

GABRIELE ROMANATO

Avviare tutti i progetti con Docker Compose in diverse directory

Se gestisci più progetti containerizzati distribuiti in diverse cartelle, può diventare scomodo dover eseguire manualmente `docker compose up` per ciascuno. In questo articolo ti mostro come creare uno script Bash per individuare automaticamente tutti i file `docker-compose.yml` e avviare ogni progetto corrispondente.

Obiettivo

Scansionare una directory principale (e tutte le sue sottodirectory) alla ricerca di file `docker-compose.yml` o `docker-compose.yaml`, ed eseguire `docker compose up -d` per ciascuno.

Script Bash

```
#!/bin/bash

# Directory principale dove cercare i progetti
ROOT_DIR="$HOME/progetti"

# Cerca tutti i file docker-compose.* e avvia i container
find "$ROOT_DIR" -type f \( -name "docker-compose.yml" -o -
name "docker-compose.yaml" \) | while read -r compose_file;
do
    project_dir=$(dirname "$compose_file")
    echo "Avvio progetto in: $project_dir"
    (
        cd "$project_dir" && docker compose up -d
    )
done
```

Istruzioni per l'Uso

1. Copia il codice sopra in un file chiamato, ad esempio, `avvia_progetti.sh`.
2. Rendi il file eseguibile:

```
chmod +x avvia_progetti.sh
```

3. Esegui lo script:

```
./avvia_progetti.sh
```

Considerazioni

- Lo script utilizza `docker compose`, la nuova sintassi raccomandata al posto del comando obsoleto `docker -compose`.
- Puoi personalizzare `ROOT_DIR` per adattarlo al tuo sistema.
- Il comando `-d` avvia i container in background. Se vuoi vedere i log, rimuovilo.
- Per evitare di entrare in directory indesiderate (come `node_modules`), puoi aggiungere un filtro a `find`.

Estensioni Possibili

Puoi adattare lo script anche per:

- Fermare tutti i progetti con `docker compose down`.
- Verificare lo stato con `docker compose ps`.
- Loggare l'output su file per eventuali errori.

Questo approccio ti permette di gestire in modo efficiente ambienti multi-progetto basati su Docker, automatizzando l'avvio e rendendo la tua postazione di sviluppo più produttiva.