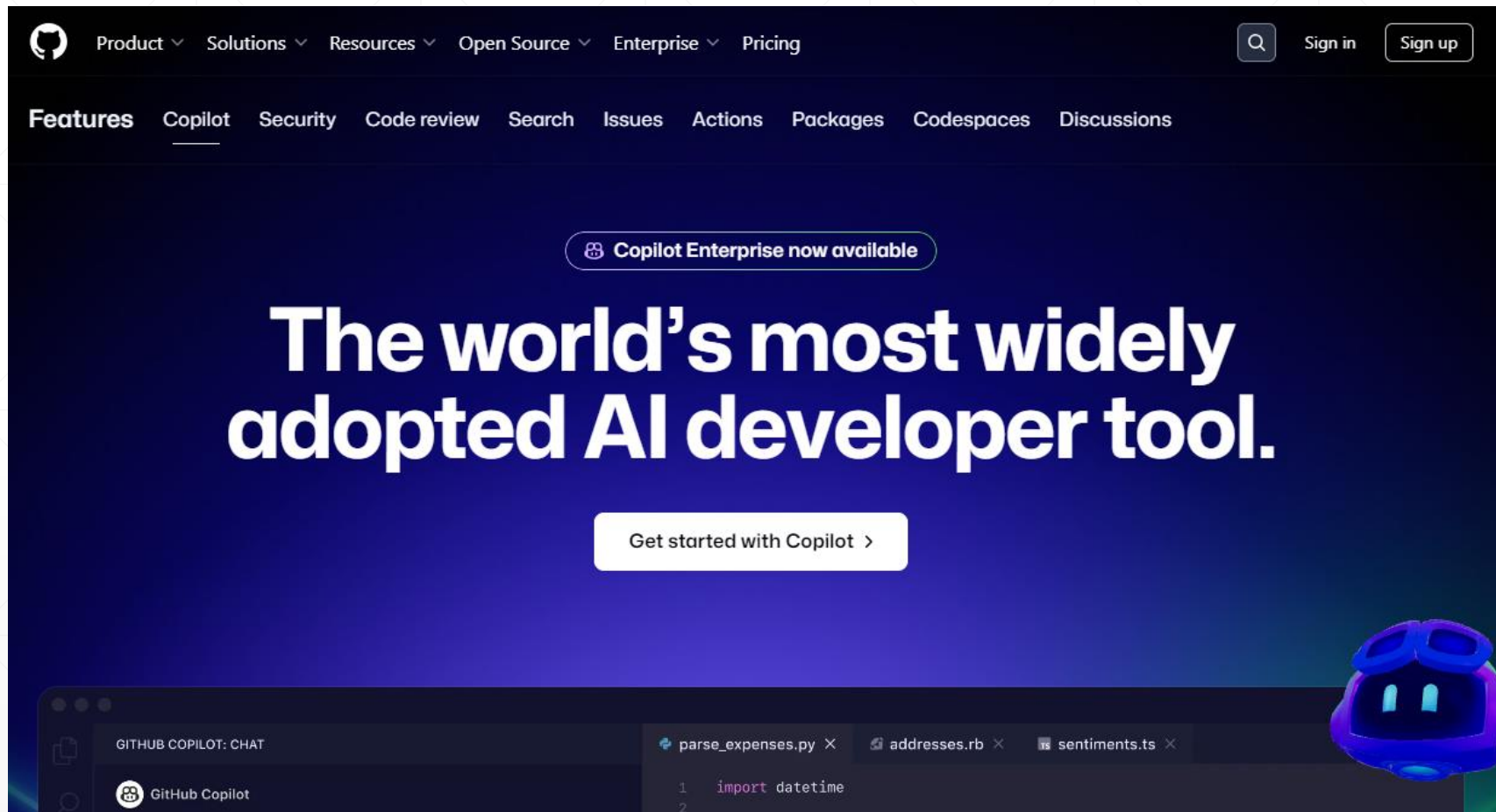


Github Copilot 協作開發實戰

亞洲大學資訊發展處 楊承勳

Github Copilot

[GitHub Copilot · Your AI pair programmer](#)



什麼是Github Copilot?

- 基於 AI / LLM 技術的開發助手，用於改善開發體驗(DX)
 - 核心技術來自於 OpenAI 的 [GPT-4模型](#)
 - 該模型基於海量的自然語言文本資料集（ GitHub 公開儲存庫 ）進行訓練，並具備一定的推理能力
 - 主要能力
 - 可以透過使用者輸入的部分程式碼和註解來產生相應的程式碼
 - 可以解釋程式碼的邏輯與行為
 - 注意事項
 - 可以透過精鍊的提示(Prompt)告訴它該如何行動(Action)
 - 是一個沒有「你」不行的輔助工具
-

大幅提昇開發者體驗

- 幾乎支援「任何」程式語言
 - 包含但不限於 Java, PHP, Python, JavaScript, Ruby, Go, C#, C++, ...
 - 只要是 GitHub 公開儲存庫找到的 Code 都看的懂！
 - 專注於解決問題
 - 幫助開發人員花更少的時間在重複與單調的開發工作上
 - 快速完成開發並產生有品質的程式碼
 - Flow - 強調寫 Code 的 Flow !
 - 離開編輯器再回來會一定程度破壞FLOW。例如：上網查詢sample code、寫到一半接客戶的電話等等。
-

Github Copilot 的價格

- 個人版：一次性 30 天免費試用，月繳 \$10 美元或年繳 \$100 美元
 - 商業版：月繳 \$19 美元/使用者 (主要使用在組織帳戶中)
 - 企業版：月繳 \$39 美元/使用者
 - 可為組織提供個人化功能，以及將GitHub Copilot整合到GitHub中，並可索引組織的程式碼庫
 - GitHub Student Developer Pack (Github學生大禮包) : 免費
 - 帳號申請教學網址一
 - 帳號申請教學網址二
 - 上傳證明文件方式建議用手機於網頁中直接拍照上傳！
-

