

Docker

如何使用 CLion 在 Docker 容器中編碼/執行程式？

📅 December 9, 2021

Q.

我是 CLion 的新手

我發現 CLion 支援 Docker。我想做以下事情：

(Refer my Linux OS as A.)

(Refer the docker container within A as B.)

The library and environments are set up in B.

The CLion IDE is running in A.

I want to code/debug the program that runs in B using the IDE in A.

這是 CLion 支援的功能嗎？我閱讀了他們的檔案，但我找不到實現它的方法

謝謝

贊助資訊 - 請繼續往下閱讀

A.

編輯：截至 2021.3 預覽版，CLion 已內建對 docker 的支援。有關詳細資訊，請參閱 [更新的答案](#)。

截至 2018 年底，CLion 中的 [遠端開發](#) 非常容易設定。我們所要做的就是設定一個 docker 容器作為我們的“遠端主機”。

我使用 <https://github.com/shuhaoliu/docker-clion-dev> 上的指南作為參考並進行了一些更改。這對我有用：

(可選) 如果您沒有安裝 CLion 的 Docker 外掛，請按照 [這些說明獲取](#)。

第 1 步 - Dockerfile

修改 [此 Dockerfile](#) 以安裝專案所需的任何依賴項。將 Dockerfile 新增到您的專案中。

```
FROM ubuntu:cosmic

#####
# Essential packages for remote debugging and login in
#####

RUN apt-get update && apt-get upgrade -y && apt-get install -y \
    apt-utils gcc g++ openssh-server cmake build-essential gdb gdbserver rsync vim

RUN mkdir /var/run/sshd
RUN echo 'root:root' | chpasswd
RUN sed -i 's/PermitRootLogin prohibit-password/PermitRootLogin yes/' /etc/ssh/sshd_config

# SSH login fix. Otherwise user is kicked off after login
RUN sed 's@session\s*required\s*pam_loginuid.so@session optional pam_loginuid.so@g' -i \
/etc/ssh/sshd_config

ENV NOTVISIBLE "in users profile"
RUN echo "export VISIBLE=now" >> /etc/profile

# 22 for ssh server. 7777 for gdb server.
EXPOSE 22 7777
```

```
RUN useradd -ms /bin/bash debugger
RUN echo 'debugger:pwd' | chpasswd

#####
# Add custom packages and development environment here
#####

#####

CMD ["/usr/sbin/sshd", "-D"]
```

第 2 步 - Docker 撰寫

在與之前的 docker 檔案相同的目錄中，建立一個 docker-compose.yml 檔案。

```
# From: https://github.com/shuhaoliu/docker-clion-dev/blob/master/docker-compose.yml

version: '3'

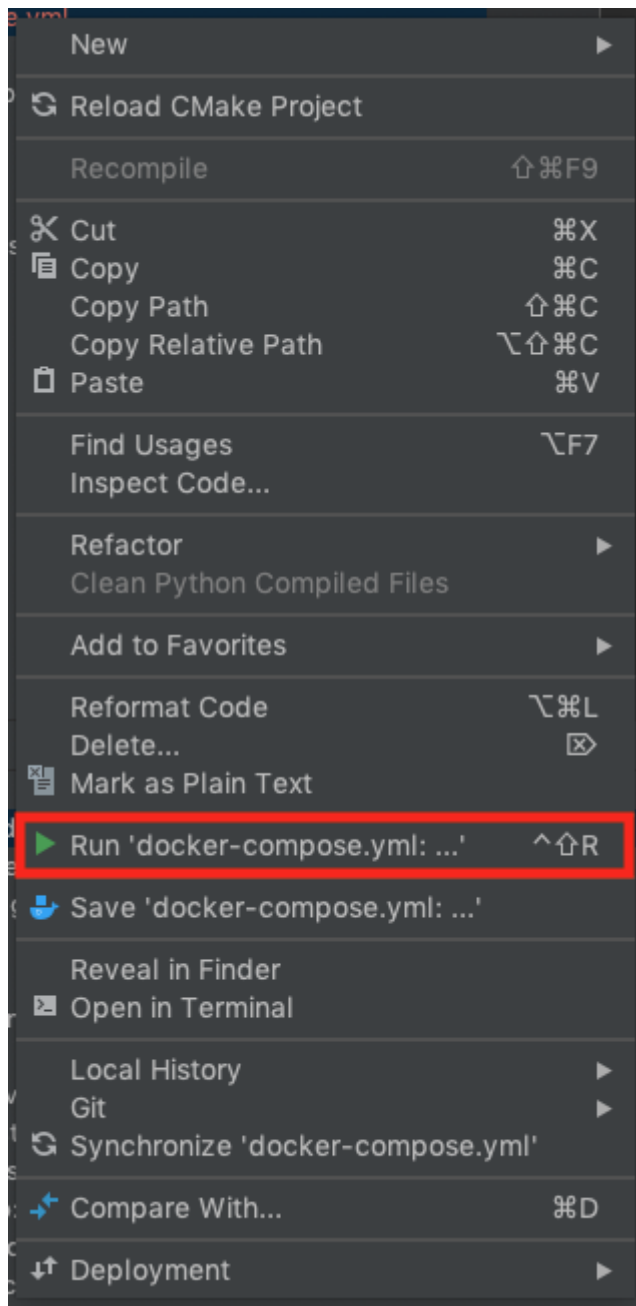
services:
  gdbserver:
    build:
      context: ./
      dockerfile: ./Dockerfile
    image: clion_dev
    security_opt:
      - seccomp:unconfined
    container_name: debug
    ports:
      - "7776:22"
      - "7777:7777"
    volumes:
      - ./home/debugger/code
    working_dir: /home/debugger/code
    hostname: debug
```

