

# 110-2 DBM Final Project Crowdfunding Platform

---

This is for NCU 110-2 Database Management Final Project - Crowdfunding Platform.

Using MySQL Workbench (8.0.29) and MySQL Server (8.0.29).

Thanks for Teachers and TAs in this course.

## My Info

---

- 姓名: 鄒翔宇
- 學號: 109403019
- 系級: 資管二 A

## Stored Procedure

---

主要部分:

- 目的 **Purpose**
- 輸入 **Input**
- 輸出 **Output**
- 所使用到的 **Tables**
- 實作步驟 **Implementation**
- 執行結果 **DEMO**

### 1. sp\_GetFollowedProposalsByMember [DONE]

- **Purpose:** 透過會員 ID (member\_id) 取得該會員追蹤的提案。
- **Input:** in\_member\_id(int)
- **Output:** number\_of\_rows\_in\_the\_result\_set (int)
- **Tables:** tblfollowingRecord, tblproposal
- **Implementation:**
  - i. SELECT tblfollowingrecord 與 tblproposal JOIN 後的資料，使用 member\_id 作為 ON 的匹配條件，member\_id 為 WHERE 之條件。
  - ii. 將 FOUND\_ROWS() 數值丟進要輸出的 number\_of\_rows\_in\_the\_result\_set (result set 的列數)。
- **Demo:**

```

1 • set @number_of_rows_in_the_result_set = 0;
2 • call db_109403019_test.sp_GetFollowedProposalsByMember(3, @number_of_rows_in_the_result_set);
3 • select @number_of_rows_in_the_result_set;
4

```

member_id	proposal_title	proposal_amount	proposal_goal
3	《 247 Catcher 》   最全面的【抓寶...	2393696	100000

## 2. sp\_RegisterMember [DONE]

- **Purpose:** 註冊使用者，並於成功註冊後列出新增會員的所有欄位。
- **Input:** in\_member\_id (int), in\_hashedPwd (varchar(200)), in\_salt (char(64)), name (varchar(64)), address (varchar(255)), phone (varchar(64))
- **Output:** affected\_row\_num (int)
- **Tables:** tblmember, tblmembercredential
- **Implementation:**
  - 需將 member table 的 member\_id 設為 Auto Increment。
  - DECLARE temp 作為更動到的列數之變數，預設為 0。
  - 使用 EXISTS 語法檢查 in\_email 是否已存在於 tblmember 中。因為 email 必須唯一，因此不可以用重複的 email 註冊。若已經存在則不執行以下動作。
  - 若尚未存在，則使用 INSERT INTO ... VALUES 將資料放入 tblmember 中。並搭配 SELECT ROW\_COUNT() INTO temp 更新更動到的列數。
  - 使用 INSERT INTO ... SELECT 語法將新建使用者 credential 的資料此筆資料加入至 tblmembercredential 中。並搭配 SELECT ROW\_COUNT() + temp INTO temp 更新更動到的列數。
  - 將指定的表格呈現出來，使用 SELECT tblmember 與 tblmembercredential JOIN 後的資料，使用 member\_id 作為 ON 的匹配條件，email = in\_email 為 WHERE 之條件。
  - 最後將 temp 數值丟進要輸出的 affected\_row\_num。
- **Demo:**

```

1 • set @out_affected_row_num = 0;
2 • call db_109403019.sp_RegisterMember('crowdfundingtest@gmail.com', '8289acdb1b78edc0aaeb74e4b0091430fe74f7334f54314bc395f0ad4e95d59',
3 • '#sviR91Ve@', '穆姿緒', '桃園市中壢區中大路300號', '09126543982', @out_affected_row_num);
4 • select @out_affected_row_num;

```

member_id	name	email	password	salt	phone	address
20	穆姿緒	crowdfundingtest@gmail.com	8289acdb1b78edc0aaeb74e4b0091430fe74f7334f54314b...	#sviR91Ve@	09126543982	桃園市中壢區中大路300號

