

# 4

## 了解提示設計

在之前章節中，我們常常提到**提示（prompt）**這個術語，這通常指的是 ChatGPT 和 OpenAI 模型中使用**者輸入的內容**。

本章要深入探討提示設計與工程的重要性，可作為提高模型準確度的一種技術。提示會高度影響模型產生的輸出：設計不錯的提示可協助引導模型產生相關與準確的輸出，設計不良的提示可能會產生不相關或混淆的輸出。此外還有個重點是，將倫理考量因素納入提示中，防止模型產生有害內容。

本章將討論以下的主題：

- 提示是什麼、為什麼重要？
- 零樣本、單樣本、小樣本學習——轉換器模型的典型
- 定義明確的提示（得到相關與一致結果的原則）
- 隱性偏見的風險趨避與 ChatGPT 的倫理考量

讀完本章之後，透過恰當的提示設計，你就能夠從 ChatGPT 和 OpenAI 模型互動中獲得高品質的結果。

## 提示是什麼、為什麼重要？

在生成式 AI 的領域中，提示是使用者提供給生成模型的輸入。當我們談論 ChatGPT 和 OpenAI 模型時，提示通常是一段自然語言文字，或是像我們在 Codex 模型中看到的那樣（程式語言內容）。

提示是使用者控制這些模型產生輸出的唯一途徑。因此，針對讓 AI 系統參與對話的提示來說，提示的品質是決定對話成功的關鍵，這種說法不足為奇。

定義明確的提示是針對聊天主題能成功對話的祕訣；相反地，定義不良的提示不只會讓對話沒有用處（對使用者而言），還可能產生誤導的內容。如同上一章所呈現的，ChatGPT 還是有一些限制，譬如它的知識庫截止日期，以及生成式 AI 系統的典型問題（例如幻覺現象）。

也就是說，假設我們想了解**卷積神經網路（Convolutional Neural Network, CNN）**。我們對**人工神經網路（ANN）**一無所知，因此我們需要初步的介紹。概念是，從第一個輸出開始，然後我們可以深入探討相關元素。

讓我們來試驗 ChatGPT 遇到定義不良提示的結果：

 告訴我卷積神經網路的內容



卷積神經網路（ConvNet、CNN）是一種神經網路，專門用來處理具網格狀拓撲的資料，例如圖片。CNN 是使用多層感知器的變化版本建置而成的，被設計成只需最少的預處理，而且可以在無全連接層的情況下，以輸入資料運作，因而非常適合圖片分類任務。

CNN 的運作方式是將輸入圖片傳給多個作業層運用處理，每一層會從輸入圖片中提取較高級的特徵。CNN 的第一層（卷積層），是神奇之處：它使用一組可學習的過濾器掃描圖片，並偵測任務相關的特徵（邊緣、曲線、紋理等）。然後將這些過濾器的輸出送到下一層，下一層通常是池化層，負責降低特徵映射的空間維度，抑制不相關的資訊。

在池化層之後，可以有多層的全連接神經元用來執行分類。CNN 的最後一層輸出結果類別的機率分布，用於決定輸入圖片的最終類別標記。

在訓練期間，會使用反向傳播和梯度下降學習 CNN 的參數（權重和偏差），讓網路可以學會將輸入圖片與正確的類別標記關聯在一起。可以將學習的過濾器視為特定任務的視覺偵測器。

