

如何製作最佳 SQL

俗語說的好:工欲善其事，必先利其器。想要寫好一個好的 SQL Statement 使用好的工具是在所難免的，而 Toad 則是大家心目中的首選。

然而 Toad 的功能如此強大，我們又應該使用哪些功能來幫助我們來撰寫 SQL Statement，而撰寫出的 SQL 是一個效率高，維護方便，而且又可以不需要太多複雜技術的方式來完成。

在這裡筆者教各位讀者兩大招式：

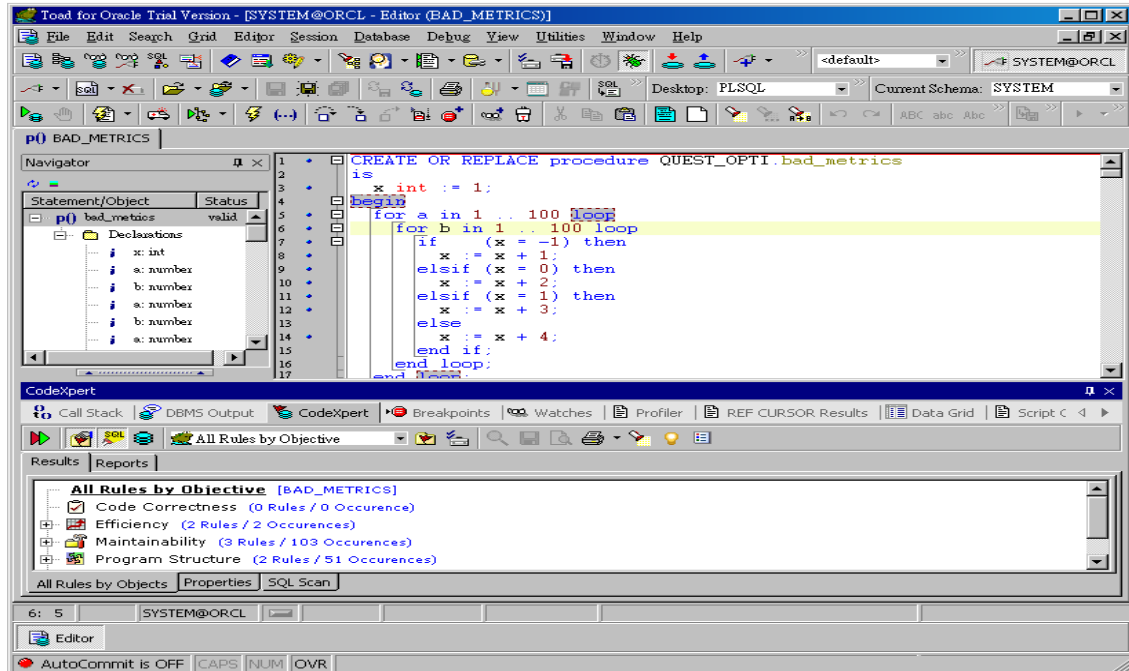
第一招 Code Xpert：

這工具是深藏在 Toad for Professional 之中，也就是大家口中的 Toad 專業版，一個好的 SQL 不只是要求有好的效率而已，而是要做全面性的分析判斷，而 Code Xpert 有著下列這些特性，可以幫助大家製作最佳 SQL。

