一、資料庫試題請以MySQL、MSSQL、Oracle其中一種資料庫語法來回答

1. 有一資料表名稱為Product，其資料如下表，

|  |  |  |  |
| --- | --- | --- | --- |
| 表格名稱 | **Product** | | |
| 欄位名稱 | **Product\_ID** | **Product\_Name** | **Type** |
| 資料內容 | A001 | 蘋果汁 | 果汁 |
| A002 | 沙士 | 汽泡飲料 |
| A003 | 可樂 | 汽泡飲料 |
| A004 | 柳橙汁 | 果汁 |
| A005 | 葡萄汁 | 果汁 |
| A006 | 雪碧 | 汽泡飲料 |

1.1 要找出Type 為「**果汁**」的**資料明細**，SQL語法該如何撰寫？

SELECT Product WHERE Type is “果汁”;

1.2 若要查詢Type為「**汽泡飲料**」的**資料筆數**，該如何撰寫？

1.3 若要查詢Product\_ID 為**A002 ~ A004**的資料明細，SQL語法該如何撰寫？

SELECT Product WHERE Product\_ID FROM A002 TO A004;

1. 有一資料表Employee，其資料結構如下表。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 表格名稱 | **Employee** | | | |
| 欄位名稱 | **Emp\_ID** | **Emp\_Name** | **Emp\_salary** | **Dep** |
| 資料內容 | 0001 | 熊大 | 30,000 | D01 |
| 0002 | 兔兔 | Null | D01 |
| 0003 | 詹姆士 | 25,000 | D02 |
| 0004 | 莎莉 | 40,000 | D01 |
| 0005 | 饅頭人 | 35,000 | D02 |

2.1 請寫出下列SQL語法執行結果  
Select Dep , SUM(Emp\_Salary) From Employee Group By Dep

2.2 要查看**Emp\_Salary 大於32000**的員工有那些，SQL語法該如何撰寫？

SELECT Employee WHERE Emp\_salary > 32000

1. Select name From Book Where Book\_no in (‘01’,’02’)  
   請寫出與上述SQL 語法**執行結果相同的其他SQL語法**。

二、程式設計

請以擅長的程式語言來進行撰寫

1. 設計一個程式，輸入5個大於零的數字，找出最大值及最小值。

// Problem\_1.cpp : 此檔案包含 'main' 函式。程式會於該處開始執行及結束執行。

//

#include <iostream>

#define SIZE 5

void process(int \*p, int len)

{

std::cout << "process:\n";

for (int i = 0; i < len; i++)

{

std::cout << "hello " << p[i] << '\n';

}

}

void bubble\_sort(int array[], int n)

{

for (int i = 0; i < n - 1; i++)

{

for (int j = 0; j < n - i - 1; j++)

{

if (array[j] > array[j + 1])

{

int temp = array[j];

array[j] = array[j + 1];

array[j + 1] = temp;

}

}

}

}

int main()

{

// std::cout << "Hello World!\n";

int size = SIZE;

// Declare an array of integers to be allocated on the stack;

int numbers[size];

// Assign a value to each subsequent element

// (numbers[0] is the first element in the array.)

for (int i = 0; i < size; i++)

{

std::cin >> numbers[i];

}

// process(numbers, size);

/\*

std::cout << "排序前 = ";

for (int i = 0; i < size; i++)

{

std::cout << numbers[i] << ' ';

}

std::cout << '\n';

\*/

bubble\_sort(numbers, size);

std::cout << "最大值: " << numbers[size - 1] << '\n';

std::cout << "最小值: " << numbers[0];

/\*

std::cout << "排序後 = ";

for (int i = 0; i < size; i++)

{

std::cout << numbers[i] << ' ';

}

std::cout << '\n';

\*/

/\*

for (int i = 0; i < size; i++)

{

std::cout << numbers[i];

}

\*/

return 0;

}

// 執行程式: Ctrl + F5 或 [偵錯] > [啟動但不偵錯] 功能表

// 偵錯程式: F5 或 [偵錯] > [啟動偵錯] 功能表

// 開始使用的提示:

// 1. 使用 [方案總管] 視窗，新增/管理檔案

// 2. 使用 [Team Explorer] 視窗，連線到原始檔控制

// 3. 使用 [輸出] 視窗，參閱組建輸出與其他訊息

// 4. 使用 [錯誤清單] 視窗，檢視錯誤

// 5. 前往 [專案] > [新增項目]，建立新的程式碼檔案，或是前往 [專案] > [新增現有項目]，將現有程式碼檔案新增至專案

// 6. 之後要再次開啟此專案時，請前往 [檔案] > [開啟] > [專案]，然後選取 .sln 檔案

2. 設計一個程式，輸入3個數字，由大到小進行排序並列出。

// Problem\_2.cpp : 此檔案包含 'main' 函式。程式會於該處開始執行及結束執行。

//

#include <iostream>

#define SIZE 3

void process(int \*p, int len)

{

std::cout << "process:\n";

for (int i = 0; i < len; i++)

{

std::cout << "hello " << p[i] << '\n';

}

}

void bubble\_sort(int array[], int n)

{

for (int i = 0; i < n - 1; i++)

{

for (int j = 0; j < n - i - 1; j++)

{

if (array[j] < array[j + 1])

{

int temp = array[j];

array[j] = array[j + 1];

array[j + 1] = temp;

}

}

}

}

int main()

{

// std::cout << "Hello World!\n";

int size = SIZE;

// Declare an array of integers to be allocated on the stack;

int numbers[size];

// Assign a value to each subsequent element

// (numbers[0] is the first element in the array.)

for (int i = 0; i < size; i++)

{

std::cin >> numbers[i];

}

// process(numbers, size);

std::cout << "排序前 = ";

for (int i = 0; i < size; i++)

{

std::cout << numbers[i] << ' ';

}

std::cout << '\n';

bubble\_sort(numbers, size);

std::cout << "排序後 = ";

for (int i = 0; i < size; i++)

{

std::cout << numbers[i] << ' ';

}

std::cout << '\n';

/\*

for (int i = 0; i < size; i++)

{

std::cout << numbers[i];

}

\*/

return 0;

}

// 執行程式: Ctrl + F5 或 [偵錯] > [啟動但不偵錯] 功能表

// 偵錯程式: F5 或 [偵錯] > [啟動偵錯] 功能表

// 開始使用的提示:

// 1. 使用 [方案總管] 視窗，新增/管理檔案

// 2. 使用 [Team Explorer] 視窗，連線到原始檔控制

// 3. 使用 [輸出] 視窗，參閱組建輸出與其他訊息

// 4. 使用 [錯誤清單] 視窗，檢視錯誤

// 5. 前往 [專案] > [新增項目]，建立新的程式碼檔案，或是前往 [專案] > [新增現有項目]，將現有程式碼檔案新增至專案

// 6. 之後要再次開啟此專案時，請前往 [檔案] > [開啟] > [專案]，然後選取 .sln 檔案