

VScode 設定 GitHub 連結

已經有github的帳戶

1.要在新的PC上安裝VSCode, 使用它在 Resume 的資料夾上, 編輯markdown格式的Resume.MD履歷表及autobiography自傳, 然後上傳至github 上的 Resume 的repository 下

2.再share給他人求職用.

-

A. 重點是VScode上要如何設定才可以上傳到正確的 github 帳戶上?

B. 請列出重要的步驟

很好 🍌 你已經有 GitHub 帳戶，現在只差 在新 PC 正確設定 VS Code 並連到「正確的 GitHub 帳號」。

我幫你整理成：

✅ A. VS Code 要如何設定，才能連到正確的 GitHub 帳戶？

在新電腦上，關鍵只有 3 件事：

◆ Step 1 安裝必要軟體

1. 安裝 Git

👉 下載：Git

安裝完成後確認：

```
bash  
  
git --version
```

2. 安裝 VS Code

👉 下載：Visual Studio Code

◆ Step 2 設定 Git 使用者資訊（非常重要）

⚠ 這一步決定 commit 顯示的是誰

打開 VS Code → Terminal 輸入：

```
bash  
  
git config --global user.name "你的GitHub名字"  
git config --global user.email "你的GitHub註冊Email"
```

確認：

```
bash

git config --global --list
```

✉ email 一定要跟 GitHub 帳戶一致
不然 commit 會變成陌生帳號

◆ Step 3 登入 GitHub（建議用官方登入方式）

方法一（最簡單，推薦）

在 VS Code 左下角 → 點擊帳戶圖示 →
「Sign in to GitHub」

會跳出瀏覽器登入 GitHub

登入後：

- VS Code 會綁定到你的 GitHub
- push 時不會再問帳密

方法二（使用 Personal Access Token）

到 GitHub → Settings → Developer settings →
Personal access tokens → 產生 token

當 push 時：

- Username：你的 GitHub 帳號
- Password：貼上 token

✅ B. 重要完整流程（從零開始）

假設 GitHub 上已經有一個 repository 叫：

```
vbnet

Resume
```

🔥 正確完整流程

◆ 1 在新電腦 Clone Repository（不要自己亂建資料夾）

⚠ 很多人會做錯這一步

請不要自己建立 Resume 資料夾
請用 clone !

```
bash

git clone https://github.com/你的帳號/Resume.git
```

這樣才會：

- 正確連到 GitHub 上的 Resume repository
- 不會 push 到錯的地方

◆ 2 用 VS Code 打開這個資料夾

```
bash

cd Resume
code .
```

◆ 3 建立 Markdown 檔案

建立：

```
vbnet

Resume.md
autobiography.md
```

Markdown 基本格式例如：

```
markdown

# Your Name

## Education
- ...

## Experience
- ...
```

◆ 4 上傳到 GitHub

```
bash

git add .
git commit -m "Add resume and autobiography"
git push origin main
```

成功後就會出現在 GitHub 上

如何分享給求職用？

有 3 種方式：

方法 1 直接分享 GitHub Repository 連結

```
arduino
```

```
https://github.com/你的帳號/Resume
```

方法 2 使用 GitHub Pages（更專業 🔥）

可以開啟 GitHub Pages
讓網址變成：

```
cpp
```

```
https://你的帳號.github.io/Resume/
```

（如果需要我可以教你怎麼開 GitHub Pages）

方法 3 轉成 PDF 上傳

可以用：

- VS Code Markdown Preview
- 或轉成 PDF 再上傳

常見錯誤（一定要避免）

- ✗ 自己建資料夾再 git init
- ✗ 沒設定 user.email
- ✗ push 到錯的 GitHub 帳號
- ✗ 用 HTTPS 卻沒有登入 VS Code