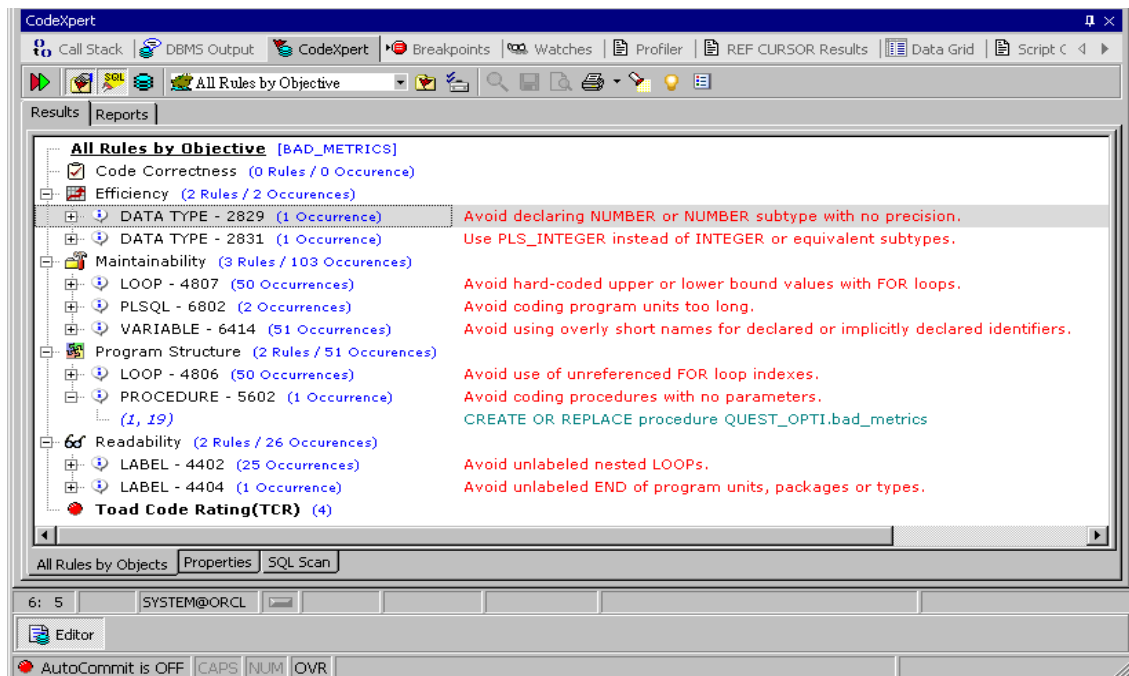
- 程式碼複雜性管理報告
- 檢查程式碼有效性
- 配置規則
- 評估：
 - Efficiency (效能性)
 - Correctness (準確性)
 - Readability (可讀性)
 - Maintainability (維護性)
 - Program structure (結構性)

那現在就開始介紹一下 Code Xpert 的使用方式：

1. 打開 Toad Editor 功能，撰寫一個 SQL，選擇 ，執行 



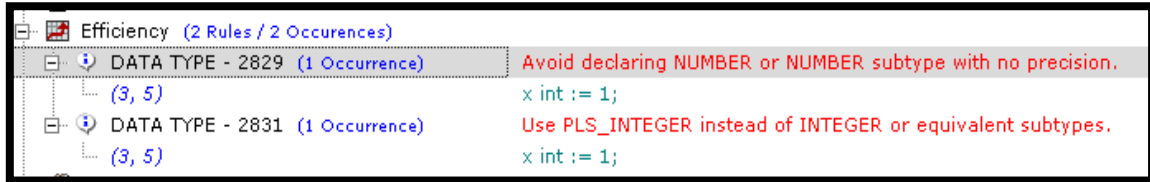
2. 就這麼簡單完成分析，現在讓我們來看看檢查結果



3.好像有不少問題，選幾個我們來看一下到底是什麼問題

- Efficiency (效能性) 的問題分析：

CodeXpert 找出資料型態不對的問題，Oracle 沒有 int 這種資料型態

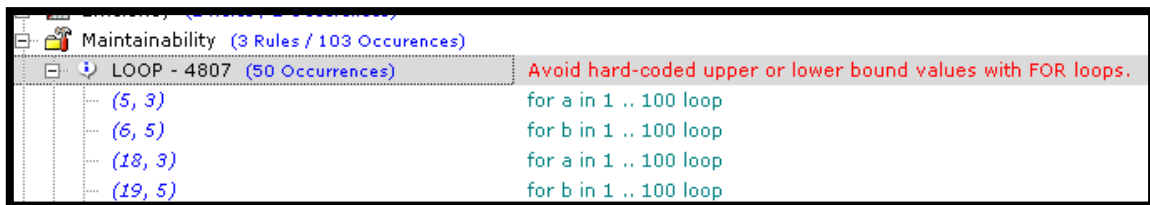


Efficiency (2 Rules / 2 Occurrences)

