

## header 筆記

header 也就放宣告的 function 或 class,

假設這個 header 給它命名為 test.h, 裡頭寫

```
1. #ifndef test_h_
2. #define test_h_
3. int compare_max(int a, int b);
4. class Car {
5.     privat:
6.         string name;
7.         int remainder;
8.     public:
9.         Car() { name = "豐田", remainder = 10; }
10.        void setter(string name, int remainder);
11. }
12. #endif
```

現在就來實現 header 裡的 compare\_max() 和 class Car 裡的 setter(); 這裡假設要去實現 header 的檔案名叫它 test.cpp,

其 test.cpp 裡的 code 為:

(記得要 #include "test.h")

1. `#include "test.h"`

2. `int compare_max(int a, int b) {`

3. `// condition ? a : b`, 如果 true 則 a, false 則 b

4. `return (a > b) ? a : b;`

5. `}`

// 實現 class 裡 setter()

6. `void Car::setter(string name, int remainder) {`

7. `this->name = name;` // this 為指標, 指向自己的 class

8. `this->remainder = remainder;` // 等號後面的值為

9. `}` // 外面傳進來的。

---

這樣子, 要使用 header 裡的 function or class, 在主檔上添加 `#include "test.h"` 即可。

如果要手動 compile, 需把 "test.h" 和 "test.cpp" 一起寫到 g++, 假設主檔叫 main.cpp

ex. 手動 compile:

`g++ -o output.exe main.cpp test.h test.cpp`