第 3 步

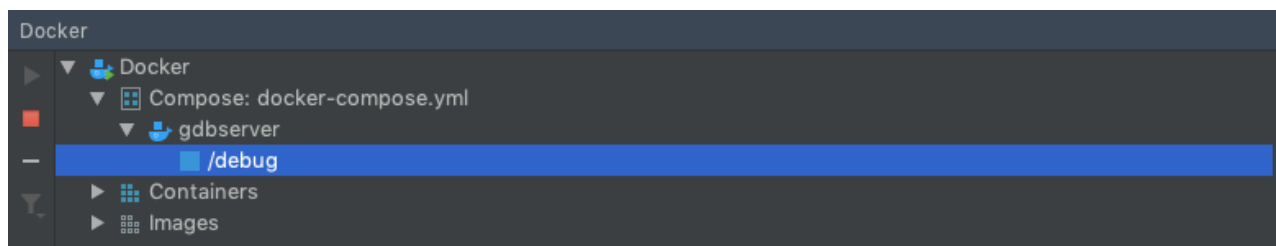
確保 `Dockerfile` 和 `docker-compose.yml` 檔案位於同一目錄中。

選項 3A (使用 CLion Docker 外掛)

右鍵點選該 `docker-compose.yml` 檔案並選擇 `Run` 。



一兩分鐘後，容器應該被建立並且可以從 CLion 的 Docker 選項卡中檢視。



選項 3B (沒有 CLion Docker 外掛)