安裝和設定Github Copilot

支援Github Copilot的編輯器

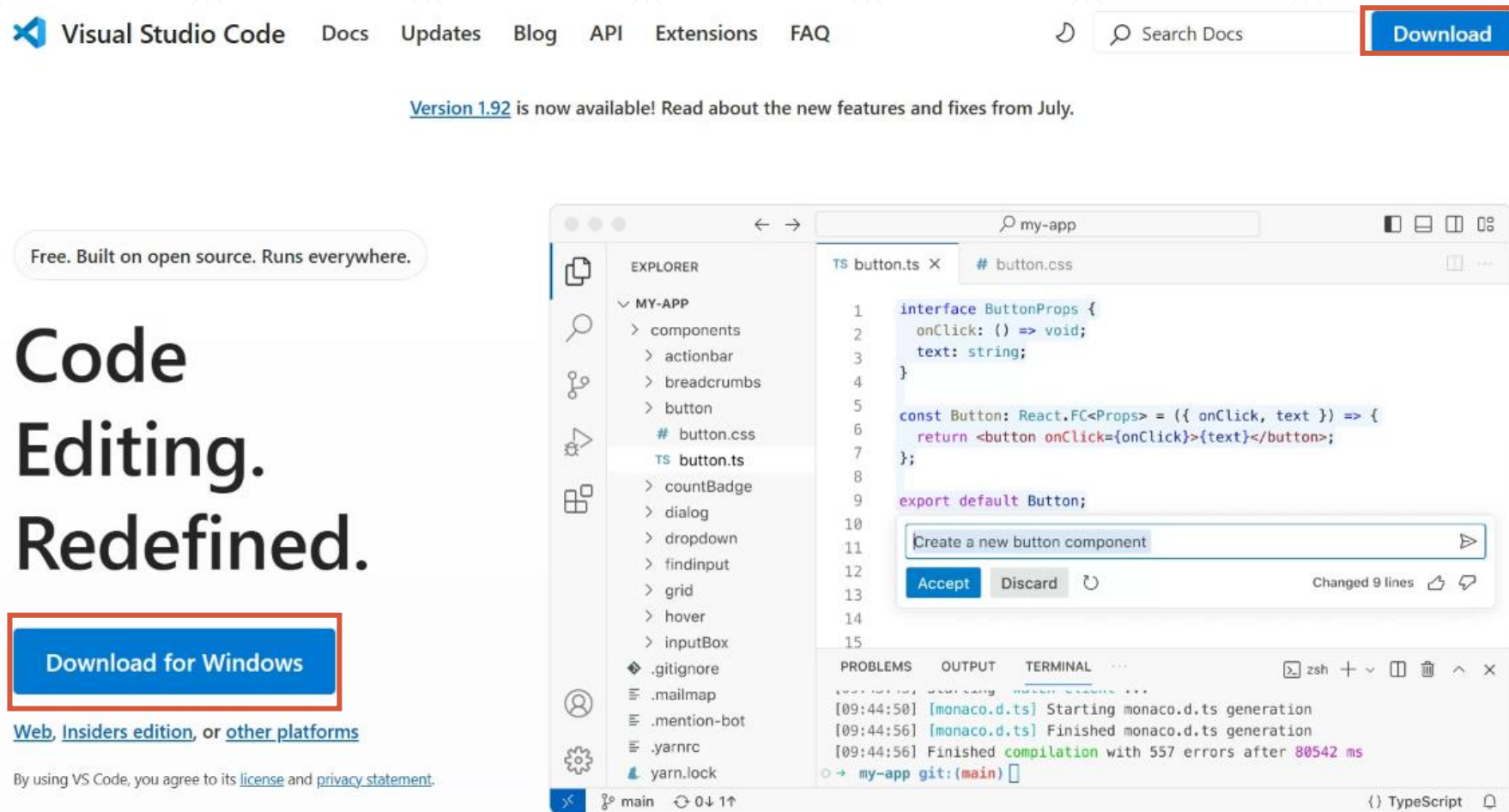
- Visual Studio Code
 - 本教材以此編輯器做範例
 - [Visual Studio Code - Code Editing. Redefined](#)
 - Visual Studio 2022 v17.6 or later ([GitHub Student Developer Pack](#))
 - [Visual Studio 2022 社群版本 – 下載最新的免費版本 \(microsoft.com\)](#)
 - JetBrains IDEs ([GitHub Student Developer Pack](#))
-

VS Code常用Github Copilot擴充套件

- Github Copilot
 - [GitHub Copilot - Visual Studio Marketplace](#)
 - Github Copilot Chat
 - [GitHub Copilot Chat - Visual Studio Marketplace](#)
 - 用聊天的方式使用Github Copilot，對於對話內容有一定程度的短期記憶能力，近似ChatGPT的使用方式
 - 請Github Copilot 解釋程式碼
 - 安裝時會同步安裝Github Copilot !
 - 支援**繁體中文** (zh-TW) !
 - GitHub Copilot Voice *(Preview)*
 - [GitHub Copilot Voice - Visual Studio Marketplace](#)
 - 用說的方式寫程式
-

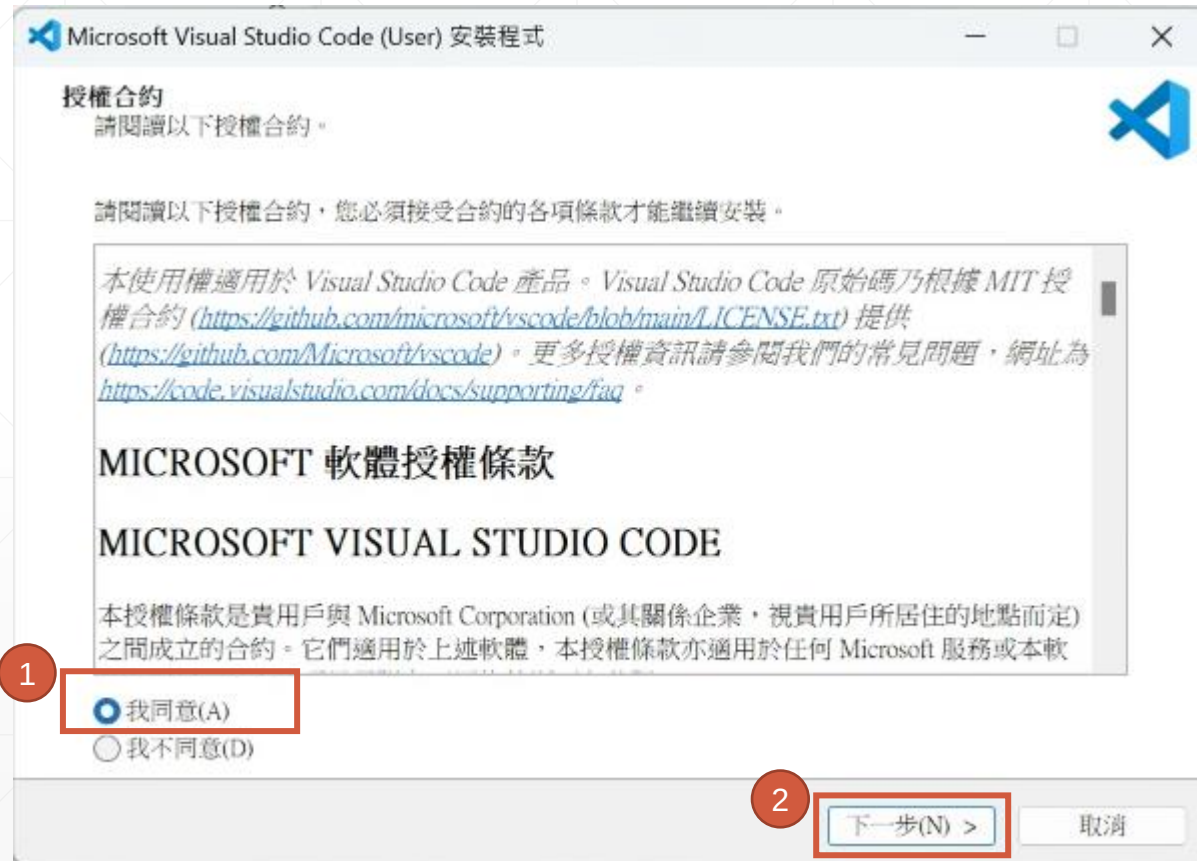
安裝VS Code-1

- 請連線到 [Visual Studio Code - Code Editing. Redefined](https://code.visualstudio.com/)，然後點擊Download