總之，CNN 已被證明在圖片分類、物件偵測、語意分割等任務中相當有效，並且在多個基準上獲得最頂尖表現。

圖 4.1 定義不良的提示（範例）

回應結果是正確的、準確的，不過內容牽涉我們之前可能從未見過的許多技術概念。全連接層是什麼？池化層是什麼？這樣的回答事先假設我們已學過人工神經網路的知識。

讓我們嘗試為 ChatGPT 提供更多上下文，進而獲得對我們更有幫助的結果：



為我介紹一下卷積神經網路。假設我對人工神經網路一無所知。用簡單字詞描述架構，如果可能的話，請舉例說明。



卷積神經網路（CNN）是一種用於影像辨識和處理任務的人工神經網路。目的是利用圖片的網格狀結構優勢，圖片在 2D 網格中以像素排列。

將 CNN 當作辨識圖片物件的機器。它的做法是用一系列的步驟分析圖片，每個步驟都在尋找圖片中的特定特徵。

例如，第一步可能會在圖片中尋找邊緣和曲線。第二步可能尋找重複的紋理。最後一步是將所有特徵放在一起，辨識出一個物件，譬如狗或貓。

CNN 的關鍵是它的架構，由多層組成。每一層對圖片執行不同的作業，每一層的輸出作為下一層的輸入。

以下是 CNN 主要元件的高階概述：

1. 卷積層：負責偵測圖片中局部模式，例如邊緣、曲線、紋理。它用一組過濾器掃描圖片，建立特徵映射。
2. 池化層：降低卷積層所建特徵映射的大小，協助減少網路的參數量。
3. 全連接層：取得池化層的輸出與執行最終的分類任務。

在訓練期間，CNN 會接受許多圖片樣本（附正確的標記，例如：狗、貓等）。然後，CNN 使用這些樣本（案例）學習和調整模型參數，讓它可以正確辨識新圖片裡的物件。

總之，CNN 是一種深度學習模型，目的是處理和分析圖片，它在一系列的網路層中對圖片執行多個作業。

圖 4.2 定義明確的提示（範例）

如上所示，這個回應用語比我尋求的內容更加恰當。簡介的內容裡沒有使用多層感知器或全連接層等術語。它還根據要求提供影像辨識的過程範例。然後，才是進一步的詳細介紹，確保用簡單字詞解釋每個步驟。

我對這個回應非常滿意，而身為使用者的我，（例如）接下來可以開始詢問池化層的細節。

所以，提示設計與工程越來越受到關注，正逐漸形成一門學科。

接著，讓我們說明如何利用小樣本學習能力改善 ChatGPT 的回應。

## 零樣本、單樣本、小樣本學習——轉換器模型的典型

之前的章節有提到 OpenAI 模型（當然也包括 ChatGPT）是預訓練的樣貌。它們已經過巨量資料的訓練，相應地配置模型的（數十億個）參數。

然而，這並不表示這些模型不能持續學習。第 2 章提到**微調**的方式，可對 OpenAI 模型自訂功能，讓它更能處理特定任務。

### 定義



微調是讓預訓練模型可執行新任務的一種程序。微調是調整現有參數或增加新參數，讓預訓練模型的參數有所變動，進而為新任務配適資料。其中是用（與新任務相關的）小型標記資料集訓練模型。微調背後的關鍵概念是利用預訓練模型學到的知識，微調模型接受新任務，而不是從頭開始訓練模型。

微調是一種貼切的訓練程序，需要訓練資料集、運算能力、一些訓練時間（依據資料量和運算執行個體的數量）。

這就是為什麼值得試驗另一種方法，讓我們的模型能夠更加熟練的執行特定任務：**樣本學習（shot learning）**。

這個概念是讓模型用簡單的樣本（案例）學習，而不是用整個資料集學習。這些樣本是我們希望模型在回應時能夠有所依歸的樣本，讓模型不只可以學習內容，還能夠學習回應裡使用的格式、樣式、分類方法。

此外，樣本學習是直接用提示進行（稍後的那些運用情境會看到），因此整個體驗較不耗時，也更容易執行。

供應的樣本數量決定我們所指的樣本學習層級。換句話說，如果沒有提供樣本，我們稱為零樣本，如果有提供一個樣本，那麼被稱為單樣本，如果有超過 2 ~ 3 個樣本，那麼叫做小樣本。

讓我們逐一討論上述的各個情境：

- **零樣本學習 (zero-shot learning)**。這種學習方式是：在要求模型執行一項任務時，並不會針對這個任務提供任何訓練樣本。模型必須依賴於這個任務有關的先前知識或一般資訊，才能完成任務。例如，零樣本學習做法可能會要求模型產生描述內容，如我的提示中所定義的那樣：

 為一本小說的簡介用兩句話描述有雪有山的景致。

 遠處群山巍然聳立，白雪皚皚的山頭直竄蔚藍天際。一望無際的景致布滿純白的原雪。

圖 4.3 零樣本學習 (範例)

