DBS Final project-PostgreSQL

玩家隊伍總傷害、角色統計資料庫

姓名：羅海綺、林雨臻

學號：11127133、11127138

班級：資訊二甲

1. 前言：

學習資料庫一個學期，從資料庫系統概論、relational data model、ER diagram到SQL，我們對於資料庫系統有更深入的了解，也學習如何應用、創建資料庫。期末專題要我們運用學到的內容從ER diagram開始設計，創建出一個資料庫。

1. 內文：  
   一、設計簡介：  
   基於名為原神的遊戲而設計的資料庫，提供玩家查詢資料、計算傷害、回溯數據、提升建議等等功能。

一張含有 文字, 圖表, 圖畫, 寫生 的圖片

自動產生的描述 二、ER diagram：

三、SQL code：

1. TABLE：  
 (a) players:

一張含有 文字, 字型, 螢幕擷取畫面 的圖片

自動產生的描述

一張含有 文字, 字型, 螢幕擷取畫面, 行 的圖片

自動產生的描述一張含有 文字, 字型, 螢幕擷取畫面, 行 的圖片

自動產生的描述(b) elements: (c) weapon\_types:

一張含有 文字, 數字, 字型, 螢幕擷取畫面 的圖片

自動產生的描述一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

自動產生的描述elements data: weapon\_types data:

(d) weapons:

一張含有 文字, 數字, 行, 字型 的圖片

自動產生的描述一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

自動產生的描述(部分資料)

(e) characters:

一張含有 文字, 螢幕擷取畫面, 數字, 字型 的圖片

自動產生的描述一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

自動產生的描述(部分資料)

(f) player\_characters:

一張含有 文字, 字型, 螢幕擷取畫面 的圖片

自動產生的描述

(g) team:

一張含有 文字, 螢幕擷取畫面, 字型 的圖片

自動產生的描述

(h) team\_backup:

一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

自動產生的描述

2. Trigger：

(a) player\_insert\_trigger：

(i) 觸發條件：before insert player

(ii) Trigger function：check\_player\_insert()

(iii)目的：

* 檢查是否有相同UID存在一張含有 文字, 字型, 螢幕擷取畫面, 行 的圖片

  自動產生的描述
* 檢查玩家等級範圍

一張含有 文字, 字型, 螢幕擷取畫面, 行 的圖片

自動產生的描述

(b) player\_update\_trigger：

(i) 觸發條件：before update player

(ii) Trigger function：check\_player\_update()

(iii)目的：

* 確保UID無法被更新 
* 限制更新玩家等級時，更新等級不能小於舊等級 一張含有 文字, 字型, 螢幕擷取畫面, 行 的圖片

  自動產生的描述
* 檢查玩家等級範圍 一張含有 文字, 字型, 螢幕擷取畫面, 行 的圖片

  自動產生的描述

(c) character\_insert\_trigger：

(i) 觸發條件：before insert player\_ characters

(ii) Trigger function：check\_character\_insert()

(iii)目的：

* 一張含有 文字, 字型, 螢幕擷取畫面, 行 的圖片

  自動產生的描述檢查UID是否存在於players表中
* 一張含有 文字, 字型, 螢幕擷取畫面, 行 的圖片

  自動產生的描述檢查角色是否存在 characters 表中
* 檢查該角色玩家是否已擁有
* 一張含有 文字, 字型, 螢幕擷取畫面, 行 的圖片

  自動產生的描述檢查角色等級範圍
* 檢查武器是否存在 weapons 表中 
* 確認武器與角色是否匹配

(d) character\_update\_trigger：

(i) 觸發條件：before update player\_ characters

(ii) Trigger function：check\_character\_update()

(iii)目的：

* 一張含有 文字, 字型, 螢幕擷取畫面, 行 的圖片

  自動產生的描述檢查玩家是否要更改角色名字
* 一張含有 文字, 字型, 螢幕擷取畫面, 行 的圖片

  自動產生的描述檢查角色等級是否下降
* 確認武器與角色是否匹配

一張含有 文字, 字型, 螢幕擷取畫面, 行 的圖片

自動產生的描述

(e) before\_insert\_team：

(i) 觸發條件：before insert team

(ii) Trigger function：check\_team\_members\_exist()