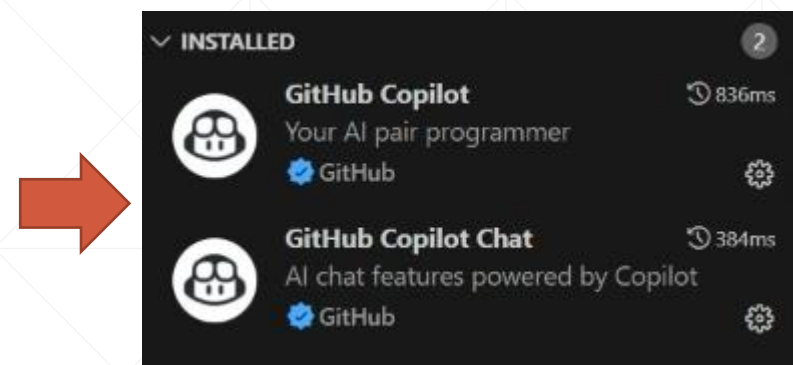
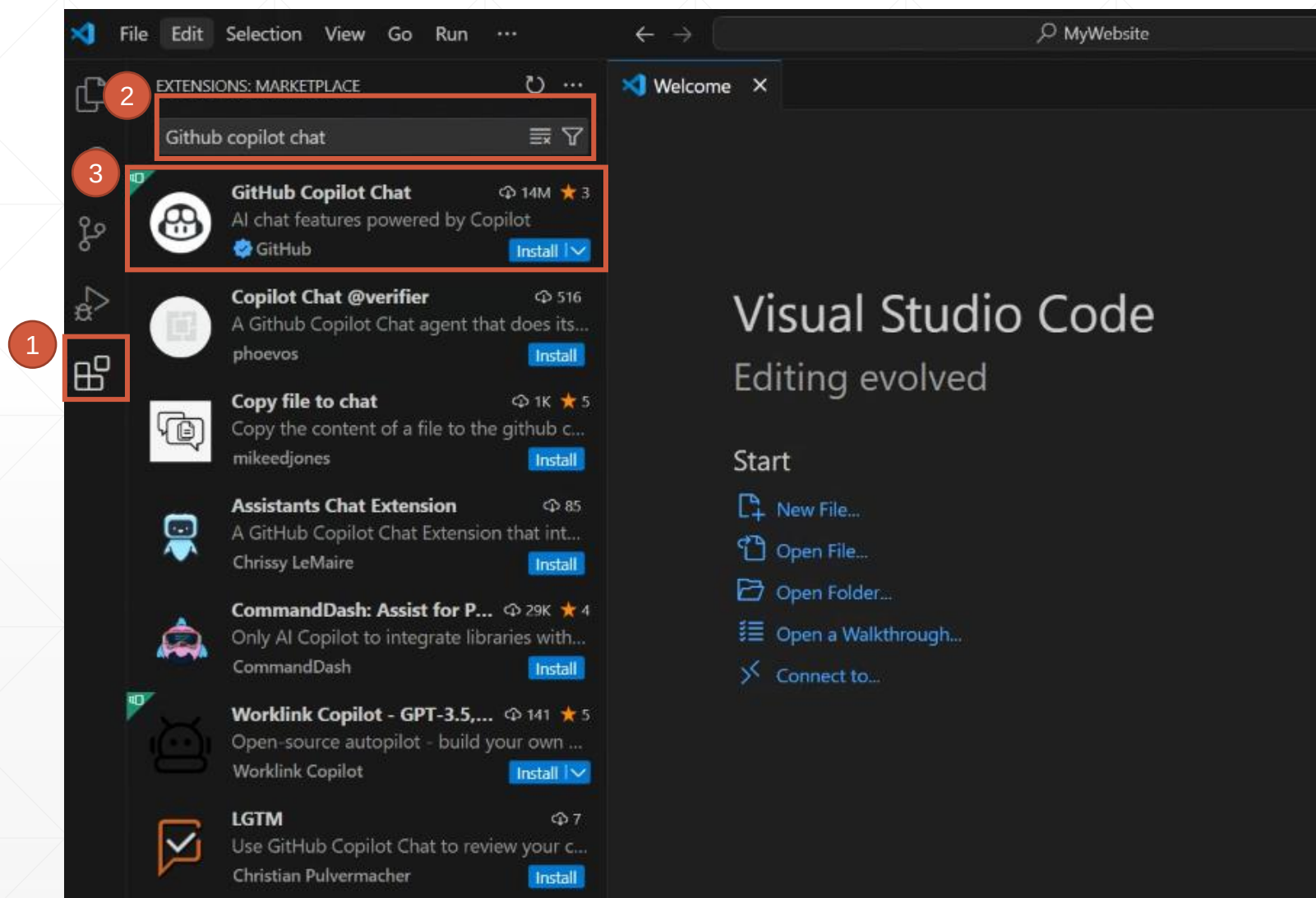
安裝VS Code-2

- 打開下載中的「安裝檔」，勾選「我同意」後，點擊「下一步」，一一確認後，點擊「安裝」。



在VS Code上安裝Github Copilot

- 打開VS Code後，可以從左側的圖示中選取「Extensions」，然後在搜尋框中輸入「Github Copilot Chat」，找到套件後點選「Install」。

