

WSL (Windows Subsystem for Linux)

Wsl slouží k instalaci linuxových distribucí na Windows.

Umožňuje spouštět linuxové aplikace přímo na Windows bez potřeby virtuálního stroje.

Instalace

► [Windows](#)

Docker

Instalace

⊗ Important

Pro instalaci Dockeru na Windows potřebujete mít nainstalovaný WSL (Windows Subsystem for Linux).

Návod na instalaci WSL najdete v sekci [WSL](#).

Základní informace

- ▶ [Vysvětlení dockeru](#)
- ▶ [Soubory dockeru](#)
- ▶ [Základní pojmy](#)

Práce s dockerem

- ▶ [Příkazy](#)
- ▶ [Dockerfile](#)
- ▶ [Zachování dat z kontejneru na lokálním disku](#)

Volumes

- ▶ [Propojení složky z Windows s kontejnerem v dockeru](#)

Řešení problémů

- ▶ [Port není dostupný](#)

Spuštění

```
docker run -d -p 9000:9000 --name portainer --restart always -v  
/var/run/docker.sock:/var/run/docker.sock -v portainer_data:/data  
portainer/portainer-ce:2.27.5
```

Note

- `-d` = Spustí kontejner na pozadí.
- `-p 9000:9000` = Mapuje port 9000 na hostitelském stroji na port 9000 v kontejneru.
- `--name portainer` = Pojmenuje kontejner "portainer".
- `--restart always` = Kontejner se automaticky restartuje, pokud dojde k chybě nebo po restartu hostitelského stroje.
- `-v /var/run/docker.sock:/var/run/docker.sock` = Umožňuje portaineru komunikovat s docker daemonem.
- `-v portainer_data:/data` = Ukládá data portaineru do trvalého úložiště (Volume)
- `portainer/portainer-ce:2.27.5` = Název docker image pro portainer.

1. Vytvoření a nastavení složky pro zálohy

```
mkdir /cesta/k/tvojí/složce
```

- Přístup pro všechny:

```
chmod 777 /cesta/k/tvojí/složce
```

Tip

Přístup pouze pro root.

```
chmod 700 /cesta/k/tvojí/složce
```

2. Nastavení oprávnění pro Docker volumes

Pro zálohu je zapotřebí povolit přístup k souborům a ke složkám v dockeru.

- Povolit rekurzivně pro složky:

```
find /docker_XX -type d -exec chmod 755 {} \;
```

- Povolit rekurzivně pro soubory:

```
find /docker_XX -type f -exec chmod 644 {} \;
```

| Popis | Cesta |
|-------------------|---|
| Host/volume | <code>/var/lib/docker/volumes</code> (cesta k diskům) |
| Path in container | <code>/docker_XX</code> (vlastní cesta v kontejneru pro Volume) |

Important

V kontejneru duplicati musí být nastaveno na "Bind".

i Tip

Chcete-li vrátit zpět oprávnění na výchozí hodnoty, použijte následující příkaz:

```
chown -R root:root/docker_XX
```

Tento příkaz změní vlastníka a skupinu všech souborů a adresářů v `/docker_XX` na `root`, což je výchozí nastavení pro většinu systémů.

1. Zjistěte, jaké máte docker volumes:

```
docker volume ls
```

2. Přenést data z docker volumes do počítače:

```
docker run --rm -v projekty_planka_config:/volume -v  
C:\Users\xxx\Docker_Volumes\planka:/backup busybox:1.37.0-glibc sh -c "cp -r  
/volume/. /backup/"
```

Note

Základní parametry:

`--rm`

- automaticky odstraní kontejner po dokončení operace
- šetří místo na disku a udržuje systém čistý

Připojení volumes:

`-v projekty_planka_config:/volume`

- připojí Docker volume `projekty_planka_config`
- v kontejneru bude dostupný jako adresář `/volume`
- slouží jako zdroj dat pro zálohu

`-v C:\Users\xxx\Docker_Volumes\planka:/backup`

- připojí lokální složku z hostitelského systému
- v kontejneru bude dostupná jako `/backup`
- cílové umístění pro zálohu dat

Použitý image a příkaz:

`busybox:1.37.0-glibc`

- minimalistický Linux image
- obsahuje základní Unix nástroje
- verze s glibc pro lepší kompatibilitu

`sh -c "cp -r /volume/. /backup/"`

- `sh -c`: spustí shell s následujícím příkazem
- `cp -r`: rekurzivní kopírování včetně podsložek
- `/volume/.`: kopíruje obsah složky volume
- `/backup/`: cílová složka pro zálohu

⊗ **Important**

Je zapotřebí znalost dockeru

► [Orchestrátory](#)