- DATA TYPE - 2829 (1 Occurrence) Avoid declaring NUMBER or NUMBER subtype with no precision.
 (3, 5) x int := 1;
- DATA TYPE - 2831 (1 Occurrence) Use PLS_INTEGER instead of INTEGER or equivalent subtypes.
 (3, 5) x int := 1;

- Maintainability (維護性) 的問題分析：

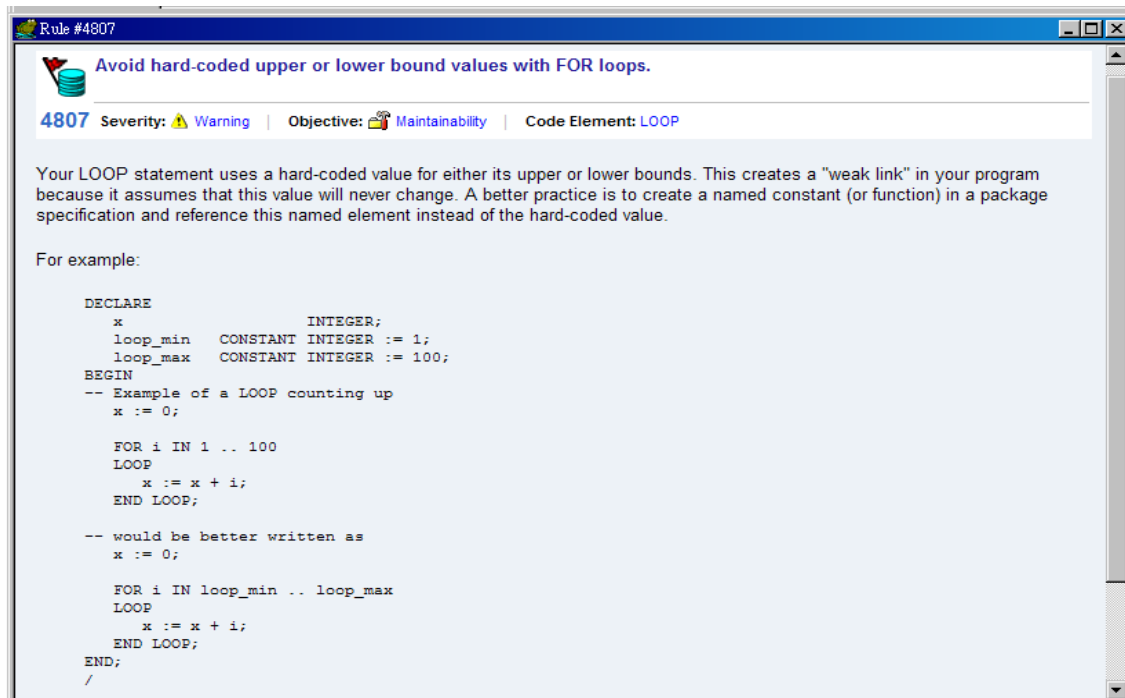
CodeXpert 發現這樣的 SQL 撰寫方式有維護上的問題



Maintainability (3 Rules / 103 Occurrences)

- LOOP - 4807 (50 Occurrences) Avoid hard-coded upper or lower bound values with FOR loops.
 (5, 3) for a in 1 .. 100 loop
 (6, 5) for b in 1 .. 100 loop
 (18, 3) for a in 1 .. 100 loop
 (19, 5) for b in 1 .. 100 loop

CodeXpert 還提供了 SQL 撰寫方式的建議



Rule #4807

Avoid hard-coded upper or lower bound values with FOR loops.

4807 Severity: ⚠ Warning | Objective: 📁 Maintainability | Code Element: LOOP

Your LOOP statement uses a hard-coded value for either its upper or lower bounds. This creates a "weak link" in your program because it assumes that this value will never change. A better practice is to create a named constant (or function) in a package specification and reference this named element instead of the hard-coded value.