從包含 `Dockerfile` 和 `docker-compose.yml` 檔案的目錄中，執行：

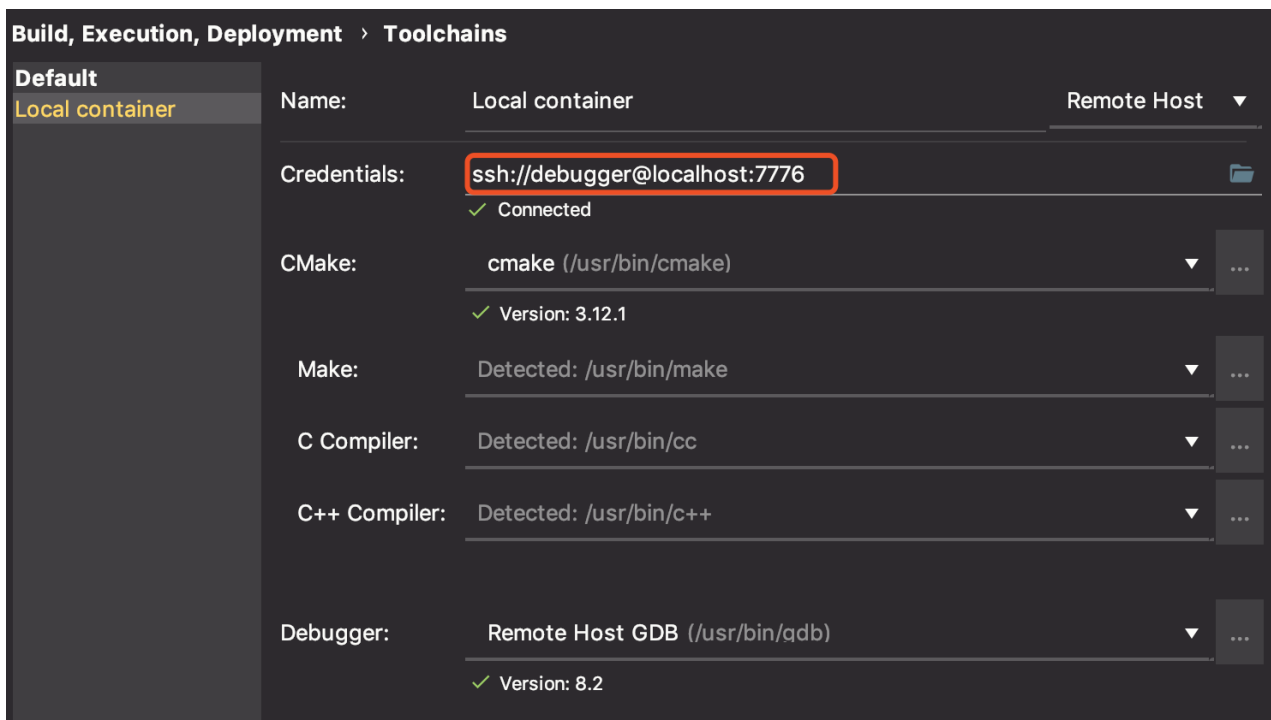
```
docker-compose up -d
```

第 4 步 - 配置工具鏈

開啟 **Settings->Build, Execution, Deployment -> Toolchains** 並建立一個新的 **Remote Host Toolchain**。