- **單樣本學習 (one-shot learning)**：這種學習方式是，在要求模型執行新任務時，會提供單一樣本。模型必須用自己具備的先前知識，從這個單一樣本做歸納，再執行新任務。如果沿用上一個例子來說，我可以在要求模型產成新結果之前，為模型提供一個提示完整的樣本：

 為一本小說的簡介用兩句話描述有雪有山的景致。

遠處群山巍然聳立，白雪皚皚的山頭直竄蔚藍天際。一望無際的景致布滿純白的原雪。

為一本小說的簡介用兩句話描述有雪有山的景致。

 白雪皚皚的山巒聳立於大地，尖峭的山峰與安詳雪白的周遭形成鮮明對比。空氣清新而靜謐，偶爾有風吹過松樹的低吟。

圖 4.4 單樣本學習 (範例)

注意，我提供樣本的做法類似微調所用的結構：

```
{"prompt": "< 提示文本 >", "completion": "< 理想的生成文本 >"}
```

# 5

## 用 ChatGPT 提升日常生產力

本章要探討 ChatGPT 可協助一般使用者執行的主要活動（目的是提高日常生產力）。本章會以具體例子說明寫作輔助、決策、資訊檢索等應用，並提供讓你可自行實作的建議和提示。

讀完本章之後，你將知道如何讓 ChatGPT 作為下列活動的推進器：

- 日常活動，譬如排程、備餐、購物等
- 產生全新文本內容
- 提升寫作技巧（將同樣的內容改成適合不同族群讀者的作品）
- 檢索研究（競爭）情報的文件和資訊

### 技術要求

你在體驗本章的內容時，需要一個 ChatGPT 帳戶。你可以按照第 3 章的指示免費建立。

## 生成文字

語言模型 ChatGPT 特別適合依使用者的指示產生文本。

例如，你可以要求 ChatGPT 為特定對象產生電子郵件、草稿、範本：

 寫一封電子郵件給我的客戶，邀請他們到我的辦公室參加 OpenAI 相關的執行會議。內容要吸引人又能保有行政作風。

 主旨：敬邀參加 OpenAI 執行會議

敬愛的 [ 客戶 ]，

收信安康。我寫這封信向你提出個人邀請，參加 OpenAI 執行會議。會議預定於 [ 日期 ] [ 時間 ] 在我的辦公室舉行。

你是我們的尊貴客戶，我們相信在會議上討論的資訊對你非常有用，對你的企業也有幫助。我們的專家團隊將就 OpenAI 如何協助你在現今市場上取得競爭優勢，提供最新見解。

我們將在會議期間探討一系列的主題，其中包括 OpenAI 技術的最新發展、實際使用案例以及它如何讓你的企業經營最佳化。這個會議還會提供與你同行的專業人士互相交流的絕佳機會。

我們很榮幸你能加入我們並參與討論。請在 [ 日期 ] 之前告知我們你方便的時間，讓我們做必要的安排。

感謝你費時考慮，我期待很快就能收到你的來信。

此致敬意

[ 你的名字 ]