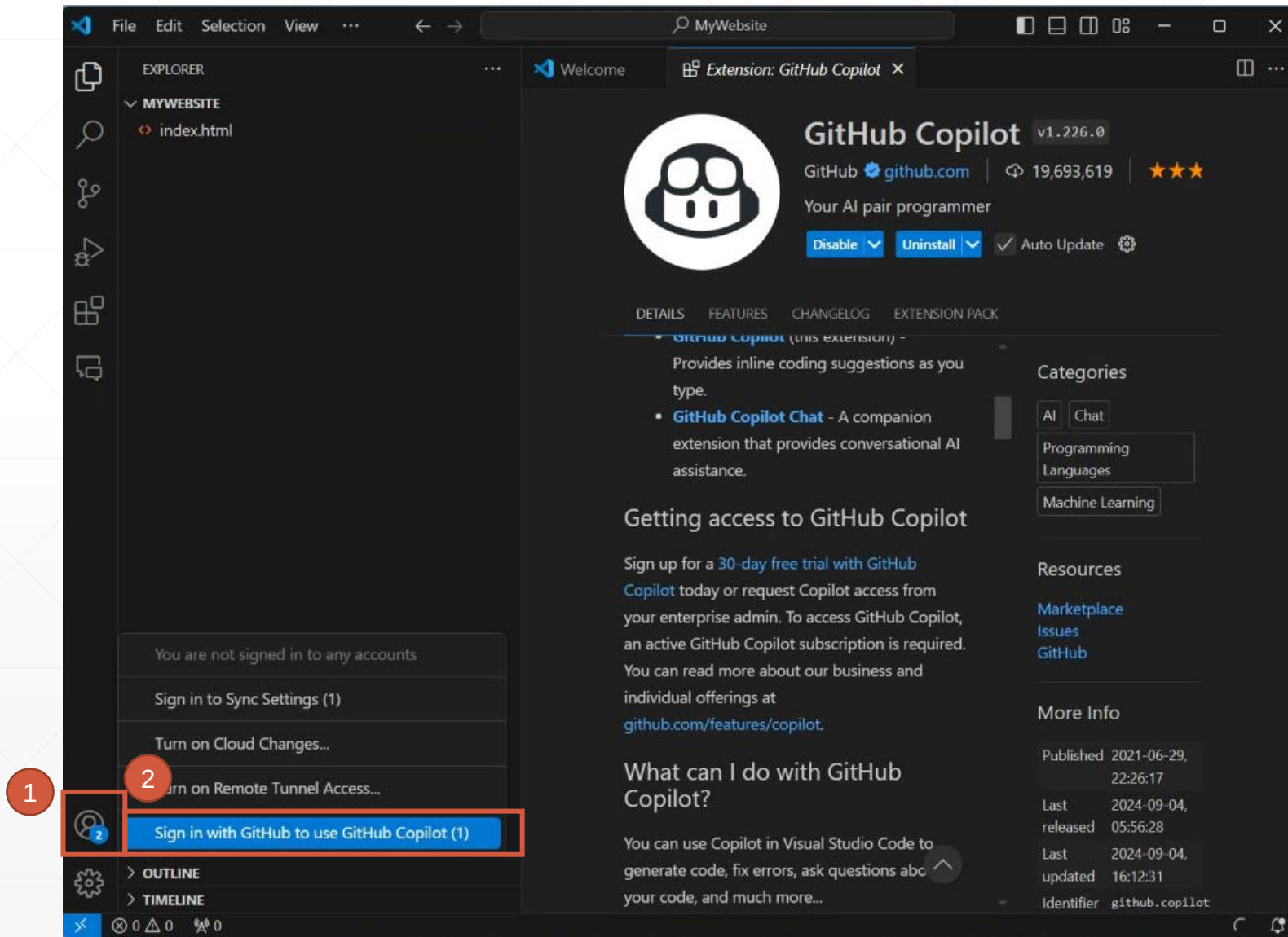


系統會同步安裝

Github Copilot

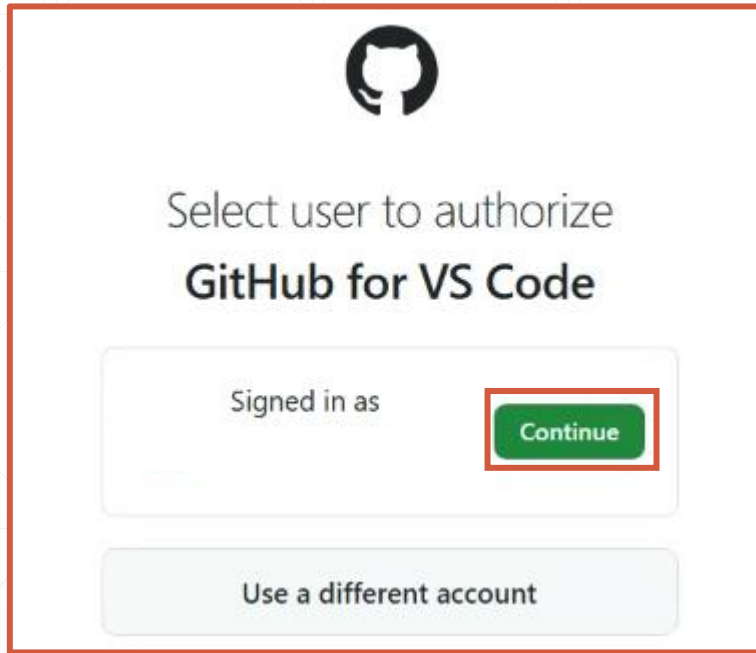
登入Github 帳號 開始使用Github Copilot -1

- 選擇左下角Account設定，然後選取Sign in with Github to use Github Copilot。



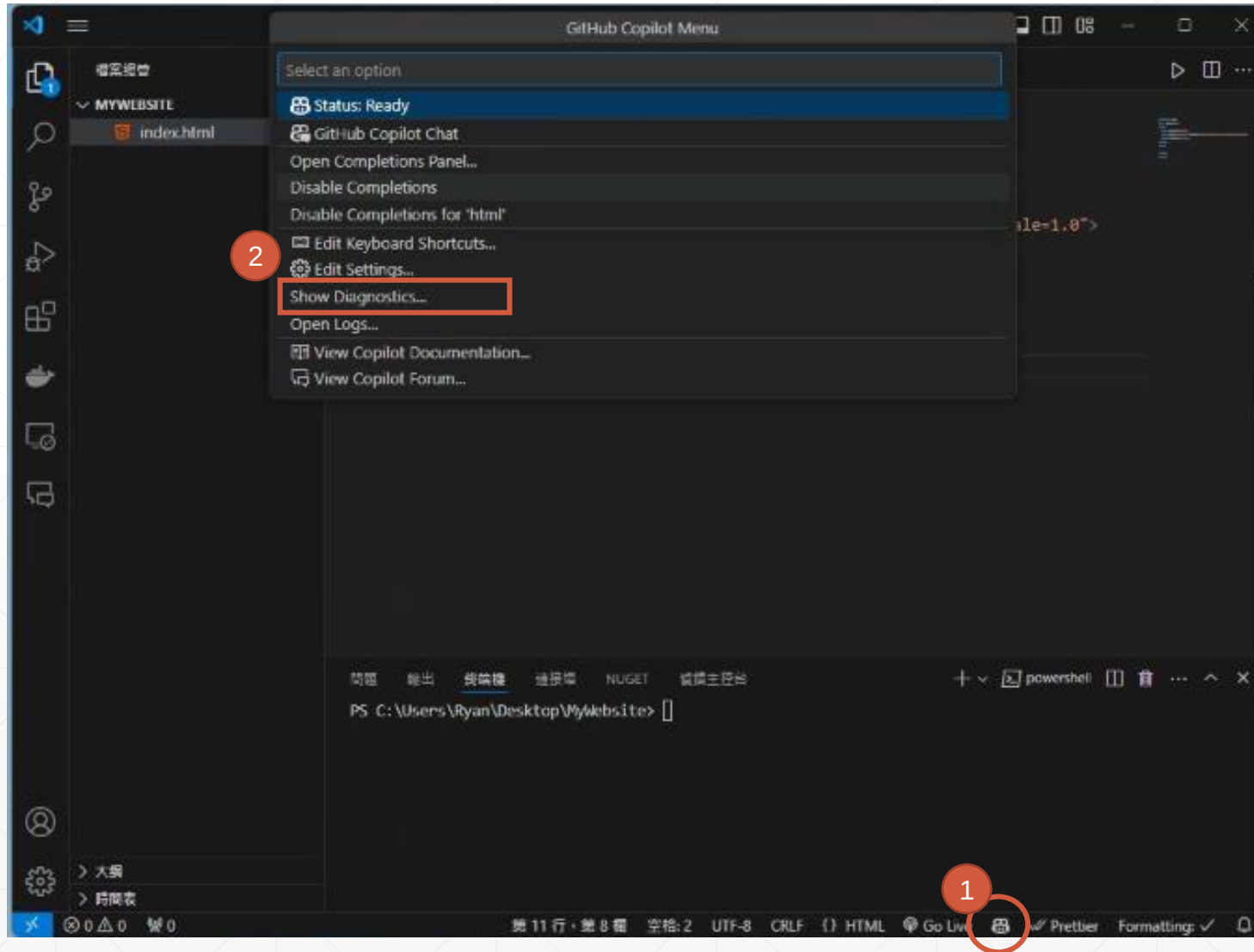
登入Github 帳號 開始使用Github Copilot -2

- 登入完成後，選擇「Continue」後，點擊「開啟」，打開VS Code完成登入。



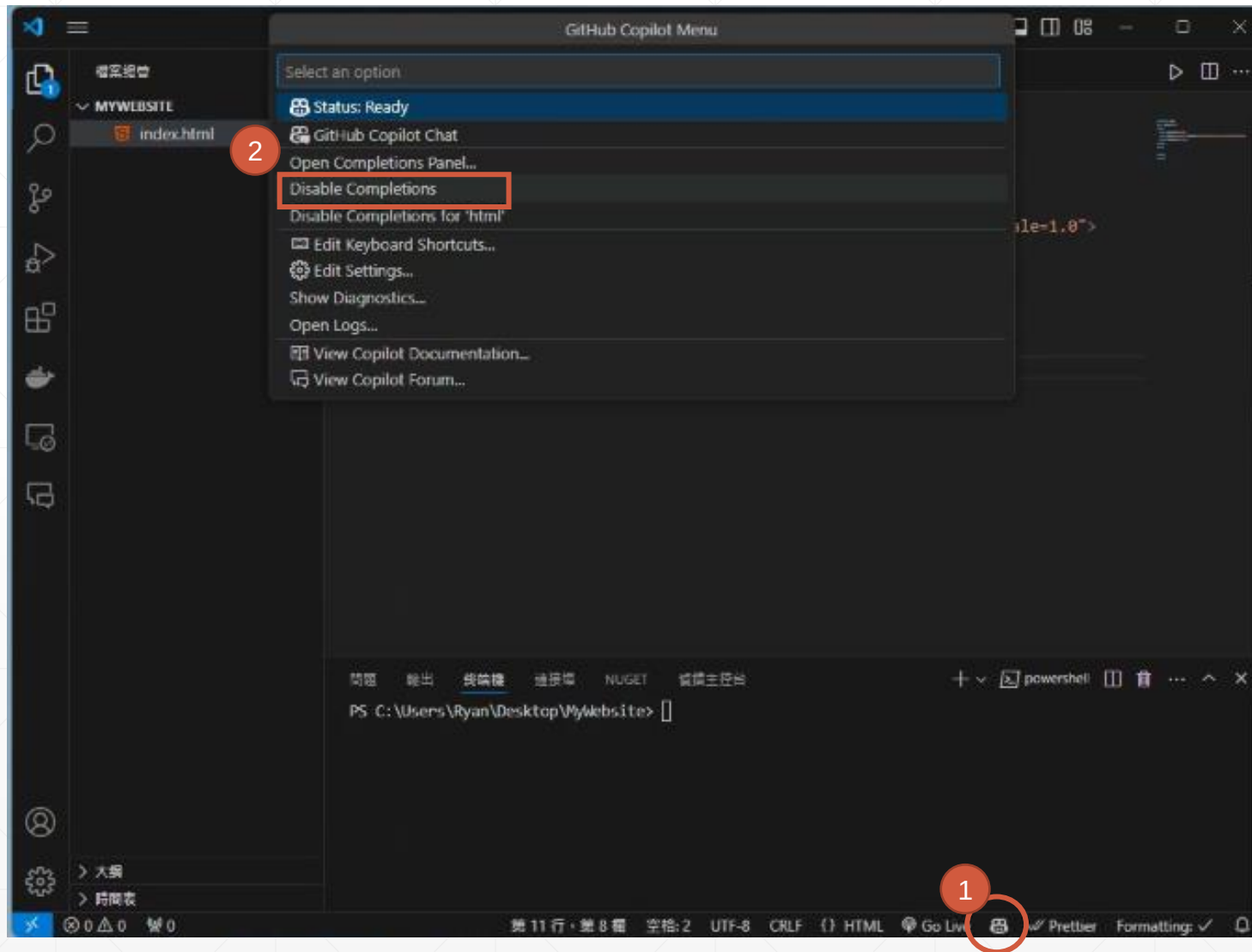
登入Github 帳號 開始使用Github Copilot -3

- 返回VS Code後，可以點選狀態列的圖示，然後選取「Show Diagnostics...」，滑到最底將會顯示登入資訊。。



在VS Code中啟用或停用Github Copilot

- 想要啟用或停用Github Copilot 可以點選狀態列的圖示，然後選取「Disable Completions」。



Github Copilot

使用Github Copilot的兩種方式

- CDD(Comment Driven Development)
 - 透過**註解**來開發
 - The Flow
 - 照著您的**開發流程**自動產生程式碼
-

CDD(Comment Driven Development)

描述你的需求

- 使用註解的方式，讓AI知道你想打造什麼功能。

靈活的描述需求

- 同樣一句話在不同地方可能代表不同意思，適時的修改註解可以讓AI更清楚知道你的意圖。

提供良好的範例

- 如果需要產生假資料，請給他一些範例，讓他知道如何造樣造句。

透過註解進行提示(Prompting)

- 好的提示應該要有什麼準則
 - 電腦只能做你告訴他們需要做的事
 - 你的指令需要非常具體
 - 他們更擅長一步一步接受命令，並且完工作
 - 沒有具體的提示，Github Copilot可能會回應錯誤的程式碼！
-

透過註解來寫Code

- 程式語言：JavaScript
- 程式需求：透過註解提示接下來要寫的Code

```
// 產生10個質數，接著依續顯示在網頁上
let count = 0;
let num = 2;
while (count < 10) {
  let isPrime = true;
  for (let i = 2; i < num; i++) {
    if (num % i === 0) {
      isPrime = false;
      break;
    }
  }
  if (isPrime) {
    document.write(num + '<br>');
    count++;
  }
  num++;
}
```

```
<script>
/* 我有一個 web api 網址是 https://edurpap2.asia.edu.tw/App\_api/api/app\_08/scr\_data
http的方法是使用get
請幫我用fetch寫一個function 去查詢
並且用表格的方式列在畫面上 */
async function getData(){
  let url = 'https://edurpap2.asia.edu.tw/App_api/api/app_08/scr_data';
  let response = await fetch(url);
  let data = await response.json();
  console.log(data);
  let str = '<table border="1">';
  str += '<tr><th>學號</th><th>姓名</th><th>國文</th><th>英文</th><th>數學</th><th>總分</th><th>平均分</th></tr>';
  for(let i=0; i<data.Length; i++){
    str += '<tr>';
    str += '<td>' + data[i].cID + '</td>';
    str += '<td>' + data[i].cName + '</td>';
    str += '<td>' + data[i].cChinese + '</td>';
    str += '<td>' + data[i].cEnglish + '</td>';
    str += '<td>' + data[i].cMath + '</td>';
