**資料庫管理-期末專案**

**指導教授 : 柯士文教授**

**助教: 林育全、邱睿揚、林筱芙**

**salt :** [**https://www.grc.com/passwords.htm**](https://www.grc.com/passwords.htm)

**Hashed :****http://www.lorem-ipsum.co.uk/hasher.php?fbclid=IwAR19\_Ytf1P\_Vd49tp95rhrXFum-z7MqFBXuZgwUSKBFmuu12pqfMH8vNj58**

➢ 作業請繳交以下檔案**(檔名請務必都寫對)**

**ER Diagram (PDF 檔)**

(檔名:db\_學號.pdf ,例如:db\_109423043.pdf)

**ER Model 匯出檔 (.mwb 檔)**

(檔名:db\_學號.mwb ,例如:db\_109423043.mwb)

**DataBase 的 schema (.sql 檔)**

(檔名:db\_學號.sql, 例如:db\_109423043.sql)

● 請將自己試想為 DBA(Database Administrator),依照以下需求,設計資料庫中的

**ER-Model,tables 與 Stored Procedures**。

● 以下為資料庫中的 stored procedures 內容與描述,總共會有 12 個 stored procedures(其中

兩個為加分題)。**請依照以下 stored procedures 的名字做相同大小寫的命名**。**(若命名不一 樣,直接不予給分)**

### **●** **加分題為[11] , [12]**

橘色螢光筆底色敘述請確認，有做修正

* 第2,3,4,8,12不用result表，其他output跟result表都要
* output的affected\_row\_num跟number of rows in the result set的意思

→ affected\_row\_num: 資料庫更動到的列數。舉例: 以第二題sp\_RegisterUser為例，

註冊會員後，user table跟user\_credential table都會被更動到，所以output等於2

→ number of rows in the result set: result的列數。舉例: 以第一題為例，跟user\_id=3

距離在20以內的餐廳有2家，所以output等於2

**[1] sp\_GetRestaurant**

1. 透過sp\_GetRestaurant 取得跟會員(user\_id) ,相差20個單位距離以內的店家和顯示其外送費
2. **input :** user\_id (int)
3. **output:** number of rows in the result set (int)

**result:(以user\_id=3為例)**

|  |  |  |
| --- | --- | --- |
| restaurant\_id | restaurant\_name | Delivery\_fee |
| 3 | 伍味鍋 | 30 |
| 8 | 九大雞魯飯 | 30 |

**[2] sp\_RegisterUser**

1. 透過sp\_RegisterUser(會員的帳號 ,salt, 姓名, 地址, 電話, loc\_x, loc\_y,加密後的 password)將Input輸入後於Member進行會員註冊。~~並顯示此次新增的Row number~~。
2. **input:** account(varchar(64)),salt(char(64)),name(varchar(64)),phone(varchar(64)), address(varchar(64)),loc\_x(int),loc\_y(int), hashedPwd(varchar(200))
3. **output:** affected\_row\_num (int) →被影響的列，為了確認store procedure有效的執行

**註:userID-->auto\_increment**

**[3] sp\_ResetNewPassword**

1. 透過sp\_ResetNewPassword(會員的id,salt,雜湊後的密碼)更改會員密碼
2. **input:** user\_id(int), salt(char(64)) ,hashedPwd(varchar(200))
3. **output:** affected\_row\_num (int) →被影響的列，為了確認store procedure有效的執行

**[4] sp\_Login**

1. 透過sp\_Login(會員account、會員hashedpwd)登入會員
2. **input:** ~~user\_id(int),~~account(varchar(64)),hashedpwd(varchar(200))
3. **output:** affected\_row\_num (int) →被影響的列，為了確認store procedure有效的執行

**[5] sp\_SearchMealsOfRestaurant**

1. 透過sp\_SearchMealsOfRestaurant(餐點meals)找出所有含有此餐點的餐廳(沒有限制在20公里以內)
2. **input:** meals(varchar(64))
3. **output:** number of rows in the result set (int)