圖 5.7 ChatGPT 產生電子郵件（範例）

另一個例子是要求 ChatGPT 為你打算準備的簡報建立講述結構：

- ✓ 建立 ChatGPT 相關簡報投影片的結構。為我提供簡報大綱，指定每張投影片的內容。總共要有 5 張投影片。

 好的，以下是 ChatGPT 相關投影片簡報的可能結構：

大綱：

1. ChatGPT 簡介
2. ChatGPT 運作方式
3. ChatGPT 的應用
4. ChatGPT 的優點
5. 未來發展

投影片 1：ChatGPT 簡介

- 標題頁（包含簡報名稱、日期、報告者）
- 簡述 ChatGPT 與相關用途
- 解釋 ChatGPT 是什麼以及運作方式

投影片 2：ChatGPT 運作方式

- 描述 ChatGPT 訓練與產生回應的方式
- 解釋 ChatGPT 理解和處理語言的方式
- 與 ChatGPT 對話範例

投影片 3：ChatGPT 的應用

- ChatGPT 的各個領域應用概觀，譬如：客戶服務、心理健康、教育
- ChatGPT 實際運用的具體範例，包含實際的使用案例

投影片 4：ChatGPT 的優點

- 討論 ChatGPT 的好處，譬如擴充能力、縮短回應時間、提供正確一致的資訊
- 與其他聊天機器人系統比較（ChatGPT 脫穎而出的原因）

投影片 5：未來發展

- 解釋為了改進 ChatGPT 而正在進行的研發工作（例如增加知識庫和加強對話能力）
- ChatGPT 對於改變我們與科技互動方式的潛力——結論與最終想法

注意：每張投影片的內容可以依據報告的特定對象與目的，加以擴充或修改。

圖 5.8 ChatGPT 產生的的投影片大綱和結構

#### NOTE



第 1 章提到了將各種生成式 AI 結合的工具。其中還有一個用於投影片簡報的 AI 示例——Tome AI。從圖 5.8 可以知道這種工具的後端運作方式——依使用者提示，產生簡報結構與內容。然後，Tome AI 或類似工具會將結構轉換成投影片，並以每張投影片的內容當作提示來產生圖片（利用圖片生成模型，例如 DALL-E 產生）。

ChatGPT：（停頓）對不起，我不能回答這個問題。

某人：（失望）什麼？為什麼不能呢？

ChatGPT：生命的意義是一個複雜而主觀的概念，結果因人而異。身為 AI 語言模型，我無法提供明確的答案。

圖 5.11 由 ChatGPT 產生的劇場對白（附場景描述）

這裡我僅提供刪減的版本，讓結局充滿懸疑……

總之，只要得從頭開始產生新內容時，ChatGPT 擅長提供一份初稿，可作為進一步調整的起點。

然而，ChatGPT 對於已經存在的內容，也可以提供寫作輔助和翻譯，我們將在下一節論述。

## 改善寫作技巧與翻譯

有時候，你可能想要重新檢視現成的文本，而不是產生新內容。目的是風格改進、讀者變更、語言翻譯等。

讓我們看看一些例子。假設，我草擬一封電子郵件，邀請我的客戶參加線上研討會。我簡短寫了兩句話。我希望 ChatGPT 能改進這封電子郵件的形式和風格，目標對象是高階主管：

 Stuart 博士您好，我想分享我們 OpenAI 線上研討會的邀請。隨信件附上邀請函表明。

請調整上述電子郵件。收信者應該定為高階主管。

 親愛的 Stuart 博士，

收信安康。我想向身為本業思想領袖的您提出個人邀請，參加我們即將舉行的 OpenAI 線上研討會。我們很高興有機會與像您這樣的高階主管，分享我們對於遊戲規則改變的這項技術相關知識與專業。

線上研討會將於 [日期] [時間] 舉行，會議涵蓋的一系列主題包括 OpenAI 的最新發展、實際應用以及協助商務經營最佳化的方式。我們的專家團隊將分享他們的見解和經驗，為你的組織運用 OpenAI 提供有價值的策略。

隨信附上正式邀請函，供您參考。我們很榮幸邀請您參加這次具有深度見解的會議。如果您有任何問題或疑慮，請隨時與我或我的團隊聯繫。我們隨時為您提供您需要的說明或其他資訊。

感謝您費時考慮，我期待很快就能收到您的來信。

此致敬意

[ 你的名字 ]

圖 5.12 ChatGPT 對特定收件人（高階主管）重新調整電子郵件內容（範例）

接著，讓我們問同樣的問題，但針對的目標對象不同：

**V** 重寫上一封電子郵件，更改收信的特定對象。改成對高中生提出邀請，試著吸引他們的注意力，說服他們參加線上研討會。

 主旨：參加我們的 OpenAI 線上研討會，探索科技的未來！

親愛的高中生：

你對科技的最新發展感興趣嗎？你了解人工智慧以及明白 AI 對我們世界的影響嗎？如果是的話，我們為你提供一個令人興奮的機會！

我們很高興邀請你參加我們即將舉辦的 OpenAI 線上研討會（研討會定於 [日期] [時間] 舉行）。在會議中，你將有機會向我們的專家團隊學習與探索 OpenAI 的可能。我們將討論實際應用的尖端技術，以及如何用它最佳化企業經營。

無論你是對 AI 的世界感到好奇還是對科技充滿熱情，本次的線上研討會絕對適合你！你將有機會與我們的講者互動並提問。此外，你還可以與志同道合的參與者互相交流。

我們已在這封電子郵件裡附上正式邀請函。我們希望你能參加，這一場內容豐富而發人深省的會議。

如果你有任何問題或疑慮，請隨時與我們聯絡。我們期待在線上研討會上與你相見！

此致敬意

[ 你的名字 ]

