

GABRIELE ROMANATO

Configurare e usare PostgreSQL con Docker Compose

PostgreSQL è uno dei database relazionali open source più potenti e popolari. Grazie a Docker e Docker Compose, è possibile avviare rapidamente un'istanza PostgreSQL in un ambiente isolato e facilmente gestibile.

Prerequisiti

- Docker installato
- Docker Compose (versione V2, integrata in Docker CLI)

Creare il file `docker-compose.yml`

Il primo passo consiste nel definire il servizio PostgreSQL all'interno di un file `docker-compose.yml`. Ecco un esempio aggiornato e funzionante:

```
services:
  db:
    image: postgres
    restart: unless-stopped
    environment:
      POSTGRES_USER: user
      POSTGRES_PASSWORD: password
      POSTGRES_DB: mydatabase
    ports:
      - "5432:5432"
    volumes:
      - pgdata:/var/lib/postgresql/data

volumes:
  pgdata:
```

Avviare il container

Per avviare PostgreSQL, eseguire il comando seguente nella stessa directory in cui si trova il file `docker-compose.yml`:

```
docker compose up -d
```

Questo comando scaricherà l'immagine ufficiale di PostgreSQL (se non presente), creerà un volume per la persistenza dei dati e avvierà il container in background.

Connettersi al database

Una volta in esecuzione, è possibile connettersi al database con un client PostgreSQL come `psql`:

```
psql -h localhost -p 5432 -U user -d mydatabase
```

La password richiesta sarà quella definita nella variabile `POSTGRES_PASSWORD`.

Fermare e rimuovere i container

Per fermare i container, utilizzare:

```
docker compose down
```

Per rimuovere anche il volume (e quindi i dati persistenti):

```
docker compose down -v
```

Conclusione

Usare PostgreSQL con Docker Compose semplifica la gestione degli ambienti di sviluppo e test. È possibile avviare, fermare e isolare database in pochi comandi, mantenendo al tempo stesso la persistenza dei dati tramite i volumi Docker.