    str += '<td>' + data[i].cTotal + '</td>';
    str += '<td>' + data[i].cAvg + '</td>';
    str += '</tr>';
  }
  str += '</table>';
  document.getElementById('show').innerHTML = str;
}
```

沒有告訴 Copilot
欄位，他只能自己
產生可能會有的
欄位！

提供範例讓Github Copilot更知道如何回應

- 程式語言：JavaScript
- 程式需求：提供範例資料產生更好的結果

```
/* 建立5筆範例資料
每個物件資料大概長這樣 [{"id": 1, "name": "亞洲大學校安中心-24小時校內專線", "tel": "04-23321053", "image": null, "map_group_id": null, "type": "phone"}] */
const data = [
  {
    "id": 1,
    "name": "亞洲大學校安中心-24小時校內專線",
    "tel": "04-23321053",
    "image": null,
    "map_group_id": null,
    "type": "phone"
  },
  {
    "id": 2,
    "name": "亞洲大學校安中心-24小時校外專線",
    "tel": "04-23321053",
    "image": null,
    "map_group_id": null,
    "type": "phone"
  },
  {
    "id": 3,
    "name": "亞洲大學校安中心-24小時校外專線",
    "tel": "04-23321053",
    "image": null,
    "map_group_id": null,
    "type": "phone"
  },
  {
    "id": 4,
    "name": "亞洲大學校安中心-24小時校外專線",
    "tel": "04-23321053",
    "image": null,
    "map_group_id": null,
    "type": "phone"
  },
  {
    "id": 5,
    "name": "亞洲大學校安中心-24小時校外專線",
    "tel": "04-23321053",
    "image": null,
    "map_group_id": null,
    "type": "phone"
  }
]
```

```
/* 我有一個 web api
網址是 https://edurpap2.asia.edu.tw/App\_api/api/app\_08/scr\_data
http的方法是使用get
資料結構為 [{"id": 1, "name": "亞洲大學校安中心-24小時校內專線", "tel": "04-23321053", "image": null, "map_group_id": null, "type": "phone"}]
請幫我用fetch寫一個function 去查詢
並且用表格的方式列在畫面上 */
async function getData(){
  let url = 'https://edurpap2.asia.edu.tw/App_api/api/app_08/scr_data';
  let response = await fetch(url);
  let data = await response.json();
  console.log(data);
  let str = '<table border="1">';
  for(let i=0; i<data.length; i++){
    str += '<tr>';
    str += '<td>' + data[i].id + '</td>';
    str += '<td>' + data[i].name + '</td>';
    str += '<td>' + data[i].tel + '</td>';
    str += '<td>' + data[i].image + '</td>';
    str += '<td>' + data[i].map_group_id + '</td>';
    str += '<td>' + data[i].type + '</td>';
    str += '</tr>';
  }
  str += '</table>';
  document.getElementById('data').innerHTML = str;
}
```