## 3. sp\_UpdatePwd [DONE]

- **Purpose:** 更新會員密碼，必須確認密碼以及 salt 都有更新。
- **Input:** in\_member\_id (int), in\_hashedPwd (varchar(200)), in\_salt (char(64))
- **Output:** affected\_row\_num (int)
- **Tables:** tblmember, tblmembercredential
- **Implementation:**

- i. DECLARE temp 作為更動到的列數之變數，預設為 0。
- ii. 使用 IF NOT EXISTS 檢查輸入 member\_id 之 salt 與 hashed\_pwd\_string 是否與輸入的值皆不同 (因為規定兩者皆需要更新)。這部分要將 tblmember 與 tblmembercredential 以 member\_id 作 JOIN。
- iii. 若確認 salt 與 hashed\_pwd\_string 皆一定會被更新，使用 UPDATE ... SET 語法更新 tblmember 中的 salt 以及 tblmembercredential 中的 hashed\_pwd\_string。並搭配 ROW\_COUNT() 更新 temp (更動到的列數)。
- iv. 使用 ROW\_COUNT 搭配 IF 條件式檢查是否有更新到資料，若大於 0 代表有更新到資料。
- v. 確認有更新到資料的話，用 SELECT 將該用戶更新後的 result 印出來。
- vi. 將 temp 數值丟進要輸出的 affected\_row\_num。

- Demo:

```

1 • set @out_affected_row_num = 0;
2 • call db_109403019.sp_UpdatePwd(2, '07703ba76ae595a212c8317146fb292fb773f34314a3708a6f3e916bbf6e499ce', '7gB%z9Svq8EWe', @out_affected_row_num);
3 • select @out_affected_row_num;
4 •

```

member_id	name	email	password	salt	phone	address
2	林芳岑	adpiscnq.enim.mi@outlook.com	07703ba76ae595a212c8317146fb292fb773f34314a3708a...	7gB%z9Svq8EWe	0988008346	600 嘉義市東區世賢路24號

## 4. sp\_Login [DONE]

- Purpose: 透過 email 以及加密過後的密碼作登入。status\_code: 1 -> 登入成功, 2 -> 密碼錯誤, 3 -> 登入失敗 (會員不存在)
- Input: in\_email (varchar(64)), in\_hashedpwd (varchar(200))
- Output: status\_code (int)
- Tables: tblmember, tblmembercredential
- Implementation:
  - i. 使用 IF EXISTS 檢查 in\_email 是否存在，若不存在 SET status\_code 為 3。若存在執行以下。
  - ii. 檢查使用者登入的 email 是否為已刪除帳號，若是，SET status\_code 為 3。若不是則執行以下。
  - iii. 如果使用者登入的 email 以及 hashed\_pwd 正確則 SET status\_code 為 1。
  - iv. 如果使用者登入的 email 正確但 hashed\_pwd 錯誤則 SET status\_code 為 2。
  - v. 其餘狀況皆 SET status\_code 為 3。

- Demo:

```

1 • set @status_code = 0;
2 • call db_109403019.sp_Login('nullam.velit@yahoo.com',
3 • '1caf7cfa34bdb90fd7a332dfed5a5e0d1a153d6a164234aa98b7ba611b83abe7', @status_code);
4 • select @status_code;

```

@status_code
1

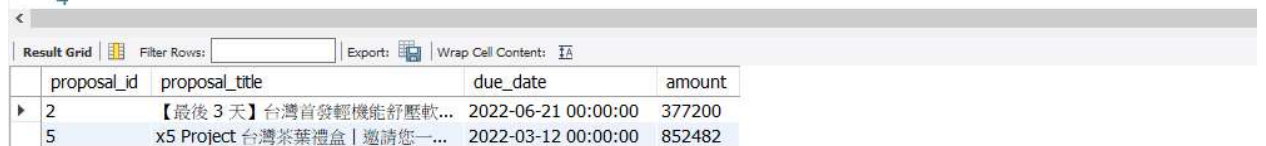
## 5. sp\_GetProposalsByKeyword [DONE]