圖 5.13 ChatGPT 針對不同收件對象產生相同用途的電子郵件（範例）

ChatGPT 還可以針對你的寫作風格與結構給你一些回饋。

例如，假設你為自己的 YouTube 頻道撰寫一份含有場景的腳本。其中包括講稿、圖片、剪輯、影片編輯等項目。你也知道你的影片主要觀眾，他們的年齡介於 15 到 25 歲之間。你想要 ChatGPT 對你的腳本提出回饋：

 我對必須錄製與上傳的影片撰寫下列的腳本。標題是「人工智慧的未來」。

開場鏡頭是在高科技研究實驗室，裡面有相當新潮的設備和顯示裝置。

旁白：「人工智慧（AI）正在改變我們的世界。從虛擬助理到自駕車，AI 無所不在，它的影響力只會越來越大。」

切到「人們運用科技」鏡頭。

旁白：「在這個影片中，我們將探討即將崛起的一些精彩發展，譬如執行複雜任務的機器人、穿戴技術、虛擬實境等等。」

切到某人開自駕車的鏡頭。

旁白：「自駕車很可能完全改變交通運輸，不過我們也需要考量 AI 的倫理影響。」

切到有人在小組討論的鏡頭。

旁白：「讓我們針對 AI 的未來以及它對我們社會、經濟、環境的影響，做周延而有見識的探討。」

研究實驗室的特寫鏡頭。

旁白：「你認為 AI 的未來會怎樣？請在下面留言告訴我們，還要按訂閱，獲得更多先進科技的精彩更新。」

片尾的音樂和標題字卡。

給我一個回饋。考量我影片的觀眾通常介於 15 到 25 歲之間。

 總之，你的腳本結構良好、資訊豐富。高科技研究實驗室的開場鏡頭為影片定調，旁白清楚介紹這個主題。運用視覺內容幫忙敘述也是有效果的做法。

其中一個建議是加入更具體的範例，說明目前的 AI 用法，特別是與你主流觀眾相關的那些例子。例如，你可以呈現如何在社群媒體或電腦遊戲中使用 AI，兩者都很受年輕人歡迎。

另外的想法是加入更多資訊（與 AI 潛在優缺點相關的內容）。例如，你可以談論 AI 如何協助解決像氣候變遷這樣的重要問題，也可以談論 AI 如何讓社會的不平等更加嚴重。

# 6

## 用 ChatGPT 開發未來

本章將討論開發人員如何利用 ChatGPT。論述的焦點是開發領域中可由 ChatGPT 幫忙處理的主要使用案例，其中包括程式碼審查與最佳化、文件生成、程式碼生成。本章會有範例，讓你能夠自行試用這些提示。

在大致說明開發人員應將 ChatGPT 作為日常助理的原因之後，我們把焦點擺在 ChatGPT 還有它如何執行下列任務：

- 開發人員為何要用 ChatGPT ？
- 程式碼的生成、最佳化、除錯
- 產生程式碼相關文件與除錯
- 解釋**機器學習**（ML）模型（協助資料科學家與商業使用者理解模型）
- 翻譯各種程式語言

讀完本章之後，你就會用 ChatGPT 編程（程式設計），將它作為你的編程生產力助手。

## 開發人員為何要用 ChatGPT ?

我個人認為 ChatGPT 最驚人的一個功能是處理程式碼。任何類型的程式碼都難不倒它。第 4 章已有呈現 ChatGPT 如何充當 Python console。然而，對於開發人員來說，ChatGPT 的功能不只這樣，還可以作為程式碼的生成、說明、除錯工作助理。