給你一個最安全做法

在新電腦：

1. 安裝 Git
2. 安裝 VS Code
3. VS Code 內登入 GitHub

4. 用 clone 把 Resume 下載
5. 編輯
6. push

這樣 100% 不會錯帳號

如果你願意，我可以幫你做一個：

- ◆ 一頁式「GitHub + VS Code 設定流程圖」
- ◆ 或幫你設計一個專業版 Markdown 履歷模板

你想做成學術風？工程師風？還是簡潔求職版？

VS Code上需安裝什麼 (延伸的模組) 嗎？

很好 👍 這題問得非常關鍵。

其實：

- ✓ 基本上「不一定要安裝任何模組」就可以 push 到 GitHub。
- 因為 Visual Studio Code 本身就內建 Git 功能。

但如果你要「順暢 + 專業 + 好維護」

我建議安裝以下幾個「實用延伸模組」

必裝（強烈建議）

GitHub 官方整合


GitHub Pull Requests and Issues

（官方出品）

用途：

- 直接在 VS Code 內登入 GitHub
- 看 PR
- 管理 repository

發佈者：GitHub

 安裝後登入一次，就會綁定正確帳戶。

2 Markdown 預覽增強（寫履歷一定要）

👉 Markdown All in One

功能：

- 自動目錄
- 快速格式化
- 表格輔助
- 快速清單

寫 Resume.md 非常好用 🍌

💡 推薦（讓履歷更漂亮）

3 Markdown Preview Enhanced

功能：

- 支援數學公式
- 支援流程圖
- 可匯出 PDF

如果你將來要做技術型履歷（AI / Research）會很有幫助。

🎨 如果你想做更漂亮履歷

4 Markdown PDF

可以直接：

```
mathematica
```

右鍵 → **Export PDF**

🧠 如果你想更清楚看 Git 變化（推薦）

5 GitLens




功能：

- 看每行是誰改的
- 看 commit 歷史

- 比較版本

這對求職展示專案很加分。

不需要安裝的

-  不需要 GitHub Copilot
-  不需要 其他複雜 Git plugin
-  不需要 Docker

單純寫履歷 + 上傳 GitHub

只需要：

- Git
- VS Code
- GitHub 登入

最精簡推薦組合（給求職用）

只裝這 3 個：

1. GitHub Pull Requests and Issues
2. Markdown All in One
3. GitLens

額外專業建議（重要）

如果你打算：

- ◆ 用 GitHub 當線上履歷
- ◆ 未來做 AI / Research 發表

我會建議你開啟：

 GitHub Pages（免費個人網站）

如果需要，我可以幫你：

- 做一份「教授風」Markdown履歷模板
- 或「工程師風」GitHub展示履歷
- 或教你開 GitHub Pages（3分鐘完成）

你想走哪種求職定位？

幫忙設計一個專業版 Markdown 履歷模板

很好 🍌

既然你本身有 AI / 醫療影像背景（蛀牙 segmentation、TensorFlow），我幫你設計一份：

🎓 學術 + AI 工程師雙向通用版 Markdown 履歷模板

適合：

- 申請研究所 / 博士
- 投稿期刊
- AI / Data / Medical AI 工作

你可以直接存成：

vbnet

Resume.md

📌 專業版 Markdown 履歷模板

markdown

Your Name, Ph.D. (or M.S.)

AI Researcher | Medical Image Analysis | Deep Learning

📍 Location, Country

✉️ your_email@email.com

🔗 GitHub: <https://github.com/yourname>

🔗 LinkedIn: <https://linkedin.com/in/yourname>

🧪 Research Interests

- Medical Image Segmentation
- Deep Learning (CNN, U-Net, Transformer-based models)
- Computer Vision in Dentistry
- Clinical AI Deployment

🎓 Education

University Name – Degree

Department, Year

Thesis: **Title of Your Thesis**

Advisor: Prof. XXX

Relevant Coursework:

- Machine Learning
- Deep Learning
- Medical Informatics
- Biostatistics

🛠️ Technical Skills

Programming

- Python (TensorFlow, PyTorch, OpenCV)
- SQL
- C#


```

### AI / Deep Learning
- U-Net / Attention U-Net
- CNN + Transformer hybrid models
- Transfer Learning
- Model Evaluation (IoU, Dice, Confusion Matrix)

### Tools
- Git / GitHub
- VS Code
- Linux
- Colab

---

## 📁 Research Projects

### ♦ Dental Caries Segmentation Using U-Net
- Built segmentation model using bitewing X-ray images
- Dataset: 300 patients
- Achieved Dice Score: XX%
- Applied data augmentation and k-fold validation
- Submitted to *Computer Methods and Programs in Biomedicine*

GitHub: https://github.com/yourname/project-link

---

### ♦ Project Title
- Short description
- Technical highlights
- Quantitative results

---

## 📄 Publications

- Your Name, et al. "Paper Title." *Journal Name*, Year.
- Your Name, et al. "Conference Paper Title." Conference, Year.

---

## 🏆 Awards & Certifications
- Scholarship Name
- AI Certification
- Academic Award

---

## 👨‍🏫 Teaching Experience
Course Name – University
- Taught AI / Data Analysis
- Designed curriculum on CNN and medical AI

---

## 🌐 Academic Service
- Reviewer for Journal XXX
- Conference Volunteer

---

## 📌 Selected Talks
- "Medical AI in Dentistry" – Conference Name, Year

---

## 🌟 Additional Information
- Languages: English (Fluent), Mandarin (Native)
- Research Collaboration: Multidisciplinary teams

```

我可以幫你改成：

- 更強調技術 stack
 - 更量化數據成果
 - Less academic, more impact-driven
-

如果你想做成「GitHub Pages 個人網站版」

我可以幫你做：

- 單頁式個人網站模板
 - 美觀排版
 - 自動部署到 GitHub Pages
-

建議你下一步

你可以告訴我：

1. 你是要申請學術？還是業界 AI？
2. 要全英文？還是中英對照？
3. 要簡潔一頁式？還是完整 CV（3~4頁）？

我可以幫你客製成「真正能用來投履歷」的版本。