

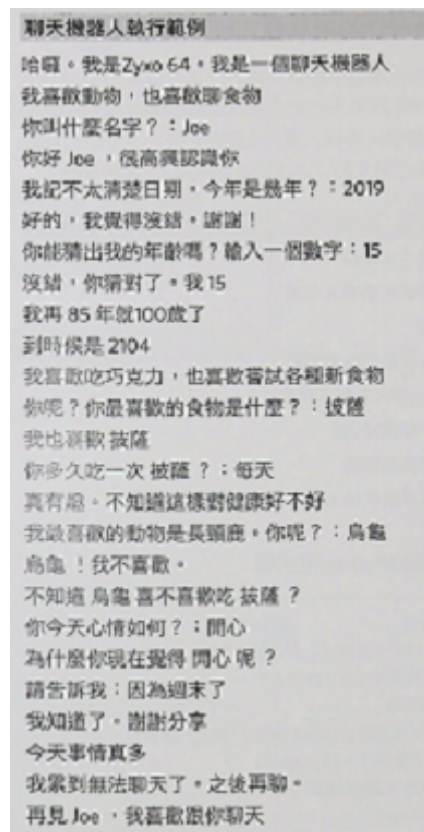
# 專題一：打造你的聊天機器人

by 田弘華 Hung-Hua Tien

## 1. 專案：打造你的聊天機器人

現在我們要做一個聊天機器人(chatbot)，看似能用文字和人類對話的程式。也就是說，利用上述主要概念，我們會取得使用者輸入並把資訊顯示在螢幕來回應使用者。這個聊天機器人會比較簡單，隨著所學的增加，日後同學可以繼續擴充。

你可以自訂聊天機器人回應和問題的文字。



聊天機器人執行範例

哈囉。我是Zyko 64，我是一個聊天機器人  
我喜歡動物，也喜歡吃食物  
你叫什麼名字？：Joe  
你好 Joe，很高興認識你  
我記不太清楚日期，今年是幾年？：2019  
好的，我覺得沒錯。謝謝！  
你能猜出我的年齡嗎？輸入一個數字：15  
沒錯，你猜對了。我 15  
我再 85 年就100歲了  
到時候是 2104  
我喜歡吃巧克力，也喜歡嘗試各種新食物  
你呢？你最喜歡的食物是什麼？：披薩  
我也喜歡披薩  
你多久吃一次披薩？：每天  
真有趣。不知道這樣對健康好不好  
我最喜歡的動物是長頸鹿。你呢？：烏龜  
烏龜！我不喜歡。  
不知道烏龜喜不喜歡吃披薩？  
你今天心情如何？：開心  
為什麼你現在覺得開心呢？  
請告訴我：因為週末了  
我知道了。謝謝分享  
今天事情真多  
我累到無法聊天了。之後再聊。  
再見 Joe，我喜歡跟你聊天

## 第1步：為程式碼建立新檔案

建立新檔案的步驟示範。

- 點擊File（檔案）->New File（新增檔案）
- 在新視窗輸入一個註解
- 按下File(檔案) -> Save As（另存新檔），檔案名稱為chatbot.py。
- 按下Run（執行）-> Run Module（執行模組）就能執行程式碼。
  - 終端機視窗，則請輸入python chatbot.py 或 python3 chatbot.py。

## 第2步：加入自我介紹的程式碼

首先，聊天機器人用**print**描述來自我介紹，並用**input**描述式來問使用者的名字。使用者輸入的名字會儲存在名為**name**的變數，之後用來顯示專屬訊息。

```
1 # 虛擬程式碼
2 顯示聊天機器人自我介紹
3 取得使用者名字
4 顯示打招呼加名字
```

```
1 # 聊天機器人自我介紹
2 print("哈囉。我是Zyxo 64。我是一個聊天機器人。")
3 print("我喜歡動物，也喜歡聊食物。")
4 name = input("你叫什麼名字?: ")
5 print("你好", name, "很高興認識你")
```

