

## CHAPTER 09

# AI開發網站體驗

HORAZON

互動媒體設計

# 時代變了：AI 與前端開發

前端技術更新太快 (Webpack, Vite, Tailwind, TypeScript...)。  
學習曲線變得很陡峭。

**但是，AI 非常擅長寫前端！**

為什麼？

1. **結構明確**：HTML 是樹狀結構，CSS 是規則集，邏輯清晰。
2. **視覺化**：描述「我要一個藍色的按鈕，滑過去會變大」，AI 聽得懂。
3. **重複性高**：很多組件 (Navbar, Card, Form) 寫法都很像。

# 如何利用 AI 做前端？

不要手刻 CSS，不要死背語法。

流程 (Workflow)：

1. **Prompt (詠唱)**：清楚描述你的需求。
  - "幫我做一個 RWD 的產品卡片，要有陰影，圖片是圓形的。"
  - "用 Vue 寫一個待辦事項清單，可以新增和刪除。"
2. **Generate (生成)**：AI 給你程式碼。
3. **Refine (微調)**：你只需要看得懂程式碼，並做小修改。

# AI 實戰：VS CODE + GITHUB COPILOT

VS Code 是目前最流行的程式碼編輯器，搭配 GitHub Copilot 更強大。

- **智慧補全 (Ghost Text)：**
  - 當你打字時，AI 會猜你想寫什麼，按 `Tab` 即可完成。
- **Copilot Chat：**
  - 直接在編輯器內跟 AI 對話：「這段程式碼在做什麼？」、「幫我修好這個 Bug」。
- **Inline Chat (Cmd/Ctrl + I)：**
  - 選取程式碼，叫 AI 直接修改：「幫我加上註解」、「改成 Vue 的寫法」。

未來工程師的核心能力，不是「寫 code」，而是「描述需求」與「審查 code」。

# 結論

1. **AI 是最強的助手**：前端開發已經進入「AI 輔助」時代。
2. 善用工具 (VS Code + Copilot)，專注在**創意與邏輯**。
3. 髒活交給 AI，你負責**審查與組裝**。