.s..." 4 minutes ago  Up 4 minutes  6379/tcp     cranky_tharp
PS C:\Users\jerry> docker stop b375b30b6b83
b375b30b6b83
PS C:\Users\jerry> docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS        NAMES
```

可看到已經沒有運行的container

# 再啟動另外一個新的 redis

Command:

`docker run -d <image name>`

d: 分離的意思

```
PS C:\Users\jerry> docker run -d redis
89812296eb10d723907a01639fcf097d3e7d0408aa2556882f24a209299fab48
```

會顯示一個新的id，可看到此id與前面一個不一樣

```
PS C:\Users\jerry> docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS          NAMES
89812296eb10   redis    "docker-entrypoint.s..." 14 seconds ago Up 14 seconds  6379/tcp      inspiring_mendel
```

# 查看所有存在的container，不論是否有啟動

```
PS C:\Users\jerry> docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED    STATUS    PORTS    NAMES
PS C:\Users\jerry> docker start 89812296eb10
89812296eb10
PS C:\Users\jerry> docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED    STATUS    PORTS    NAMES
89812296eb10   redis    "docker-entrypoint.s..." 6 minutes ago    Up 3 seconds    6379/tcp    inspiring_mendel
```

Command:  
docker ps -a

a: 所有、全部的意思

```
PS C:\Users\jerry> docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED    STATUS    PORTS    NAMES
89812296eb10   redis    "docker-entrypoint.s..." 8 minutes ago    Exited (0) About a minute ago    inspiring_mendel
b375b30b6b83   redis    "docker-entrypoint.s..." 36 minutes ago    Exited (0) 31 minutes ago
```

可看到所有存在的container

### 3. 執行多個container



# 啟動其他版本的 redis

Command:  
docker run <image name>:<version number>

```
PS C:\Users\jerry> docker run redis:7.0.13
Unable to find image 'redis:7.0.13' locally
7.0.13: Pulling from library/redis
a378f10b3218: Already exists
1692ede27ff4: Pull complete
086dbb1beab5: Pull complete
7c46c2447fd7: Pull complete
88e07738ded1: Pull complete
4f4fb700ef54: Pull complete
1fd6ec2bc98a: Pull complete
Digest: sha256:63a4a2a437b885ce47c87a760fbbf4b3c45f2bfdceef0f6792f5b663be79549
Status: Downloaded newer image for redis:7.0.13
1:C 12 Oct 2023 02:38:37.590 # o00o000o000o Redis is starting o00o000o000o
1:C 12 Oct 2023 02:38:37.591 # Redis version=7.0.13, bits=64, commit=00000000, modified=0, pid=1, just started
1:C 12 Oct 2023 02:38:37.591 # Warning: no config file specified, using the default config. In order to specify a config
file use redis-server /path/to/redis.conf
1:M 12 Oct 2023 02:38:37.591 * monotonic clock: POSIX clock_gettime
1:M 12 Oct 2023 02:38:37.591 * Running mode=standalone, port=6379.
1:M 12 Oct 2023 02:38:37.591 # Server initialized
1:M 12 Oct 2023 02:38:37.591 # WARNING Memory overcommit must be enabled! Without it, a background save or replication m
ay fail under low memory condition. Being disabled, it can also cause failures without low memory condition, see htt
ps://github.com/jemalloc/jemalloc/issues/1328. To fix this issue add 'vm.overcommit_memory = 1' to /etc/sysctl.conf and
then reboot or run the command 'sysctl vm.overcommit_memory=1' for this to take effect.
1:M 12 Oct 2023 02:38:37.592 * Ready to accept connections
```

直接使用run指令會自動偵測local是否有該版本的image，沒有的話會自動pull，pull結束會自動運行於container中。

# 同時運行兩個不同版本的redis

Command:

`docker start <container name or id>`

start: 重新啟動剛剛所stop的container

```
PS C:\Users\jerry> docker start 89812296eb10
89812296eb10
PS C:\Users\jerry> docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
cc76e7154f49	redis:7.0.13	"docker-entrypoint.s..."	About a minute ago	Up About a minute	6379/tcp	focused_matsu
89812296eb10	redis	"docker-entrypoint.s..."	14 minutes ago	Up 1 second	6379/tcp	inspiring_men

```
del
```