Github Copilot 知道欄位後，就可以產生正確的程式碼！

使用Q&A問答模式

- 直接問Github Copilot，近似於ChatGPT

```
<script>  
//請問我要怎麼還原一個 git add?  
//git reset HEAD <file>
```

The Flow

- 順著你的開流程式自動產生程式碼（命名很重要）

```
<script>  
  const current = new Date();
```

```
<script>  
  const current = new Date();  
  const isWeekend = current.getDay() === 6 || current.getDay() === 0;
```

- 不需要透過下指令的方式即可自動產生程式碼（命名很重要）

```
function dayDiff(first, second) {  
  return Math.round((second - first) / (1000 * 60 * 60 * 24));  
}
```

- Github Copilot 會根據程式的上下文（Context），自動組成GPT-4 所需的提示
-

讓Copilot 成為你的副駕駛

- 透過良好的類別、函式、參數、變數命名讓 Copilot 知道你的現狀與意圖
 - 透過當前程式的上下文自動判斷你接下來要寫的 Code
 - 開啟需要參考的程式碼並依照你的撰寫習慣去寫程式
-

Github Copilot Chat

內建三種不同的聊天視窗

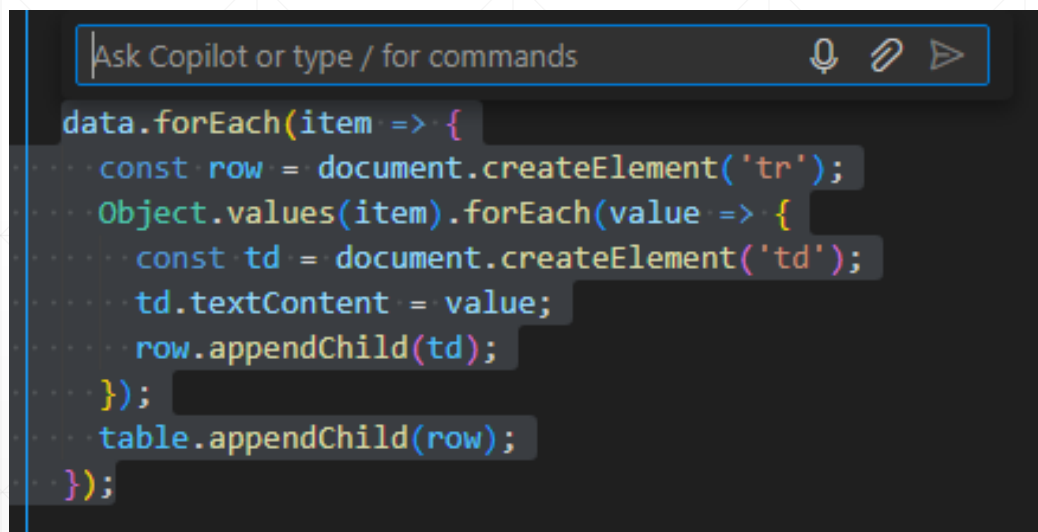
- 一般聊天視窗 (Ctrl+Alt+I)
 - 位於編輯器側邊欄
 - 具有短期記憶能力
 - 快速聊天視窗 (Ctrl+Shift+I)
 - 位於視窗上方
 - 內嵌聊天視窗 (Ctrl+I)
 - 位於游標位置
 - 開新文件時，系統建議使用此功能開始使用Copilot
-

查看聊天視窗中的slash命令 (/help)

- @workspace
 - /explain 關於您當前工作區的檔案提出問題
 - /tests 請Copilot解釋所選程式碼的運行原理
 - /fix 為所選程式產生單元測試程式碼
 - /new 請Copilot來修復您的程式碼
 - /newNotebook 為新的專案建立專案架構
 - /newNotebook 建立新的Jupyter筆記本
 - @terminal 請教如何在終端中執行某些操作
 - @vscode 提問關於 VS Code 的問題
 - /api 請教有關 VS Code 擴充套件開發的問題
 - /search 請教有關如何在 VS Code 裡面搜尋檔案
-

直接在程式碼聊起來

- 在當程式碼區塊按下 **Ctrl+I** 就可以快速啟動Code chat 視窗
 - 即可開始聊天



The image shows a code editor with a dark theme. At the top, there is a search bar with the text "Ask Copilot or type / for commands" and icons for voice search, a link, and a play button. Below the search bar, there is a JavaScript code snippet:

```
data.forEach(item => {  
  const row = document.createElement('tr');  
  Object.values(item).forEach(value => {  
    const td = document.createElement('td');  
    td.textContent = value;  
    row.appendChild(td);  
  });  
  table.appendChild(row);  
});
```

Explain 解釋程式碼 (繁體中文)

2

請問這段code是在做什麼呢

輸入問題

3

已使用 /explain (未重新執行)

Copilot辨識出需要/explain程式碼

> 已使用 1 個參考

4

這段程式碼是在使用 JavaScript 動態地將資料插入到 HTML 表格中。讓我們來解釋一下這段程式碼的運作方式：

1. `data.forEach(item => { ... });`：這是一個 `forEach` 迴圈，用於遍歷 `data` 陣列中的每個元素。在每次迭代中，`item` 代表著 `data` 陣列中的一個物件。
2. `const row = document.createElement('tr');`：這行程式碼建立了一個 `<tr>` 元素，代表 HTML 表格中的一列。
3. `Object.values(item).forEach(value => { ... });`：這是另一個 `forEach` 迴圈，用於遍歷 `item` 物件中的每個值。在每次迭代中，`value` 代表著 `item` 物件中的一個屬性值。
4. `const td = document.createElement('td');`：這行程式碼建立了一個 `<td>` 元素，代表 HTML 表格中的一個儲存格。
5. `td.textContent = value;`：這行程式碼將 `value` 的值設定為 `<td>` 元素的文字內容。
6. `row.appendChild(td);`：這行程式碼將 `<td>` 元素加入到 `<tr>` 元素中，將儲存格加入到列中。
7. `table.appendChild(row);`：這行程式碼將 `<tr>` 元素加入到名為 `table` 的 HTML 元素中，將列加入到表格中。

這樣，透過迴圈的遍歷，程式碼會根據 `data` 陣列中的每個物件，動態地建立表格的列和儲存格，並將

開始解釋程式碼

Ask Copilot or type / for commands



在聊天中檢視 關閉 ↺ ▼

1

```
data.forEach(item => {  
  const row = document.createElement('tr');  
  Object.values(item).forEach(value => {  
    const td = document.createElement('td');  
    td.textContent = value;  
    row.appendChild(td);  
  });  
  table.appendChild(row);  
});
```

選取程式碼，輸入Ctrl+I進入內嵌聊天視窗

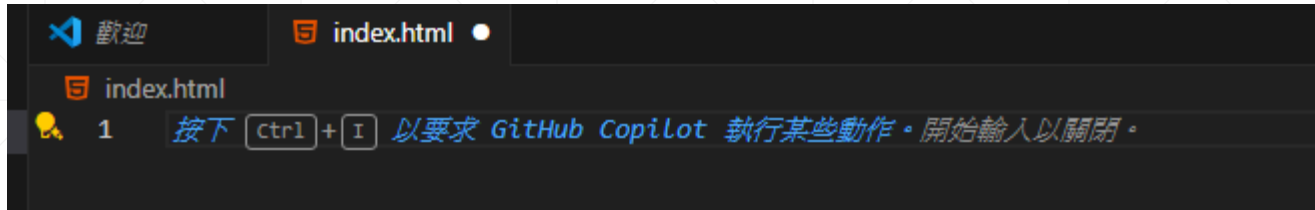
Github Copilot (Chat) 可以完成哪些工作？

- 撰寫各種**常見**的程式邏輯
 - 撰寫複雜的正則表示式 (Regular Expression)
 - 將 HTML 轉成 JavaScript 產生 DOM 的程式碼
 - 撰寫呼叫函式/方法的程式碼
 - 撰寫**特定**演算法的程式碼
 - 撰寫各種複雜 SQL 語句
 - 撰寫 OpenAPI Spec
 - 自動生成測試資料
 - 自動生成測試程式碼
 - 刪除不要的程式碼
 - 自動補完程式碼
 - 參考現有專案原始碼
 - 參考現有專案程式碼樣
-