在 **Credentials** 欄位中，點選右側的小資料夾，然後輸入在 `Dockerfile` 中建立的偵錯程式使用者的憑據。

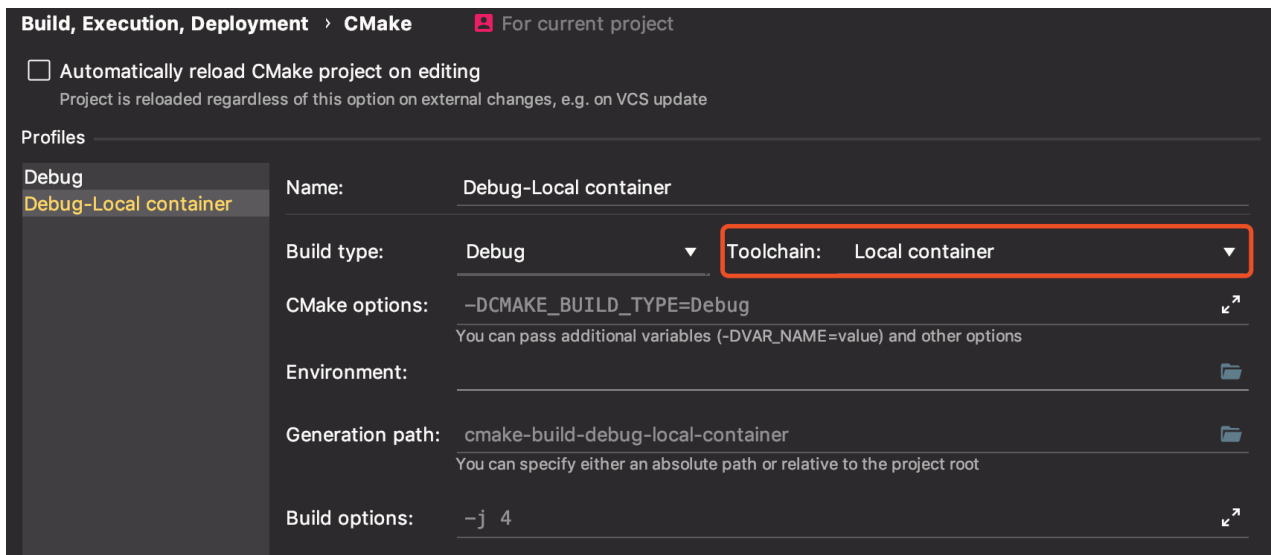
在上面的範例中，使用者名是“debugger”，密碼是“pwd”。



第 5 步 - CMake 配置檔案

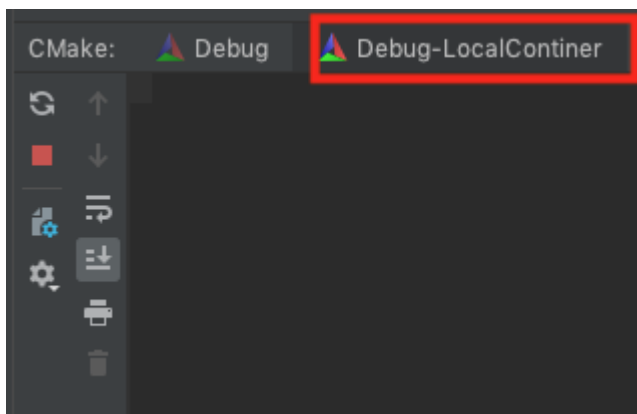
現在我們必須設定一個 CMake 配置檔案以使用我們新的遠端主機工具鏈。

導航到 **Settings->Build, Execution, Deployment -> Cmake** 並建立一個新的配置檔案。唯一必要的更改是選擇在上一步中建立的工具鏈。

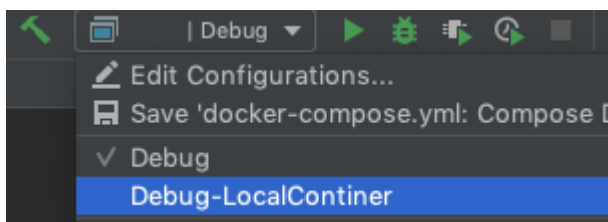


第 6 步 - 執行/除錯程式

在 CMake 選項卡中，確保您選擇了新建立的 CMake 配置檔案。



將 CMake 專案載入到容器中後，您應該可以在 CLion 右上角的執行配置切換器中選擇要使用的 CMakeProfile。



希望如果一切順利，您現在應該能夠在 docker 容器中執行和除錯程式碼！

如果事情沒有完全按計劃進行，這裡有一些參考資料可以幫助我讓事情順利進行：

[docker-clion-dev 指南](#)

[CLion 遠端專案指南](#)

[Clion CMake 配置檔案](#)

[CLion CMake 工具鏈](#)

引用自：<https://stackoverflow.com/questions/55272484>

贊助資訊 - 請繼續往下閱讀

Quickpon 極客幫

Quickpon 極客幫，致力於成為中文世界程式設計學習者、工作者的超級助手，提供內容優質、主題多元的程式相關疑難解答，定期彙整來自 Stack Overflow 等各類文章精選內容，讓您解 Bug 零語言障礙、與國外技術社群即刻接軌！

Advertisement

主題分類



➤ Algorithm	2780
➤ Amazon web services	1943
➤ Arrays	1479
➤ Database	1281
➤ Delphi	1727
➤ Docker	2210
➤ Git	9353
➤ Go	1788
➤ R	10284
➤ Regex	2225
➤ Rust	1169
➤ Scala	4236
	1753

➤ Spring	
➤ Svn	1296
➤ 更多主題...	

Advertisement

Advertisement

相關問答

Docker

如何殺死 **docker-compose** 生成的容器？

📅 May 11, 2022

檢視問答

Docker

如何更正**makefile**中的**docker**，它需要至少**1**個參數來刪除所有容器命令

📅 May 3, 2022

檢視問答

Docker

如何在 WSL2 上公開 Docker TCP 套接字？（WSL 安裝的 Docker，而不是 Docker Desktop）

📅 April 28, 2022

檢視問答

Amazon-Web-Services

GitLab Runner 突然無法使用 Docker Machine 和 AWS Autoscaling 執行作業

📅 April 28, 2022

檢視問答

Docker

無法使用新的 Windows 終端在 WSL2 中啟動 Docker

📅 April 27, 2022

檢視問答

Docker

Kubernetes 從不安全的 docker 系統資料庫中提取

📅 April 25, 2022

檢視問答

Quickpon 極客幫

[首頁](#)

[主題分類](#)

[聯絡我們](#)

[根據 Cc By-Sa 許可](#)

[隱私權政策](#)

Designed By [Quickpon! 極客幫](#)