# run一個新的 redis container 同時設定 port

Command:

```
docker run -p<localPort>:<containerPort> <image>
```

p: port的意思

```
PS C:\Users\jerry> docker run -p6000:6379 redis
1:C 12 Oct 2023 02:52:17.718 # WARNING Memory overcommit must be enabled! Without it, a background save or replication may fail under low memory condition. Being disabled, it can also cause failures without low memory condition, see https://github.com/jemalloc/jemalloc/issues/1328. To fix this issue add 'vm.overcommit_memory = 1' to /etc/sysctl.conf and then reboot or run the command 'sysctl vm.overcommit_memory=1' for this to take effect.
1:C 12 Oct 2023 02:52:17.718 * o000o000o000o Redis is starting o000o000o000o
1:C 12 Oct 2023 02:52:17.718 * Redis version=7.2.1, bits=64, commit=00000000, modified=0, pid=1, just started
1:C 12 Oct 2023 02:52:17.718 # Warning: no config file specified, using the default config. In order to specify a config file use redis-server /path/to/redis.conf
1:M 12 Oct 2023 02:52:17.719 * monotonic clock: POSIX clock_gettime
1:M 12 Oct 2023 02:52:17.719 * Running mode=standalone, port=6379.
1:M 12 Oct 2023 02:52:17.719 * Server initialized
1:M 12 Oct 2023 02:52:17.719 * Ready to accept connections tcp
```

# 可觀察到port以綁定

```
PS C:\Users\jerry> docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
79dec855a10d	redis	"docker-entrypoint.s..."	59 seconds ago	Up 59 seconds	0.0.0.0:6000->6379/tcp	gallant
_hofstadter						
cc76e7154f49	redis:7.0.13	"docker-entrypoint.s..."	14 minutes ago	Up 14 minutes	6379/tcp	focused
_matsumoto						

```
PS C:\Users\jerry> docker run -p6000:6379 redis:7.0.13
docker: Error response from daemon: driver failed programming external connectivity on endpoint kind_ganguly (c83ddad86e922af4770dd5e41039c5d319a89665e40cb26e17dd84ba7ac7493f): Bind for 0.0.0.0:6000 failed: port is already allocated.
```

run另外一個redis並設定port，可發現到有error，因為port 6000 前面已經綁訂了

# 更改port即可除錯

```
PS C:\Users\jerry> docker run -p6000:6379 redis:7.0.13
docker: Error response from daemon: driver failed programming external connectivity on endpoint kind_ganguly (c83ddad86e922af4770dd5e41039c5d319a89665e40cb26e17dd84ba7ac7493f): Bind for 0.0.0.0:6000 failed: port is already allocated.
PS C:\Users\jerry> docker run -p6001:6379 redis:7.0.13
1:C 12 Oct 2023 02:56:57.391 # o000o000o000o Redis is starting o000o000o000o
1:C 12 Oct 2023 02:56:57.391 # Redis version=7.0.13, bits=64, commit=00000000, modified=0, pid=1, just started
1:C 12 Oct 2023 02:56:57.391 # Warning: no config file specified, using the default config. In order to specify a config file use redis-server /path/to/redis.conf
1:M 12 Oct 2023 02:56:57.391 * monotonic clock: POSIX clock_gettime
1:M 12 Oct 2023 02:56:57.392 * Running mode=standalone, port=6379.
1:M 12 Oct 2023 02:56:57.392 # Server initialized
1:M 12 Oct 2023 02:56:57.392 # WARNING Memory overcommit must be enabled! Without it, a background save or replication may fail under low memory condition. Being disabled, it can also cause failures without low memory condition, see https://github.com/jemalloc/jemalloc/issues/1328. To fix this issue add 'vm.overcommit_memory = 1' to /etc/sysctl.conf and then reboot or run the command 'sysctl vm.overcommit_memory=1' for this to take effect.
1:M 12 Oct 2023 02:56:57.392 * Ready to accept connections
```

將-p6000 改為 -p6001 即可

# 可看到兩個設定port的container都成功運行

```
PS C:\Users\jerry> docker ps
CONTAINER ID   IMAGE      COMMAND                  CREATED        STATUS        PORTS                               NAMES
0a76ae0e1d07   redis:7.0.13 "docker-entrypoint.s..." 54 seconds ago Up 53 seconds 0.0.0.0:6001->6379/tcp             vigorous_ishizaka
79dec855a10d   redis      "docker-entrypoint.s..." 5 minutes ago  Up 5 minutes  0.0.0.0:6000->6379/tcp             gallant_hofstadter
PS C:\Users\jerry>
```

## 4. 查看container運行狀態 的資訊

# 查看container狀態資訊

Command:  
docker logs <container id>