(iii)目的：

* 檢查每個隊伍成員是否存在於玩家的角色列表資料中

一張含有 文字, 字型, 螢幕擷取畫面, 代數 的圖片

自動產生的描述

(f) trigger\_handle\_team\_insertion：

(i) 觸發條件：before insert team

(ii) Trigger function：handle\_team\_insertion()

(iii)目的：

* 確保至少有一個成員存在 一張含有 文字, 字型, 螢幕擷取畫面, 行 的圖片

  自動產生的描述
* 確保隊伍中沒有相同的角色 一張含有 文字, 字型, 螢幕擷取畫面, 行 的圖片

  自動產生的描述
* 一張含有 文字, 收據, 字型, 白色 的圖片

  自動產生的描述判斷是否要將隊伍資料備份，若要則自動備份

3. View：

(a) view\_character\_hp：

依照公式：[ ( 角色等級-1 ) \* 角色屬性的生命值每等級增加數值+角色等級為1時的基礎生命值 ] \* ( 1+裝備武器的生命值加成% )

一張含有 文字, 字型, 螢幕擷取畫面 的圖片

自動產生的描述計算角色生命值。

SQL: 一張含有 文字, 字型, 白色, 印刷術 的圖片

自動產生的描述

得到查訊結果

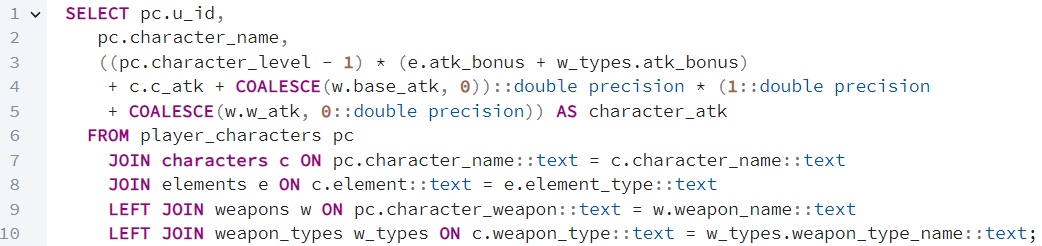
一張含有 文字, 螢幕擷取畫面, 數字, 字型 的圖片

自動產生的描述

(b) view\_character\_atk：

計算角色攻擊力，依照公式：

[ ( 角色等級-1 ) \* ( 角色屬性的攻擊力每等級增加數值+ 角色武器類型的攻擊力每等級增加數值) +角色等級為1時的基礎攻擊力 ] \* ( 1+裝備武器的生命值加成% )



SQL: 一張含有 文字, 字型, 螢幕擷取畫面, 行 的圖片

自動產生的描述

得到查訊結果

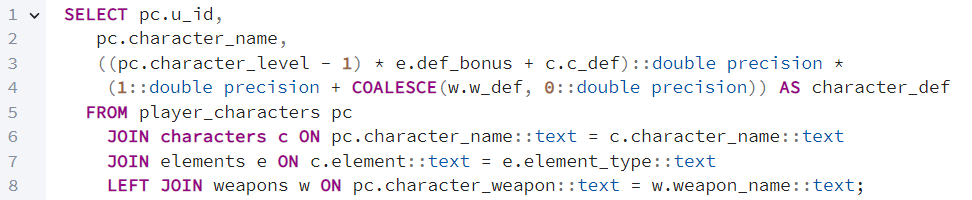
一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

自動產生的描述

(c) view\_character\_def：

計算角色防禦力，依照公式：

[ ( 角色等級-1 ) \* 角色屬性的防禦力每等級增加數值+角色等級為1時的基礎防禦力 ] \* ( 1+裝備武器的防禦力加成% )