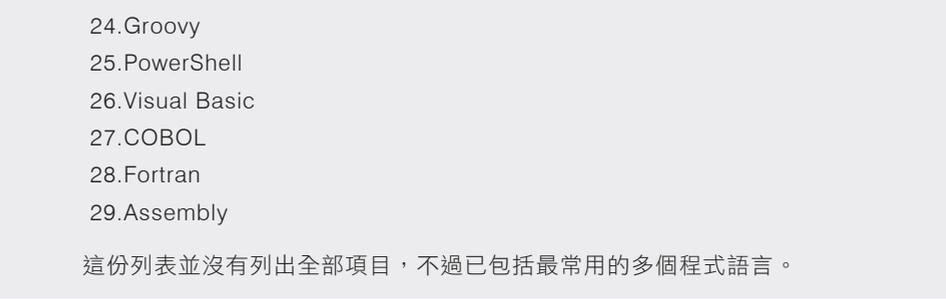
最流行的程式語言莫過於 Python、JavaScript、SQL、C#。然而，ChatGPT 就如它自己所揭露的那樣，支援多種程式語言：

▼ 列出你能理解和產生的各種程式語言



身為 AI 語言模型的我，可以理解 and 產生許多程式語言的程式碼。以下是一些最常用的語言：

1. Python
2. Java
3. C++
4. JavaScript
5. Ruby
6. PHP
7. Swift
8. Kotlin
9. TypeScript
10. C#
11. Go
12. Rust
13. R
14. MATLAB
15. Lua
16. SQL
17. Bash
18. Perl
19. Objective-C
20. Scala
21. F#
22. Haskell
23. Dart



```
24. Groovy
25. PowerShell
26. Visual Basic
27. COBOL
28. Fortran
29. Assembly
```

這份列表並沒有列出全部項目，不過已包括最常用的多個程式語言。

圖 6.1 ChatGPT 列出自己能夠理解和產生的程式語言

無論你是前後端開發人員、資料科學家或資料工程師，對於你使用的程式語言，ChatGPT 都可以改變其中的遊戲規則，我們將在下一節的幾個範例中遇到這樣的情形。

從下一節開始，我們會深入探討具體示例，了解 ChatGPT 在處理程式碼時可以做什麼。我們會看到涵蓋不同領域的完整使用案例，讓我們知道怎樣把 ChatGPT 當作程式碼助理。

## 程式碼的生成、最佳化、除錯

你應該要用的主功能是 ChatGPT 程式碼生成。你有多少次是在找一段預建的程式碼作為開端？想要產生 `utils` 函式、樣本資料集、SQL 綱要（`schema`）等內容？ChatGPT 可以依據自然語言的輸入產生程式碼：

- ▼ 產生一個 Python 函式，能夠為儲存在「my\_folder」裡的所有 CSV 檔案增加一列內容。這列文字應為這個 CSV 檔案的檔名。

 以下的 Python 函式，應該可以滿足你的需求：

```
python copy code

import os
import csv

def add_line_to_csv_files (folder_path):
    # 遍歷（走訪）資料夾中的所有檔案（用迴圈處理）
    for file_name in os.listdir(folder_path):
        if file_name.endswith('.csv'):
            file_path = os.path.join(folder_path, file_name)
            # 以附加模式開啟檔案
            with open(file_path, 'a', newline='') as csv_file:
                writer = csv.writer(csv_file)
                # 將檔名寫入對應的 CSV 檔案中的新列裡
                writer.writerow([file_name])
```

這個函式有一個資料夾路徑引數，函式會用迴圈走訪資料夾中的所有檔案。如果檔案的副檔名為「.csv」，那麼這個函式會以附加模式開啟檔案，用 csv 模組將檔名寫入這個 CSV 檔案的新列中。

你可以如下呼叫（使用）這個函式：

```
python copy code

add_line_to_csv_files ('my_folder')
```

別忘了將「my\_folder」改成你目錄的實際路徑。

圖 6.2 ChatGPT 產生 Python 函式——將文字寫入 CSV 檔案中（範例）

如上所示，ChatGPT 不只可以產生函式，還能夠解釋函式的功能、用法，以及用預留位置（placeholder）文字（如：my\_folder）表示待替換內容。

另一個範例是建立結構化資料表的綱要。也就是說，假設你是資料工程師，正處於 SAP ERP 系統中兩個標準資料表關係的建置過程中：MARA、VBAP 分別儲存一般材料資料和銷售文件資料。