```
PS C:\Users\jerry> docker logs 0a76ae0e1d07
1:C 12 Oct 2023 02:56:57.391 # o000o000o000o Redis is starting o000o000o000o
1:C 12 Oct 2023 02:56:57.391 # Redis version=7.0.13, bits=64, commit=00000000, modified=0, pid=1, just started
1:C 12 Oct 2023 02:56:57.391 # Warning: no config file specified, using the default config. In order to specify a config file use red
is-server /path/to/redis.conf
1:M 12 Oct 2023 02:56:57.391 * monotonic clock: POSIX clock_gettime
1:M 12 Oct 2023 02:56:57.392 * Running mode=standalone, port=6379.
1:M 12 Oct 2023 02:56:57.392 # Server initialized
1:M 12 Oct 2023 02:56:57.392 # WARNING Memory overcommit must be enabled! Without it, a background save or replication may fail under
low memory condition. Being disabled, it can can also cause failures without low memory condition, see https://github.com/jemalloc/j
emalloc/issues/1328. To fix this issue add 'vm.overcommit_memory = 1' to /etc/sysctl.conf and then reboot or run the command 'sysctl
vm.overcommit_memory=1' for this to take effect.
1:M 12 Oct 2023 02:56:57.392 * Ready to accept connections
1:signal-handler (1697079613) Received SIGTERM scheduling shutdown...
1:M 12 Oct 2023 03:00:13.366 # User requested shutdown...
1:M 12 Oct 2023 03:00:13.367 * Saving the final RDB snapshot before exiting.
1:M 12 Oct 2023 03:00:13.377 * DB saved on disk
1:M 12 Oct 2023 03:00:13.377 # Redis is now ready to exit, bye...
```



# 啟動新容器設定port並為container命名

Command:

```
docker run -d -p<localPort>:<containerPort> <named> <image>
```

```
PS C:\Users\jerry> docker run -d -p6001:6379 --name redis-older redis:7.0.13  
0f509c84fe6b0a1bb06394c4f9e37cc255869dc9ae5056d87f9aa02be44b5e86
```

```
PS C:\Users\jerry> docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
0f509c84fe6b	redis:7.0.13	"docker-entrypoint.s..."	5 seconds ago	Up 4 seconds	0.0.0.0:6001->6379/tcp	redis-older

```
PS C:\Users\jerry> docker stop redis-older  
redis-older
```

```
PS C:\Users\jerry> docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
--------------	-------	---------	---------	--------	-------	-------

# 進入container內並操作指令

每個container都是建立於微型的Linux環境中，可使用相關指令進入container內進行操作

Command: `docker exec -it <container id> /bin/bash` -it: 與container的終端機互動

```
PS C:\Users\jerry> docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                    NAMES
0f509c84fe6b   redis:7.0.13   "docker-entrypoint.s..." 3 minutes ago  Up 15 seconds  0.0.0.0:6001->6379/tcp    redis-older
PS C:\Users\jerry> docker exec -it 0f509c84fe6b /bin/bash
root@0f509c84fe6b:/data# ls
dump.rdb
root@0f509c84fe6b:/data# pwd
/data
root@0f509c84fe6b:/data# cd /
root@0f509c84fe6b:/# ls
bin boot data dev etc home lib lib32 lib64 libx32 media mnt opt proc root run sbin srv sys tmp usr var
root@0f509c84fe6b:/# env
HOSTNAME=0f509c84fe6b
REDIS_DOWNLOAD_SHA=97065774d5fb8388eb0d8913458decfcb167d356e40d31dd01cd30c1cc391673
PWD=/
HOME=/root
REDIS_VERSION=7.0.13
GOSU_VERSION=1.16
TERM=xterm
REDIS_DOWNLOAD_URL=http://download.redis.io/releases/redis-7.0.13.tar.gz
SHLVL=1
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
_=/usr/bin/env
OLDPWD=/data
root@0f509c84fe6b:/# exit
exit
PS C:\Users\jerry>
```

離開container終端機輸入: exit

**End**