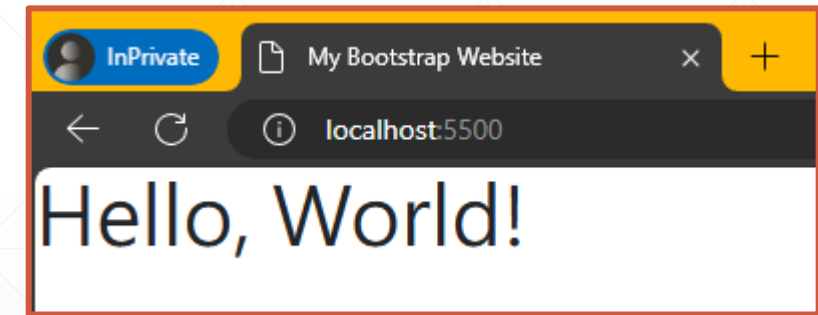
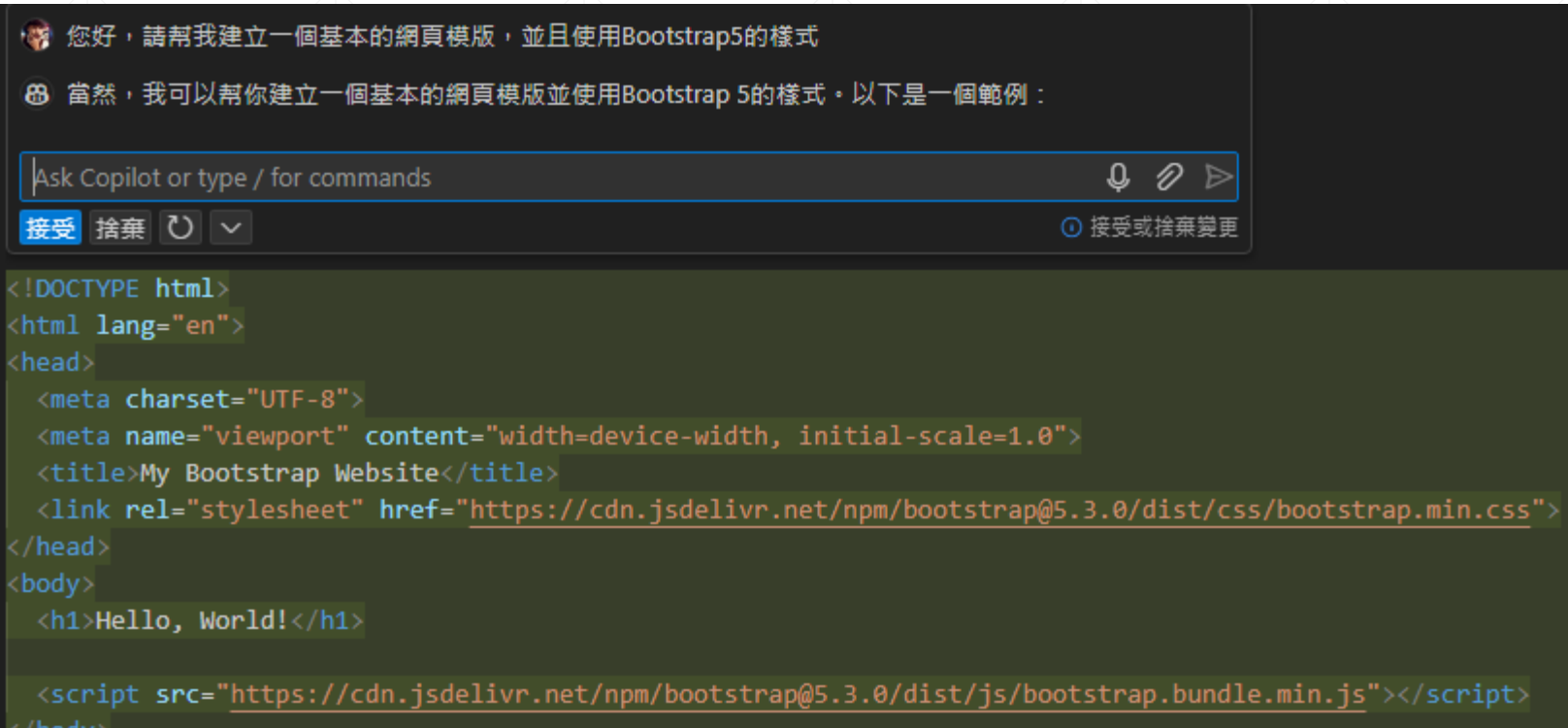
使用Github Copilot 打造一個BMI計算網頁

BMI計算網頁-1

- 新增一個index.html文件，系統提示使用 **Ctrl+I** 開啟Code Chat開始使用



- 輸入咒語「您好，請幫我建立一個基本的網頁模版，並且使用Bootstrap5的樣式」並送出。確認程式碼後，點擊「接受」。



預覽圖

BMI計算網頁-2

- 將<h1>的內容改成「BMI計算器」，按Enter到下一行，此時可以發現Github Copilot已經順著我們的Flow產生程式碼了！

```
<h1>BMI計算器</h1>
<form action="bmi.php" method="post">
  <div class="mb-3">
    <label for="height" class="form-label">身高</label>
    <input type="text" class="form-control" id="height" name="height" />
  </div>
  <div class="mb-3">
```

- 將滑鼠移到灰底字，會出現一個對話框，顯示1/3，代表Github Copilot總共產生了三組程式碼，可以點擊「>」預覽看看是否產生更適合使用的程式碼，本範例直接使用第一種，於是直接點擊「Tab」鍵。

```
<body>
  <h1>BMI計算器</h1>
  <form action="bmi.php" method="post">
    <div class="mb-3">
      <label for="height" class="form-label">身高 (公分)</label>
      <input type="number" class="form-control" id="height" name="height" required />
    </div>
    <div class="mb-3">
      <label for="weight" class="form-label">體重 (公斤)</label>
      <input type="number" class="form-control" id="weight" name="weight" required />
    </div>
    <button type="submit" class="btn btn-primary">計算</button>
  </form>
```



BMI計算器

身高 (公分)

體重 (公斤)

計算

預覽圖

BMI計算網頁-3

- 我們可以看到產生的程式碼中<form> 上的action 屬性設定為「bmi.php」，代表我們需要另外使用php來完成這個計算，但本範例想要使用JavaScript來計算即可。
- 於是我們可以將<form>中的程式碼區塊選取起來，並且鍵入 **Ctrl+I** 快速進入 Code chat，並且輸入咒語「我沒有要使用php來完成bmi計算，請幫我改成javascript的版本」然後按下「Enter」，確認程式沒問題後點擊「接受」。

```
<body>
<h1>BMI計算器</h1>
<form action="bmi.php" method="post">
  <div class="mb-3">
    <label for="height" class="form-label">身高 (公分)</label>
    <input type="number" class="form-control" id="height" name="height">
  </div>
  <div class="mb-3">
    <label for="weight" class="form-label">體重 (公斤)</label>
    <input type="number" class="form-control" id="weight" name="weight">
  </div>
  <button type="submit" class="btn btn-primary">計算</button>
</form>
```