For example:

```

DECLARE
    x          INTEGER;
    loop_min  CONSTANT INTEGER := 1;
    loop_max  CONSTANT INTEGER := 100;
BEGIN
    -- Example of a LOOP counting up
    x := 0;

    FOR i IN 1 .. 100
    LOOP
        x := x + i;
    END LOOP;

    -- would be better written as
    x := 0;

    FOR i IN loop_min .. loop_max
    LOOP
        x := x + i;
    END LOOP;
END;
/
    
```

還有報表以及其他的分析建議，等待各位前往體驗 Code Xpert 的魅力。

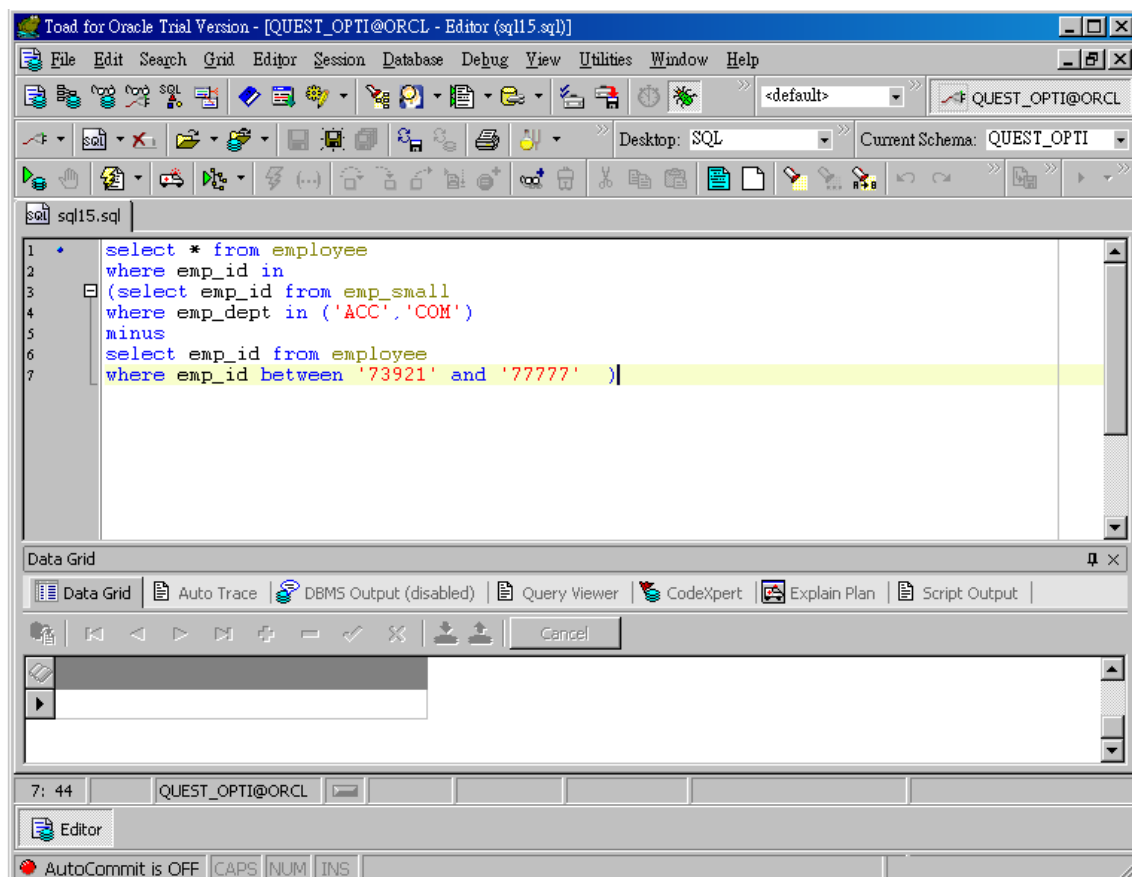
第二招 SQL Optimizer :