SQL: 一張含有 文字, 字型, 螢幕擷取畫面, 行 的圖片

自動產生的描述

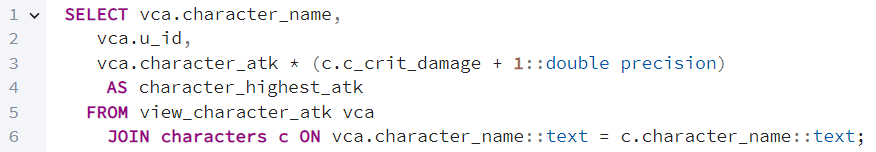
得到查訊結果

一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

自動產生的描述

(d) view\_character\_highest\_atk：

計算角色最高傷害，依照公式：

角色攻擊力\*(1+爆擊傷害%)

SQL: 一張含有 文字, 字型, 螢幕擷取畫面, 白色 的圖片

自動產生的描述

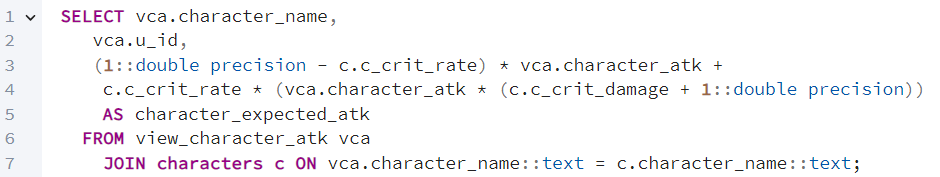
得到查訊結果

一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

自動產生的描述

(e) view\_character\_expected\_atk：

計算角色期望傷害，依照公式：

[ ( 角色等級-1 ) \* ( 角色屬性的攻擊力每等級增加數值+ 角色武器類型的攻擊力每等級增加數值) +角色等級為1時的基礎攻擊力 ] \* ( 1+裝備武器的生命值加成% )