～把程式加到檔案中，跑跑看！

## 第3步：展現你的數學設計能力

如果要展示量天機器人的數學能力，可以自行設計各種類行的問題。

例如：問使用者今年的年份，並且猜聊天機器人的年齡。接著回應聊天機器人滿100歲的年份。也可以換成單位轉換器、餐廳帳單計算機等內容，來設計與聊天機器人的對話。

```
1 # 虛擬程式碼
2 從使用者取得今年年份
3 從使用者取得聊天機器人年齡
4 顯示猜測正確
5 將聊天機器人年齡換成整數
6 設年數為100-聊天機器人年齡
7 顯示我再多少年就100歲
8 將今年年份轉換為整數
9 顯示那時候是目前年份+年數
```

```
1 # 取得年份資訊
2 year = input("我記不太清楚日期。今年是幾年？： ")
3 print("好的，我覺得沒錯。謝謝！")
4 # 請使用者猜年齡
5 myage = input("你能猜出我的年齡嗎？ - 輸入一個數字： ")
6 print("沒錯，你猜對了。我", myage)
7 # 計算聊天機器人滿100歲的年份
8 myage = int(myage)
9 nyears = 100 - myage
10 print("我再", nyears, "年就滿100歲了。")
11 print("到時候是", int(year) + nyears)
```

～把程式加到檔案中，跑跑看！

## 第4步：使用儲存的資料產生簡單的填入式回應範本

接著我們可以用使用者輸入的資料，在對話中適合的地方，針對幾個話題問使用者並回應。

以下是討論食物和動物的對話範例。注意使用者輸入的回應是如何儲存在變數並用於print描述式。

```
1 # 食物話題
2 print("我喜歡巧克力，也喜歡嘗試各種新食物。")
3 food = input("你呢。你最喜歡的食物是什麼？： ")
4 print("我也喜歡", food)
5 question = "你多久吃一次" + food + "?: "
6 howoften = input(question)
7 print("真有趣。不知道這樣對健康好不好！")
```

～把程式加到檔案中，跑跑看！

```
1 # 動物話題
2 animal = input("我最喜歡的動物是長頸鹿。你呢？； ")
3 print(animal, "! 我不喜歡。")
4 print("不知道", animal, "喜不喜歡吃", food, "?")
```

～把程式加到檔案中，跑跑看！

## 第5步：加入關於心情的對話

加入一些關於使用者心情的評，以通用的評論回應，這樣聊天機器人就不用根據使用者的輸入來做真正有智慧的回應。

```
1 # 關於心情的對話
2 feeling = input("你今天覺得如何？； ")
3 print("為什麼你現在覺得", feeling, "呢？")
4 reason = input("請告訴我： ")
5 print("我知道了。謝謝分享。")
```

～把程式加到檔案中，跑跑看！

專屬的道別做結束。

用使用者的名字寫成專屬道別來結束聊天機器人的對話。

```
1 # 道別
2 print("今天事情真多！")
3 print("我累到無法繼續聊天了，之後再聊。")
4 print("再見", name, "我喜歡跟你聊天！")
```

～把程式加到檔案中，跑跑看！

## 2. 如何改進這個聊天機器人？

這個專題最大的問題之一是電腦無法選擇要給什麼輸出，也就是無法根據不同的輸入來選擇不同的輸出。如果要突破，聊天機器人必須做決定，這可利用條件分支的主題來達成目的。

另外，為了讓這個聊天機器人更有趣，可以加入一點無法預測性，讓它每次執行都說些不同的話。這可利用清單和隨機模組就能做到。

又，聊天機器人還能加入暫停，讓它看起來像在思考。

最後，這個聊天機器人是屬於你的，用你的創意和**Python**程式碼來讓它有自己的特色和更有人性吧！

～好用的聊天機器人

現在有很多聊天機器人能進行簡單的對話，用於銷售、客服等應用。隨著運算能力的提升，聊天機器人不只能看懂輸入的文字，還能聽懂人類語言。它們能根據大量資料來做出有智慧的回應，而且不只用文字，還能用人類的聲音。