SQL Optimizer 是一套 Quest 獨有技術的工具軟體，利用其獨特的技術將使用者所撰寫的 SQL Statement 執行效能最佳化演算，使用電腦來自動撰寫相同功能的 SQL Statement，再將電腦撰寫好的 SQL Statement 予以分析，找出效能最好的 SQL Statement。

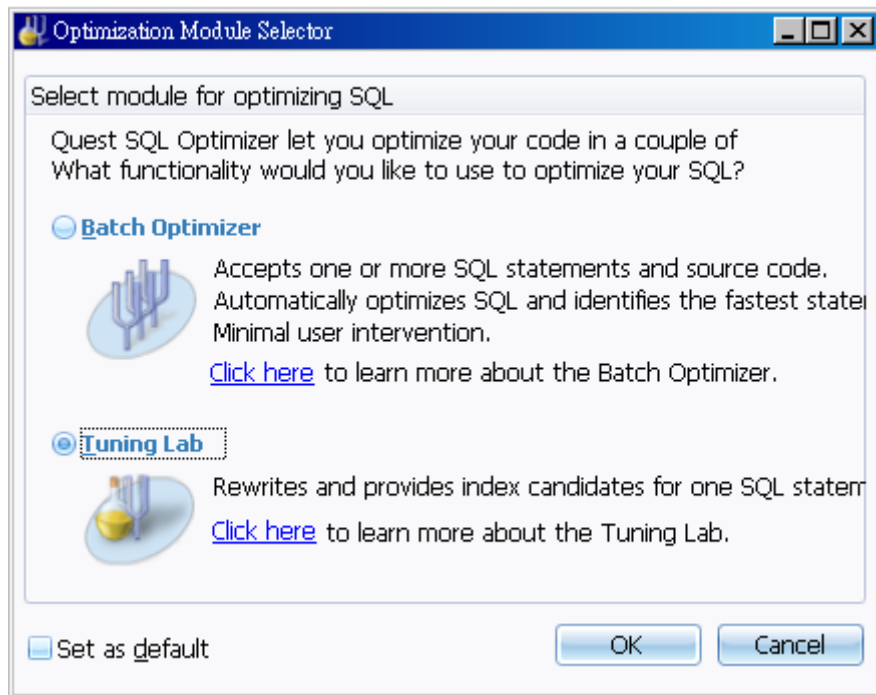
相信許多倍力資訊的客戶，以及 Quest 產品的愛好者對於該項產品並不陌生，只是覺得很奇怪這不是一項獨立的产品嗎？什麼時候變成 Toad 的一部份。其實並不奇怪，在 Toad for Oracle Xpert 版本，也就是 Toad 專家版的時候已經收編旗下，這樣的一個動作使得 Toad 在程式開發上更加的強悍。

