**WeHelp**

**作業 - 第2週**

**完成以下任務，僅可使用 Python 和 JavaScript，且不得使用第三方函式庫。**

**注意：**

* Python 程式碼需運行於 Python 3 或以上版本。

**任務 1**

我們收到來自 5 位朋友的消息（JSON 格式），希望透過台北捷運的綠線（包括小碧潭站）去見其中一位朋友。請撰寫程式碼，根據目前所在車站及車站間的數量，找出最近的朋友並列印其名字。

**限制：**

不得修改既有程式碼。

**Python**：

python

複製程式碼

def find\_and\_print(messages, current\_station):

# 填寫程式碼

messages = {

"Leslie": "I'm at home near Xiaobitan station.",

"Bob": "I'm at Ximen MRT station.",

"Mary": "I have a drink near Jingmei MRT station.",

"Copper": "I just saw a concert at Taipei Arena.",

"Vivian": "I'm at Xindian station waiting for you."

}

find\_and\_print(messages, "Wanlong") # 輸出 Mary

find\_and\_print(messages, "Songshan") # 輸出 Copper

find\_and\_print(messages, "Qizhang") # 輸出 Leslie

find\_and\_print(messages, "Ximen") # 輸出 Bob

find\_and\_print(messages, "Xindian City Hall") # 輸出 Vivian

**JavaScript**：

javascript

複製程式碼

function findAndPrint(messages, currentStation){

// 填寫程式碼

}

const messages = {

"Bob": "I'm at Ximen MRT station.",

"Mary": "I have a drink near Jingmei MRT station.",

"Copper": "I just saw a concert at Taipei Arena.",

"Leslie": "I'm at home near Xiaobitan station.",

"Vivian": "I'm at Xindian station waiting for you."

};

findAndPrint(messages, "Wanlong"); // 輸出 Mary

findAndPrint(messages, "Songshan"); // 輸出 Copper

findAndPrint(messages, "Qizhang"); // 輸出 Leslie

findAndPrint(messages, "Ximen"); // 輸出 Bob

findAndPrint(messages, "Xindian City Hall"); // 輸出 Vivian

**任務 2**

假設有一些顧問提供諮詢服務，請根據時段、服務時長及選擇標準，幫助客戶預訂最合適的顧問。

**條件：**

1. 預約請求需依序進行，順序不可更改。
2. 只有在現有預約時段不重疊時，顧問才可提供服務。
3. 如果標準是「價格」，請選擇價格最低的顧問。
4. 如果標準是「評分」，請選擇評分最高的顧問。
5. 如果所有顧問均無法服務，請輸出「No Service」。

**限制：**

不得修改既有程式碼。

**Python**：

python

複製程式碼

# 填寫程式碼

def book(consultants, hour, duration, criteria):

# 填寫程式碼

consultants = [

{"name": "John", "rate": 4.5, "price": 1000},

{"name": "Bob", "rate": 3, "price": 1200},

{"name": "Jenny", "rate": 3.8, "price": 800}

]

book(consultants, 15, 1, "price") # 輸出 Jenny

book(consultants, 11, 2, "price") # 輸出 Jenny

book(consultants, 10, 2, "price") # 輸出 John

book(consultants, 20, 2, "rate") # 輸出 John

book(consultants, 11, 1, "rate") # 輸出 Bob

book(consultants, 11, 2, "rate") # 輸出 No Service

book(consultants, 14, 3, "price") # 輸出 John

**JavaScript**：

javascript

複製程式碼

// 填寫程式碼

function book(consultants, hour, duration, criteria){

// 填寫程式碼

}

const consultants = [

{"name": "John", "rate": 4.5, "price": 1000},

{"name": "Bob", "rate": 3, "price": 1200},

{"name": "Jenny", "rate": 3.8, "price": 800}

];

book(consultants, 15, 1, "price"); // 輸出 Jenny

book(consultants, 11, 2, "price"); // 輸出 Jenny

book(consultants, 10, 2, "price"); // 輸出 John

book(consultants, 20, 2, "rate"); // 輸出 John

book(consultants, 11, 1, "rate"); // 輸出 Bob

book(consultants, 11, 2, "rate"); // 輸出 No Service

book(consultants, 14, 3, "price"); // 輸出 John

**任務 3**

找出名字中中間字唯一的那個人，並列印該名字。可以假設輸入的每個名字都是中文，且字數為 2 ~ 5 個字。  
若名字有 2 個字，則中間字為第 2 個字；若名字有 4 個字，則中間字為第 3 個字。

**限制：**

不得修改既有程式碼。

**Python**：

python

複製程式碼

def func(\*data):

# 填寫程式碼

func("彭大牆", "陳王明雅", "吳明") # 輸出 彭大牆

func("郭靜雅", "王立強", "郭林靜宜", "郭立恆", "林花花") # 輸出 林花花

func("郭宣雅", "林靜宜", "郭宣恆", "林靜花") # 輸出 沒有

func("郭宣雅", "夏曼藍波安", "郭宣恆") # 輸出 夏曼藍波安

**JavaScript**：

javascript

複製程式碼

function func(...data){

// 填寫程式碼

}

func("彭大牆", "陳王明雅", "吳明"); // 輸出 彭大牆

func("郭靜雅", "王立強", "郭林靜宜", "郭立恆", "林花花"); // 輸出 林花花

func("郭宣雅", "林靜宜", "郭宣恆", "林靜花"); // 輸出 沒有

func("郭宣雅", "夏曼藍波安", "郭宣恆"); // 輸出 夏曼藍波安

**任務 4**

給定一個數列：0, 4, 8, 7, 11, 15, 14, 18, 22, 21, 25, …  
找出此數列中的第 n 項。

**限制：**

不得修改既有程式碼。

**Python**：

python

複製程式碼

def get\_number(index):

# 填寫程式碼

get\_number(1) # 輸出 4

get\_number(5) # 輸出 15

get\_number(10) # 輸出 25

get\_number(30) # 輸出 70

**JavaScript**：

javascript

複製程式碼

function getNumber(index){

// 填寫程式碼

}

getNumber(1); // 輸出 4

getNumber(5); // 輸出 15

getNumber(10); // 輸出 25

getNumber(30); // 輸出 70

**任務 5 (選做)**

給定火車每節車廂的剩餘座位數、狀態位圖及乘客人數，撰寫一個程序找出最合適的車廂服務乘客。如果沒有車廂可以容納這些乘客，請列印 -1。

**條件：**

* 剩餘座位數：包含每節車廂剩餘座位數的列表/陣列。
* 狀態位圖：包含 0 或 1 的列表/陣列，其中 1 表示該車廂目前可服務乘客。
* 乘客人數：需要一次性服務的乘客人數。

**限制：**

不得修改既有程式碼。

**Python**：

python

複製程式碼

def find(spaces, stat, n):

# 填寫程式碼

find([3, 1, 5, 4, 3, 2], [0, 1, 0, 1, 1, 1], 2) # 輸出 5

find([1, 0, 5, 1, 3], [0, 1, 0, 1, 1], 4) # 輸出 -1

find([4, 6, 5, 8], [0, 1, 1, 1], 4) # 輸出 2

**JavaScript**：

javascript

複製程式碼

function find(spaces, stat, n){

// 填寫程式碼

}

find([3, 1, 5, 4, 3, 2], [0, 1, 0, 1, 1, 1], 2); // 輸出 5

find([1, 0, 5, 1, 3], [0, 1, 0, 1, 1], 4); // 輸出 -1

find([4, 6, 5, 8], [0, 1, 1, 1], 4); // 輸出 2

翻譯完成，如有其他需求，請隨時告知！