- **Purpose:** 透過關鍵字進行對提案標題 (proposal\_title) 的模糊搜尋。
- **Input:** in\_keyword (varchar(64))
- **Output:** number\_of\_rows\_in\_the\_result\_set (int)
- **Tables:** tblproposal
- **Implementation:**
  - i. 使用 SELECT 搭配 LIKE 作關鍵字搜尋 (預設不分大小寫)。後方 pattern 使用 CONCAT 串接 ('%', in\_keyword, '%') 。
- **Demo:**

```

1 • set @number_of_rows_in_the_result_set = 0;
2 • call db_109403019_test.sp_GetProposalsByKeyword('台灣', @number_of_rows_in_the_result_set);
3 • select @number_of_rows_in_the_result_set;
4

```



proposal_id	proposal_title	due_date	amount
2	【最後 3 天】台灣首發輕機能舒壓軟...	2022-06-21 00:00:00	377200
5	x5 Project 台灣茶葉禮盒   邀請您一...	2022-03-12 00:00:00	852482

## 6. sp\_UpdateProposalStatus [DONE]

- **Purpose:** 更改提案狀態。有三種狀態，需注意更新順序需遵循 1 -> 2 -> 3。若 status 為 2 且達標率大於 90% 可以延長截止日期 30 天。
- **Input:** proposal\_id (int), status (int)
- **Output:** affected\_row\_num (int)
- **Tables:** tblproposal
- **Implementation:**
  - i. DECLARE temp 作為更動到的列數之變數，預設為 0。
  - ii. 檢查 in\_status 是不是在合理範圍內 (1~3)，若在合理範圍內則執行以下步驟。
  - iii. 使用 UPDATE ... SET 更新 tblproposal 內的資料，以 proposal\_id = in\_proposal\_id 作為 WHERE 的匹配條件。並且原 status 必須比 in\_status 少 1 (更新順序只能遵循 1 → 2 → 3)。
  - iv. 延續上個步驟，將 status 更新為 in\_status。
  - v. 並利用 IF(condition, true\_value, false\_value) 有條件的更新 due\_date，只有在 in\_status = 2 時，必須將當下時間 + 90 天 設為 due\_date。這邊利用 DATE\_ADD(NOW(), INTERVAL value addunit) 回傳相加過後的日期。
  - vi. 檢查達標率 (amount/goal) 是否大於 0.9，若大於一樣使用 DATE\_ADD() 將截止時間加長 30 天。
  - vii. 搭配 ROW\_COUNT() 更新 temp (更動到的列數)。
  - viii. 使用 IF ROWCOUNT() > 0 檢查該指定資料列是否有被更新，若有則用 SELECT 將我們要的資料列印出來。
  - ix. 將 temp 數值丟進要輸出的 affected\_row\_num。
- **Demo:**

```

1 • set @out_affected_row_num = 0;
2 • call db_109403019.sp_UpdateProposalStatus(11, 2, @out_affected_row_num);
3 • select @out_affected_row_num;
4

```

proposal_id	status
11	2

## 7. sp\_GetHistorySponsorByMember [DONE]

- **Purpose:** 透過會員 ID (member\_id) 查詢該會員過去贊助過的提案。
- **Input:** proposal\_id (int), status (int)
- **Output:** number\_of\_rows\_in\_the\_result\_set (int)
- **Tables:** tblsponsorrecord, tblproposaloption, tblproposal
- **Implementation:**
  - 將 tblsponsorrecord 以 proposal\_option\_id 作為 ON 之匹配條件與 proposaloption table 作 JOIN。
  - 再將 tblproposal 以 proposal\_id 作為 ON 之匹配條件與前者作 JOIN。
  - 使用 SELECT 將 in\_member\_id 之歷史贊助紀錄列印出來。
  - 將 FOUND\_ROWS() 數值丟進要輸出的 number\_of\_rows\_in\_the\_result\_set。
- **Demo:**

```

1 • set @number_of_rows_in_the_result_set = 0;
2 • call db_109403019_test.sp_GetHistorySponsorByMember(1, @number_of_rows_in_the_result_set);
3 • select @number_of_rows_in_the_result_set;
4