SQL: 一張含有 文字, 字型, 螢幕擷取畫面, 行 的圖片

自動產生的描述

得到查訊結果

一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

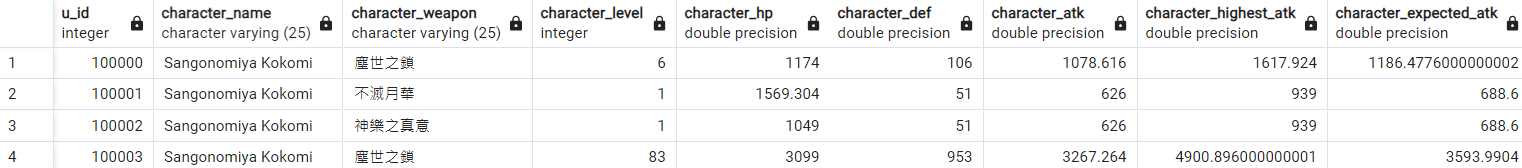
自動產生的描述

(f) view\_show\_player\_character\_data：

顯示完整角色資訊。

SQL：一張含有 文字, 字型, 螢幕擷取畫面, 行 的圖片

自動產生的描述

得到查訊結果

(g) view\_character\_owner\_rate：

計算角色在玩家間的持有率。

SQL: 一張含有 文字, 字型, 螢幕擷取畫面, 白色 的圖片

自動產生的描述

得到查訊結果

一張含有 文字, 數字, 螢幕擷取畫面, 字型 的圖片

自動產生的描述

(h) view\_most\_used\_character\_by\_user：

查看玩家最常使用的角色。

SQL: 一張含有 文字, 字型, 白色, 螢幕擷取畫面 的圖片

自動產生的描述

得到查訊結果

一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

自動產生的描述

(i) view\_team\_damage\_summary：

計算隊伍傷害。

SQL：一張含有 文字, 字型, 螢幕擷取畫面, 行 的圖片

自動產生的描述

得到查訊結果

一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

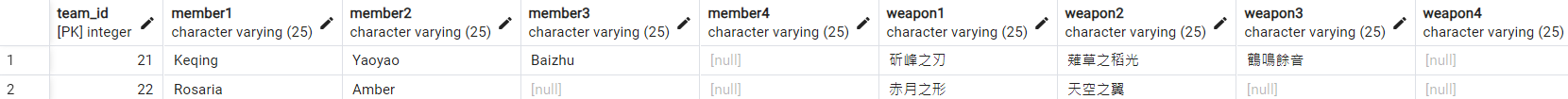
自動產生的描述

4.Function：

(a) create\_team\_from\_backup(backup\_team\_id int, new\_u\_id int)：

範例一，回復以往隊伍資料：

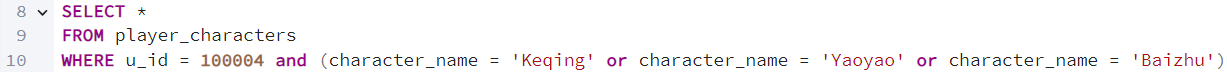
先搜尋目前UID = 100004的玩家目前的備份隊伍有哪些。

一張含有 文字, 字型, 螢幕擷取畫面, 白色 的圖片

自動產生的描述

選擇team\_id = 21，作為希望回復的隊伍。

檢查目前角色裝備設定：



一張含有 文字, 螢幕擷取畫面, 字型, 行 的圖片

自動產生的描述

發現，Yaoyao跟Baizhu的武器跟備份上的數據不一樣。

執行function : create\_team\_from\_backup(backup\_team\_id int, new\_u\_id int)

backup\_team\_id要傳入選擇的team\_id，new\_u\_id要傳入玩家的UID



若武器不同，自動更換武器，且因為重新將備份資料中的隊伍加入目前隊伍中，因此目前隊伍中最早創建的隊伍自動備份。

一張含有 文字, 螢幕擷取畫面, 字型, 數字 的圖片

自動產生的描述

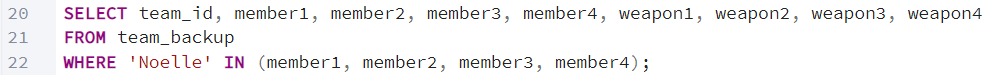
檢查UID = 100004玩家的目前隊伍資料：

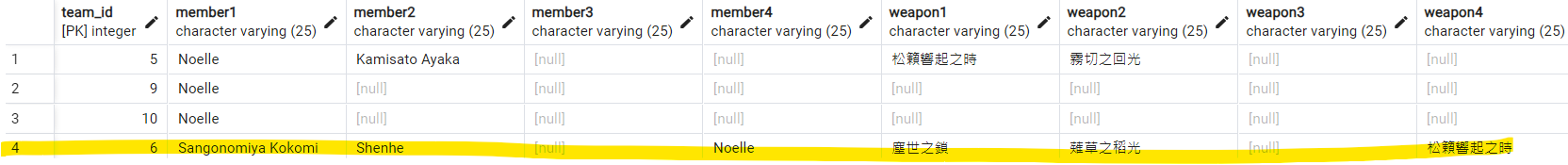
多了一筆team\_from\_backup的資料。  
一張含有 文字, 字型, 白色, 螢幕擷取畫面 的圖片

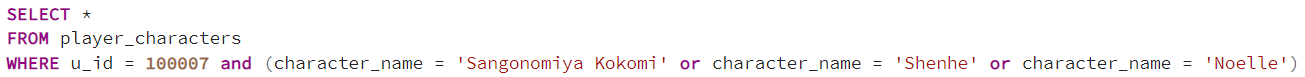
自動產生的描述



範例二，使用他人曾創建的隊伍資料創建自己的隊伍：







一張含有 文字, 螢幕擷取畫面, 字型, 行 的圖片

自動產生的描述 一張含有 文字, 字型, 螢幕擷取畫面, 圖形 的圖片

自動產生的描述

一張含有 文字, 字型, 螢幕擷取畫面, 收據 的圖片

自動產生的描述若希望創建的隊伍中有自己沒有的角色，則在創建過程中自動掠過，並輸出提示”缺少角色”，且會比較強度差距，若低於預期，則輸出建議。

(b) get\_used\_characters\_rank(rank\_n int)：

一張含有 文字, 螢幕擷取畫面, 數字, 字型 的圖片

自動產生的描述一張含有 文字, 螢幕擷取畫面, 數字, 字型 的圖片

自動產生的描述一張含有 文字, 字型, 螢幕擷取畫面, 白色 的圖片

自動產生的描述輸出每個玩家前rank\_n個最常使用的角色，若使用次數一樣則，名字排序的先後順序作為排名依據。