我沒有要使用php來完成bmi計算，請幫我改成javascript的版本

已進行變更。

Ask Copilot or type / for commands

接受 捨棄 ↺ ▼ ⓘ 接受或捨棄變更

```
<form>
  <div class="mb-3">
    <label for="height" class="form-label">身高 (公分)</label>
    <input type="number" class="form-control" id="height" required />
  </div>
  <div class="mb-3">
    <label for="weight" class="form-label">體重 (公斤)</label>
    <input type="number" class="form-control" id="weight" required />
  </div>
  <button type="button" class="btn btn-primary" onclick="calculateBMI()">計算</button>
</form>

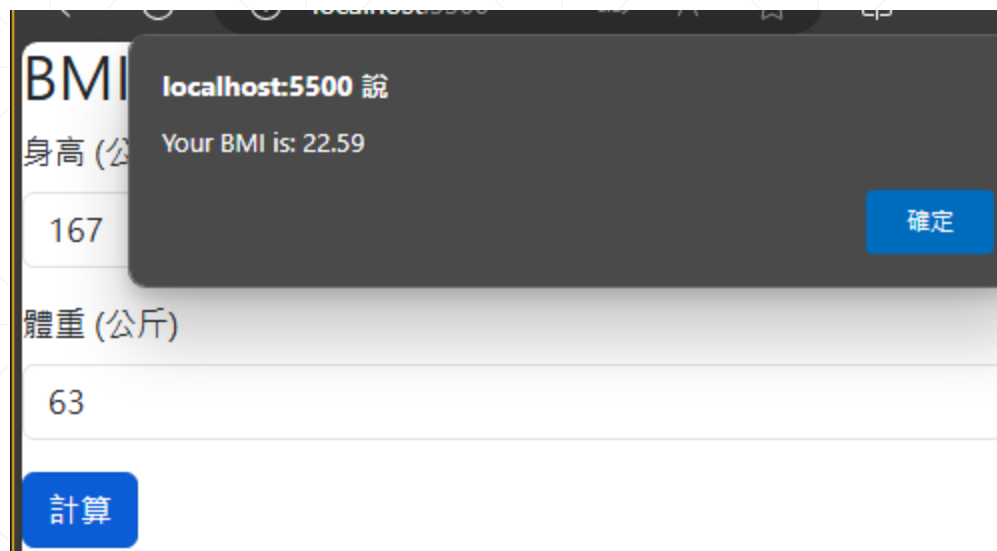
<script>
  function calculateBMI() {
    var height = document.getElementById("height").value;
    var weight = document.getElementById("weight").value;

    var bmi = weight / ((height / 100) * (height / 100));

    alert("Your BMI is: " + bmi.toFixed(2));
  }
</script>
```

BMI計算網頁-4

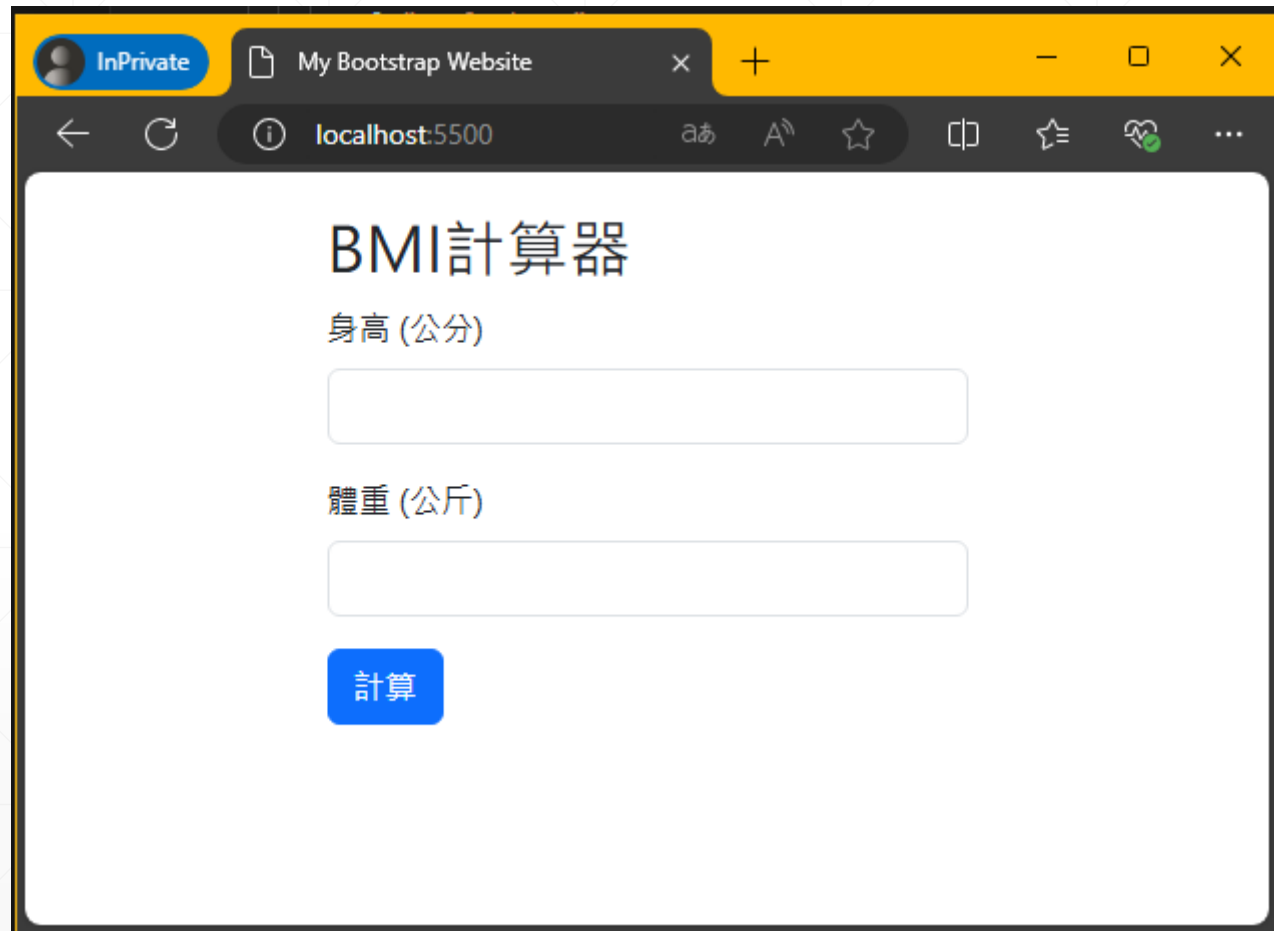
- 然後我們可以在網頁上進行測試，輸入身高167，體重63點擊「計算」按鈕後即算出BMI為22.59。



The screenshot shows a web browser window with a BMI calculator. The page has a title 'BMI'. There are two input fields: '身高 (公分)' (Height in cm) with the value '167' and '體重 (公斤)' (Weight in kg) with the value '63'. Below these is a blue button labeled '計算' (Calculate). A dark gray modal dialog is open in the center, displaying 'localhost:5500 說' (localhost:5500 says) and 'Your BMI is: 22.59'. A blue button labeled '確定' (Confirm) is in the bottom right of the modal.

- 最後，我們來把樣式做個調整，選取<form>中的內容，進入Code chat 後輸入咒語「*請讓畫面置中，並且設定最大寬度為360px，然後加入適當的留白*」，然後點擊「**接受**」。

BMI計算網頁-5



The image shows a web browser window with a yellow title bar. The address bar displays 'localhost:5500'. The page content is titled 'BMI計算器' (BMI Calculator). It features two input fields: '身高 (公分)' (Height in cm) and '體重 (公斤)' (Weight in kg). Below these fields is a blue button labeled '計算' (Calculate).

BMI計算器

身高 (公分)

體重 (公斤)

計算

完成圖

謝謝聆聽