```

member_id	proposal_id	proposal_title	proposal_option_title	amount	status
1	10	【HOMI】地表最強全情境短袖 & EA...	75mm 超級早鳥價格	18800	3
1	3	數感宇宙探索課程   為國小生打造的...	單組 體驗初學	1790	2

## 8. sp\_sp\_GetUnrepliedComments [DONE]

- **Purpose:** 透過會員 ID (member\_id) 搜尋該會員擁有的所有提案中尚未回覆的留言。
- **Input:** proposal\_id (int), status (int)
- **Output:** number\_of\_rows\_in\_the\_result\_set (int)
- **tables:** tblcomment, tblproposal
- **Implementation:**
  - 將 tblcomment 以 proposal\_id 作為 ON 之匹配條件與 tblproposal 作 JOIN。
  - SELECT 我們所需要的資料，WHERE 之條件為表中 member\_id = in\_member\_id，且 proposer\_response 為空字串 '' 或 IS NULL。
  - 將 FOUND\_ROWS() 數值丟進要輸出的 number\_of\_rows\_in\_the\_result\_set。

- Demo:

```
1 • set @number_of_rows_in_the_result_set = 0;
2 • call db_109403019.sp_GetUnrepliedComments(7, @number_of_rows_in_the_result_set);
3 • select @number_of_rows_in_the_result_set;
4
```

member_id	proposal_id	proposal_title	comment_id	member_comment	comment_time	proposer_response
7	1	《 247 Catcher 》   最全面的【抓寶... 9		有機會可以早點出貨嗎? 6/1要等很久	2020-11-18 22:12:12	

## 9. sp\_GetProposalByCompletionRate [DONE]

- Purpose: 透過輸入 ratio (amount/goal) 列出所有高於達標比例的提案，並由高至低排序。
- Input: in\_ratio (float)
- Output: number\_of\_rows\_in\_the\_result\_set (int)
- tables: tblproposal
- Implementation:
  - 從 tblproposal 中 SELECT 要列印出的資料，ratio 以 amount / goal 作計算。WHERE 之條件為 amount / goal 大於等於 in\_ratio。
  - 使用 ORDER BY amount / goal DESC 作降冪排序列印資料。
  - 將 FOUND\_ROWS() 數值丟進要輸出的 number\_of\_rows\_in\_the\_result\_set。
- Demo:
 

```
1 • set @outNumFound = 0;
2 • call db_109403019.sp_GetProposalByCompletionRate(5, @outNumFound);
3 • select @outNumFound;
```

proposal_id	proposal_title	amount	goal	ratio
1	《 247 Catcher 》   最全面...	2393696	100000	23.9370
5	x5 Project 台灣茶葉禮盒   ...	852482	50000	17.0496
10	【HOMI   地表最強全情境...	1556942	100000	15.5694
3	數感宇宙探索課程   為國小...	3737138	500000	7.4743
9	SIRUI 75mm ?變形鏡頭	532600	100000	5.3260

## 10. sp\_CreateProposal [DONE]

- Purpose: 創建新的提案 (proposal)。
- Input: member\_id (int), title (varchar(120)), content (TEXT), goal (int), category\_id (int)
- Output: affected\_row\_num (int)
- tables: tblproposal, tblproposalmember, tblcategory
- Implementation:
  - 需將 tblproposal 以及 tblproposalmember table 的 id 設為 Auto Increment。
  - DECLARE temp 作為更動到的列數之變數，預設為 0。
  - 判斷 in\_member\_id 是否存在於 tblmember 中，以及 in\_category\_id 是否存在於 tblcategory 中，若存在則執行以下步驟。



- iv. 使用 `INSERT INTO ... VALUES` 插入新的 proposal 到 `tblproposal` 中。並搭配 `ROW_COUNT()` 更新 `temp` (更動到的列數)。
- v. 使用 `INSERT INTO ... VALUES` 插入新的關係資料至 `tblproposalmember` 中。並搭配 `ROW_COUNT()` 更新 `temp` (更動到的列數)。
- vi. 將 `tblproposal` 以 `category_id` 作為 `ON` 之匹配條件與 `tblcategory` 作 `JOIN`。並從中 `SELECT` 我們需要的資料列印出來。
- vii. 將 `temp` 數值丟進要輸出的 `affected_row_num`。