**result: (以"壽司"為例)**

|  |  |  |
| --- | --- | --- |
| **name** | **meals\_name** | **price** |
| 爭鮮迴轉壽司 | 鮭魚壽司 | 40 |
| 壽司郎 | 鮪魚壽司 | 40 |
| 壽司郎 | 豆皮壽司 | 30 |

**[6] sp\_UpdateOrderStatus**

1. 透過sp\_UpdateOrderStatus(order\_id)更改訂單狀態
2. **input:** order\_id(int),order\_status(int)
3. **output:** affected\_row\_num(int)**→被影響的列，為了確認store procedure有效的執行**

**result: (以order\_id=1 為例)**

|  |  |
| --- | --- |
| **order\_id** | **order\_status** |
| 1 | 3 |

**[7] sp\_HistoryOrder**

1. 透過sp\_HistoryOrder(user\_id)查詢使用者歷史紀錄
2. **input:** user\_id(int)
3. **output:** number of rows in the result set (int)

**result: (以user\_id=2 為例)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **user\_id** | **order\_id** | **deliveryman\_id** | **order\_status** | **total\_price** |
| 2 | 1 | 1 | 2 | 130 |

**[8] sp\_CancelOrder**

1. 透過sp\_CancelOrder(order\_id)取消訂單
2. **input:** order\_id(int)
3. **output:**  affected\_row\_num (int) →被影響的列，為了確認store procedure有效的執行

**[9] sp\_CreateNewOrder**

1. 透過sp\_CreateNewOrder(會員id、外送員id、餐廳id、選取的兩個餐點)建立新訂單
2. **input:** user\_id(int),deliveryman\_id(int),restaurant\_id(int),meals\_1(int),meals\_1\_count(int),meals\_2(int),meals\_2\_count(int)
3. **output:** affected\_row\_num (int) →被影響的列，為了確認store procedure有效的執行

**result: (以user\_id = 6並在restaurant\_id = 3為例)**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **user\_id** | **order\_id** | **restaurant\_id** | **meals\_id** | **unit\_price** | **count** |
| 6 | 8 | 3 | 17 | 120 | 1 |
| 6 | 8 | 3 | 19 | 100 | 2 |

**[10] sp\_SearchStarRestaurant**

1. 透過sp\_SearchStarRestaurant(餐廳id)列出擁有 x 星評價以上(restaurant\_rating)的所有餐廳 (餐廳得到的restaurant\_rating所有評價之平均有x星以上，沒有限制在20公里以內)
2. **input:** x (int)
3. **output:** number of rows in the result set (int)

**result: (以x=4為例)**

|  |  |  |
| --- | --- | --- |
| **restaurant\_id** | **name** | **restaurant\_rating** |
| 2 | 摩斯漢堡 | 5 |
| 8 | 九大雞魯飯 | 4 |
| 10 | 路易莎咖啡 | 4 |

**[11] sp\_CreateNewOrderWithCoupon**

1. 透過sp\_CreateNewOrderWithCoupon(會員id、外送員id、餐廳id、選取的兩個餐點、coupon\_id)並使用coupon建立新訂單
2. **input:** user\_id(int),deliveryman\_id(int),restaurant\_id(int),meals\_1(int),meals\_1\_count(int),meals\_2(int),meals\_2\_count(int),coupon\_id(int)
3. **output:** affected\_row\_num (int) →被影響的列，為了確認store procedure有效的執行

**result:(以user\_id = 5並在restaurant\_id = 10為例)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **order\_id** | **user\_id** | **deliveryman\_id** | **coupon\_id** | **order\_status** | **delivery\_fee** | **total\_price** |
| 6 | 5 | 8 | 15 | 1 | 30 | 295 |

**[12] sp\_InsertNewPassword**

1. 透過sp\_InsertNewPassword**(**會員的id,salt,雜湊後的密碼)並使用insert方式來更改密碼，為了確保被更改的紀錄不會消失
2. **input:** user\_id(int), salt(char(64)) ,hashedPwd(varchar(200))
3. **output:** affected\_row\_num (int) →被影響的列，為了確認store procedure有效的執行