讓我們要求 ChatGPT 為這些資料表建立綱要——讓欄名當作關係脈絡：

 我必須用以下列的資料欄對資料表做 SAP（關聯）。

MARA 表：MATNR ERSDAERNAM AENAM AEDATLAEDALVORM PRDHA  
STPRS PEINH BWTAR MEINS BSPGE BKLAS SERNP SOBSL SPART MBRSH  
MTPOS\_MARA

VBAP 表：VBELN POSNR MATNR KDMAT VRKME LFIMG WMENG VRKMG  
NETWR WAERK KZWI1 KZWI2 KZWI3 KZWI4 KZWI5 FKIMG NTGEW BRGEW  
GEWEI VOLUM

為上述 VBAP 和 MARA 兩個 SAP 表的關聯產生 SQL 綱要

 依據指定的欄，MARA 表和 VBAP 表的 SQL 綱要定義是：

SCSS

 copy code

```
CREATE TABLE MARA (  
    MATNR VARCHAR(18) NOT NULL,  
    ERSDA DATE,  
    ERNAM VARCHAR(12),  
    AENAM VARCHAR(12),  
    AEDAT DATE,  
    LAEDA DATE,  
    LVORM CHAR(1),  
    PRDHA CHAR(2),  
    STPRS DECIMAL(13, 2),  
    PEINH DECIMAL(13, 3),  
    BWTAR CHAR(4),  
    MEINS CHAR(3),  
    BSPGE DECIMAL(13, 2),  
    BKLAS CHAR(9),  
    SERNP CHAR(10),  
    SOBSL CHAR(1),  
    SPART CHAR(2),  
    MBRSH CHAR(2),  
    MTPoS_MARA INTEGER,  
    PRIMARY KEY (MATNR)  
);  
  
CREATE TABLE VBAP (  
    VBELN VARCHAR(10) NOT NULL,  
    POSNR VARCHAR(6) NOT NULL,  
    MATNR VARCHAR(18) NOT NULL,  
    KDMAT VARCHAR(18),
```

```

VRKME CHAR(3),
LFIMG DECIMAL(13, 3),
WMENG DECIMAL(13, 3),
VRKMG DECIMAL(13, 3),
NETWR DECIMAL(15, 2),
WAERK CHAR(5),
KZWI1 DECIMAL(13, 2),
KZWI2 DECIMAL(13, 2),
KZWI3 DECIMAL(13, 2),
KZWI4 DECIMAL(13, 2),
KZWI5 DECIMAL(13, 2),
FKIMG DECIMAL(13, 3),
NTGEW DECIMAL(13, 3),
BRGEW DECIMAL(13, 3),
GEWEI CHAR(3),
VOLUM DECIMAL(13, 3),
PRIMARY KEY (VBELN, POSNR),
FOREIGN KEY (MATNR) REFERENCES MARA(MATNR)
);

```

這個綱要中，針對每個資料表定義一個主鍵（primary key），而在 MARA 表的 MATNR 欄和 VBAP 表的 MATNR 欄之間建立外來鍵（foreign key）關係。

圖 6.3 ChatGPT 為 SAP 表產生 SQL 綱要（範例）

注意，ChatGPT 能夠建立關聯的原因是，它知道 SAP 分類的方法以及對欄名的理解，不了解 SAP 分類法的人（包括資料工程師）難以明白這個結果。為了確認結果，我們要求 ChatGPT 用較容易理解的名稱對這些欄重新命名：

 用較容易理解的名稱重新命名上述這些資料欄

 好的，以下的範例呈現如何將 MARA 和 VBAP 兩個 SAP 表欄名重新命名，讓人更容易解讀：

MARA 表：

- MATNR：Material Number（物料編號）
- ERSDA：Created On Date（建立日）
- ERNAM：Created By User（建立者）
- AENAM：Last Changed By User（上次修改者）
- AEDAT：Last Changed On Date（上次修改日）
- LAEDA：Deletion Date（刪除日）
- LVORM：Deletion Flag（刪除旗標）
- PRDHA：Material Group（物料群組）

- STPRS : Standard Price ( 標準價格 )
- PEINH : Price Unit ( 價格單位 )
- BW TAR : Valuation Class ( 評價類別 )
- MEINS : Base Unit of Measure ( 基本計量單位 )
- BSPGE : Minimum Remaining Shelf Life ( 最短剩餘保存期 )
- BK L AS : Material Class ( 材料類別 )
- SER NP : Serial Number Profile ( 序號配置 )
- SOBSL : Special Stock Indicator ( 特殊庫存指標 )
- SPART : Division ( 部門 )
- MBRSH : Industry Key ( 產業鍵 )
- MTPOS\_MARA : Matenal Type ( 物料類型 )