- **Demo:**

```

1 • set @out_affected_row_num = 0;
2 • call db_109403019.sp_CreateProposal(7, '好困枕【讓你好好睡】脖子的溫柔靠山，釋放你的壓力與電力 | 一夜好眠不是夢，
3   '台灣有超過20萬以上的人有睡眠障礙，你是不是也是其中一員？經常睡不好、睡不飽、醒來卻還是滿身疲累，一夜好眠成了一種奢求。...',
4   100000,
5   5,
6   @out_affected_row_num);
7 • select @out_affected_row_num;

```

proposal_id	proposal_title	proposal_content	amount	goal	status	viewed_num	create_date	due_date	category
24	好困枕【讓你好好...	台灣有超過20萬以...	0	100000	1	0	2022-06-11 21:05:49	NULL	空間

## 11. sp\_GetRecommendedProposals [DONE]

- **Purpose:** 輸入會員 ID (`member_id`) 獲取推薦的提案 (`proposal`)。列出點擊率 (`viewed_num`) 最高的前五名。若不符合推薦內容篩選標準則單純按照點擊率列出。
- **Input:** `member_id` (int)
- **Output:** number of rows in the result set (int)
- **Recommended conditions:**
  - i. 和 `in_member_id` 贊助相同提案的人也贊助過的提案 (最複雜的)
  - ii. 為 `in_member_id` 不曾贊助過的提案。 `tblsponsorrecord.member_id != in_member_id`
  - iii. 提案狀態必須是 2。 `status = 2`
  - iv. 推薦內容不可以是自己的提案 `tblproposalmember.member_id != in_member_id`
- **Tables:** `tblsponsorrecord`, `tblproposaloption`, `tblproposal`, `tblproposalmember`
- **Idea:**
  - i. `tblsponsorrecord JOIN tblproposaloption` 可以找出所有贊助者以及贊助之選項。
  - ii. 再 `JOIN tblproposal` 可以找出贊助之選項的提案 ID。
  - iii. `proposalmember` 可以用找出提案者。
  - iv. 因此全部 `JOIN` 在一起，再用 `WHERE` 將條件一一解決。
- **Implementation:**
  - i. 先使用 `IF NOT EXISTS` 檢查 `in_member_id` 是否存在於 `tblmember` 中。若不存在則直接列出點擊率最高的五個提案。若存在則執行以下步驟。
  - ii. 判斷是否符合四個條件，若符合則將其列印出來。若不符合則列出點擊率最高的五個提案。
  - iii. 將 `tblsponsorrecord`, `tblproposaloption`, `tblpropsoal` `JOIN` 在一起，透過 `WHERE` 設立條件式。
  - iv. 該會員不曾贊助過的提案， `tblsponsorrecord.member_id != in_member_id`。

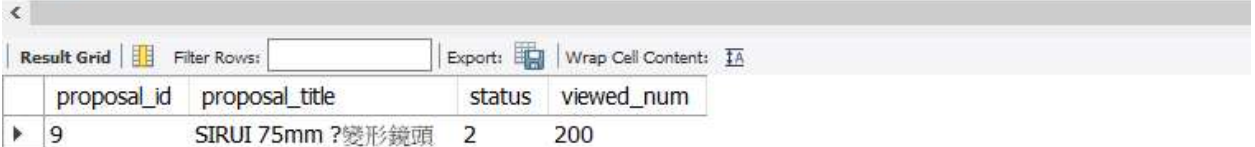
- v. 提案狀態為 2，tblproposal.status = 2。
- vi. 不可以是該會員自己的提案，tblproposalmember.member\_id != in\_member\_id。
- vii. 找到與該會員贊助的提案也贊助過的提案，因為不要重複因此使用 SELECT DISTINCT。
- viii. 先找到該會員贊助過的提案，接著查找贊助相同提案的人。
- ix. 查找他們贊助的提案。設立條件式為他們贊助過的提案且不是該會員贊助的提案。
- x. 使用 ORDER BY viewed\_num DESC 達到以點閱率作降冪排序。
- xi. 使用 LIMIT 限制最多列出五個。
- xii. 將 FOUND\_ROWS() 數值丟進要輸出的 number\_of\_rows\_in\_the\_result\_set。