現在大家來體驗一下 Toad 與 SQL Optimizer 結合後的強大威力：

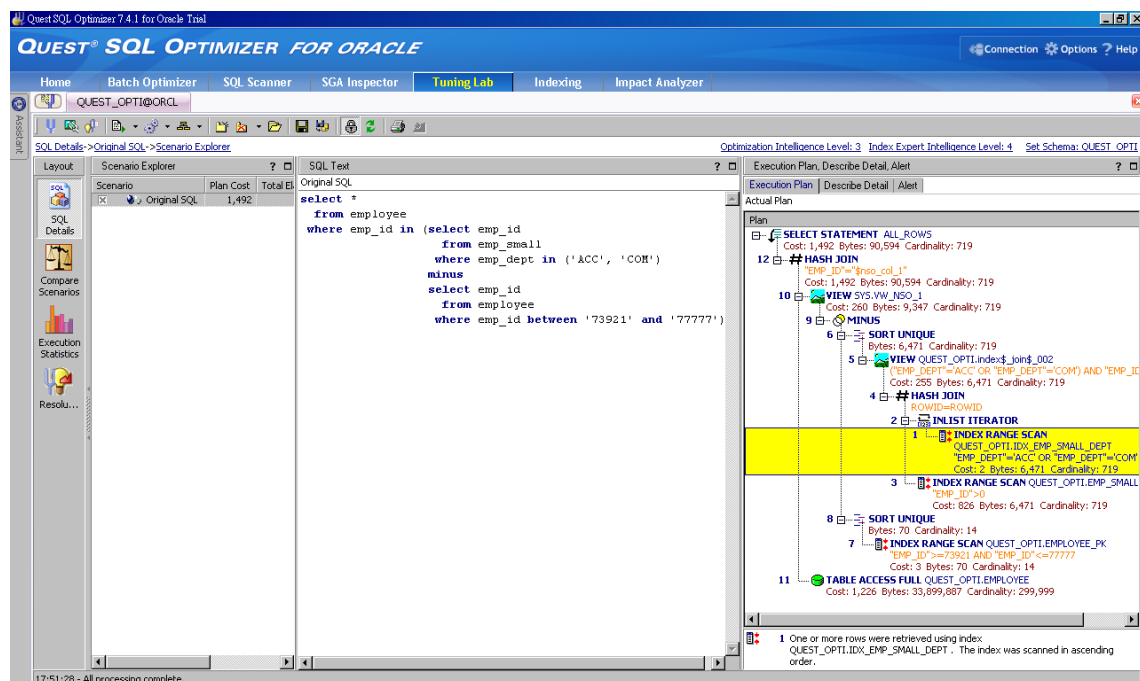
1. 打開 Toad Editor 功能，撰寫一個 SQL，執行



2. 選擇 Tuning Lab，來執行 SQL Tuning



3. 進入 Tuning Lab，請選擇 來執行產生相同功能的 SQL Statement



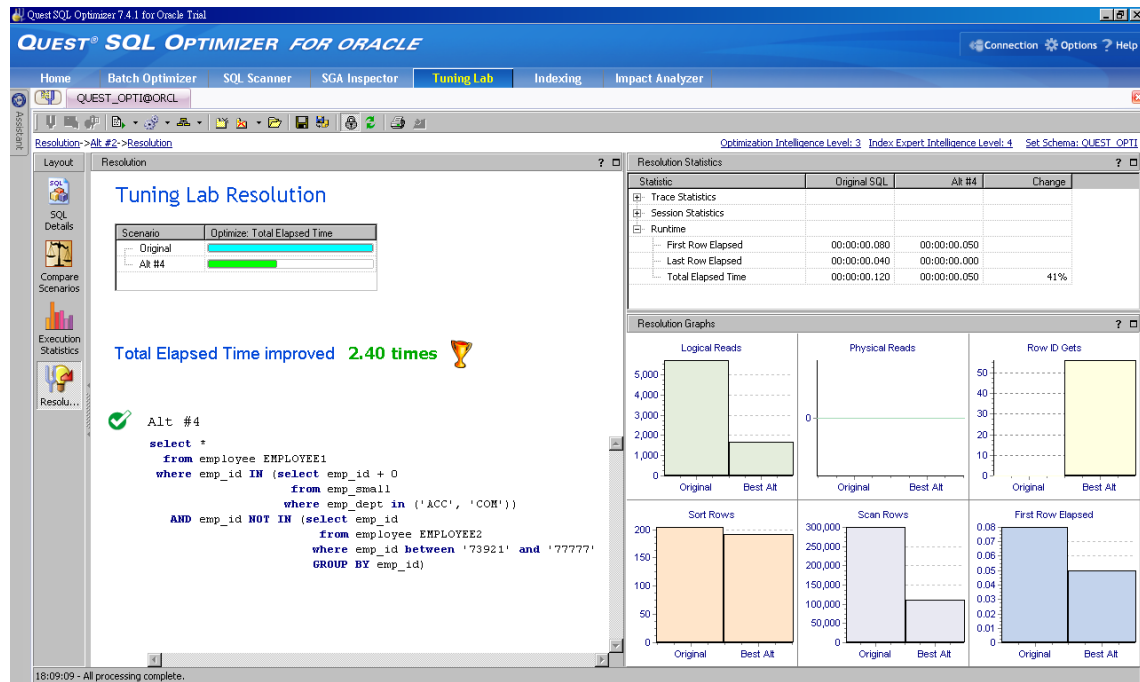
4.此時我們可以看到原來的 SQL 在 Plan Cost 排名第 20 位，
我們可選擇部份 SQL 來執行，確認實際在 DB 上的效能與資源耗用情況