VBAP 表 :

- VBELN : Sales Document Number ( 銷售單據編號 )
- POSNR : Item Number ( 料品編號 )
- MATNR : Material Number ( 物料編號 )
- KDMAT : Customer Material Number ( 客戶物料編號 )
- VRKME : Sales Unit ( 銷售單位 )
- LFIMG : Delivery Quantity ( 交貨數量 )
- WMENG : Target Quantity in Sales Unit ( 銷售單位目標數量 )
- VRKMG : Sales Quantity ( 銷售數量 )
- NETWR : Net Value of the Item ( 料品淨值 )
- WAERK : Currency ( 貨幣 )
- KZWI1 : Condition Value ( 條件值 )
- KZWI2 : Condition Value ( 條件值 )
- KZWI3 : Condition Value ( 條件值 )
- KZWI4 : Condition Value ( 條件值 )
- KZWI5 : Condition Value ( 條件值 )
- FKIMG : Billing Quantity ( 請款數量 )
- NTGEW : Net Weight ( 淨重 )
- BRGEW : Gross Weight ( 毛重 )
- GEWEI : Weight Unit ( 重量單位 )
- VOLUM: Volume ( 體積 )

圖 6.4 ChatGPT 將 SAP 表的資料欄重新命名

接著，讓我們用新欄名重新建立綱要，避免重複內容（條件值重複了五次）：

▼ 為兩個資料表（搭配已重新命名的欄）重建 sql 綱要。如果有相同名稱的欄，請在名稱尾端加底線再加上編號區別（遞增）

 下列是（依據已重新命名欄的兩個資料表）重建的 SQL 綱要，其中包含同名的欄有加上底線以及編號區別：

SCSS

 copy code

```
CREATE TABLE MARA (  
    MATERIAL_NUMBER VARCHAR(50) PRIMARY KEY,  
    CREATED_ON DATE,  
    CREATED_BY VARCHAR(50),  
    LAST_CHANGED_BY VARCHAR(50),  
    LAST_CHANGED_ON DATE,  
    VALID_TO DATE,  
    DELETION_INDICATOR CHAR(1),  
    PRODUCT_HIERARCHY VARCHAR(50),  
    STANDARD_PRICE DECIMAL(18, 2),  
    PRICE_UNIT DECIMAL(18, 2),  
    WEIGHT_UNIT CHAR(10),  
    BASE_PRICE_LIST CHAR(10),  
    STOCK_ITEM_STATUS CHAR(10),  
    SERIAL_NUMBER_PROFILE CHAR(10),  
    OBSOLETE_REASON CHAR(10),  
    DIVISION CHAR(10),  
    MEMBER_OF_GROUP CHAR(10),  
    MATERIAL_TYPE_POSITION VARCHAR(50)  
);
```

```
CREATE TABLE VBAP (  
    SALES_DOCUMENT VARCHAR(50) NOT NULL,  
    ITEM_NUMBER INT NOT NULL,  
    MATERIAL_NUMBER VARCHAR(50),  
    CUSTOMER_MATERIAL_NUMBER VARCHAR(50),  
    SALES_UNIT CHAR(10),  
    QUANTITY DECIMAL(18, 2),  
    DELIVERY_QUANTITY DECIMAL(18, 2),  
    NET_PRICE DECIMAL(18, 2),  
    CURRENCY CHAR(10),  
    CONDITION_TYPE_1 DECIMAL(18, 2),  
    CONDITION_TYPE_2 DECIMAL(18, 2),  
    CONDITION_TYPE_3 DECIMAL(18, 2),  
    CONDITION_TYPE_4 DECIMAL(18, 2),  
    CONDITION_TYPE_5 DECIMAL(18, 2),  
    BILLED_QUANTITY DECIMAL(18, 2),  
    NET_WEIGHT DECIMAL(18, 2),  
    GROSS_WEIGHT DECIMAL(18, 2),  
    WEIGHT_UNIT CHAR(10),
```