#### • Demo:

```

1 • set @outNumFound = 0;
2 • call db_109403019_test.sp_GetRecommendedProposals(7, @outNumFound);
3 • select @outNumFound;
4

```



proposal_id	proposal_title	status	viewed_num
9	SIRUI 75mm ?變形鏡頭	2	200

#### • Report:

這題應該算是魔王題，花了幾個小時的時間思考以及除錯，各種表的交叉以及條件設定。本來還在想要用 temp table 來存放 sponsorrecord 與 proposaloption 合併後的紀錄，不過 temp table 一次呼叫只能 select 一次，因此後來就是全部都 JOIN 在一起去處理。這邊放兩張照片來壓壓驚

##### ◦ 多層選取

```

18 -- Recommend under certain conditions
19 -- ppm.member_id 是提案者 id : sprc.member_id 是贊助者 id
20 SELECT pp.proposal_id, pp.title as proposal_title, pp.status, pp.viewed_num
21 FROM
22     tblproposal pp
23 JOIN tblproposaloption ppo ON ppo.proposal_id = pp.proposal_id
24 JOIN tblsponsorrecord spr ON spr.proposal_option_id = ppo.proposal_option_id
25 JOIN tblproposalmember ppm ON ppm.proposal_id = pp.proposal_id
26 WHERE spr.member_id != in_member_id -- b. Input member 不曾贊助過的提案
27     AND pp.status = 2 -- c. 提案狀態需為 2
28     AND ppm.member_id != in_member_id -- d. 不可以是 input member 自己的提案
29     AND ppo.proposal_id IN -- a. input member 不曾贊助以及贊助相同提案的人也贊助過的提案 (不重複)
30     (SELECT DISTINCT ppo.proposal_id -- 其他贊助者也贊助過的相同的提案提案 (不重複)
31      FROM tblproposaloption ppo JOIN tblsponsorrecord spr ON spr.proposal_option_id = ppo.proposal_option_id
32      WHERE spr.member_id IN
33           (SELECT spr.member_id -- 贊助相同提案的人
34            FROM tblproposaloption ppo JOIN tblsponsorrecord spr ON spr.proposal_option_id = ppo.proposal_option_id
35            WHERE
36                 ppo.proposal_id IN
37                 (SELECT ppo.proposal_id -- 贊助過 input member 贊助過的提案
38                  FROM tblproposaloption ppo JOIN tblsponsorrecord spr ON spr.proposal_option_id = ppo.proposal_option_id
39                  WHERE spr.member_id = in_member_id))
40     AND ppo.proposal_id NOT IN -- 且不是 input member 贊助過的提案
41     (SELECT ppo.proposal_id
42      FROM tblproposaloption ppo JOIN tblsponsorrecord spr ON spr.proposal_option_id = ppo.proposal_option_id
43      WHERE spr.member_id = in_member_id))
44 ORDER BY viewed_num DESC
45 LIMIT 5;

```

##### ◦ 醜陋的巨無霸條件式





```
1 • set @affected_row_num = 0;
2 • call db_109403019_test.sp_DeleteMember(4, @affected_row_num);
3 • select @affected_row_num;
4
```

Result Grid						
Filter Rows:		Export:		Wrap Cell Content:		
member_id	name	phone	email	salt	address	
4	USER_DELETED	0958351743	USER_DELETED__@@_4	0	2022-06-13 15:56:55	

## 使用資料

- 修正版 [期末專案情境說明 & Business Logics](#)
- 修正版 [期末專案 Stored Procedures 規格書](#)
- [期末專案 Stored Procedures 問題整理](#)