Scenario	Results Comparison	Pl...	Optimize: Total Elapsed Time	Total Elapsed Time	Total CPU	First Row Elapsed	Last Row Elapsed	Physical Reads	Logical Reads	Scan Rows	Scan Gets
Alt #1			288								
Alt #2			288								
Alt #3			302								
Alt #4			302								
Alt #5			984								
Alt #6			985								
Alt #7			985								
Alt #8			985								
Alt #9			986								
Alt #10			986								
Alt #11			1,416								
Alt #12			1,416								
Alt #13			1,490								
Alt #14			1,490								
Alt #15			1,490								
Alt #16			1,490								
Alt #17			1,490								
Alt #18			1,490								
Alt #19			1,490								
Original SQL			1,492								
Alt #20			1,492								
Alt #21			1,492								

5.實際執行後我們可以看到所有 SQL 在 DB 上的效率與資源耗用情況

Scenario	Comparison	Plan Cost	Optimize: Total Elapsed Time	Total Elapsed Time	Scan Rows	Logical Reads	Latch	Parse...	Total CPU	Physical Reads	Sort Rows	First Row El
Alt #5	Identical	984	00:00:00.050	00:00:00.050	20	712	0.00	0.00	0.06	0	0	00:00:00
Alt #7	Identical	985	00:00:00.050	00:00:00.050	26	709	0.00	0.00	0.06	0	0	00:00:00
Alt #9	Identical	986	00:00:00.050	00:00:00.050	32	709	0.00	0.00	0.08	0	0	00:00:00
Alt #10	Identical	986	00:00:00.050	00:00:00.050	35	709	0.00	0.00	0.06	0	0	00:00:00
Alt #2	Identical	288	00:00:00.060	00:00:00.060	11	707	0.00	0.00	0.08	0	192	00:00:00
Alt #6	Identical	985	00:00:00.060	00:00:00.060	23	711	0.00	0.00	0.08	0	0	00:00:00
Alt #8	Identical	985	00:00:00.060	00:00:00.060	29	709	0.00	0.00	0.07	0	0	00:00:00
Alt #14	Identical	1,490	00:00:00.110	00:00:00.110	300101	5757	0.00	0.00	0.12	0	0	00:00:00
Alt #15	Identical	1,490	00:00:00.110	00:00:00.110	300104	5757	0.00	0.00	0.12	0	0	00:00:00
Alt #18	Identical	1,490	00:00:00.110	00:00:00.110	300110	5757	0.00	0.00	0.13	0	0	00:00:00
Alt #20	Identical	1,492	00:00:00.110	00:00:00.110	300116	5757	0.00	0.00	0.12	0	204	00:00:00
Alt #21	Identical	1,492	00:00:00.110	00:00:00.110	300119	5757	0.00	0.00	0.13	0	204	00:00:00
Alt #23	Identical	1,493	00:00:00.110	00:00:00.110	300125	5757	0.00	0.00	0.15	0	204	00:00:00
Alt #24	Identical	1,493	00:00:00.110	00:00:00.110	300128	5757	0.00	0.00	0.12	0	204	00:00:00
Alt #27	Identical	6,132	00:00:00.110	00:00:00.110	300167	5775	0.00	0.00	0.13	0	396	00:00:00
Original SQL	Identical	1,492	00:00:00.120	00:00:00.120	300059	5704	0.00	0.00	0.13	0	204	00:00:00

6.最後會依實際執行的結果，推薦最佳的 SQL，並且分析資源耗用情況供使用者參考



最後，幫助各位讀者無論在資料庫管理或者是在開發介面的應用上更加方便，是 Toad 的目標，也是倍力資訊代理 Quest 產品的目的，但是筆者在這裡還是不免跟各位囉唆一下，善用頭腦才是王道，Toad 提供了絕佳的輔助工具，還是要各位讀者善加利用，方能成為各位的最佳幫手。