

程式設計（112-1）作業八一書面報告

B12705055 張佑丞

November 2023

1 檢視 Item 和 Product 類別

由兩類別的宣告如下,

```
class Item{
private:
    char* name;
    int materialCost;
public:
    Item(char* n, int mc);
    ~Item();
    int getMaterialCost();
    bool isThisItem(char* n);
};

class Product{
private:
    char* name;
    int price;
    int cost;
    int salesQty;
    int itemCnt;
    Item** itemList;
public:
    Product(char* n, int p, int c, int sq, int ic);
    ~Product();
    bool isInFrontOf(const Product& prod, int criterion);
    int profit() const; // 1.
    void addItem(Item* itemPtr);
    bool isThisProduct(char* n);
    int getCost() const; // 2.
};
```

可以發現只有 `Product` 的兩函式 `profit()` 以及 `getCost()` 需要加上 `const` 關鍵字, 以下將分別敘述為何需要加上 `const`

1. `profit()`: 可以看到 `Product::isInFrontOf()` 函式中我們傳入了一個屬性為 `const` 的參數 `prod`, 所以我們在對 `prod` 做的任何函數都變成了 `const member function`, 所以因為 `profit()` 會被該參數 call 到, 所以我們需要將其指定為 `const member function`, 也就是說在宣告該函式後須加上 `const` 的關鍵字
2. `getCost()`: 原因同 `profit()`, 因為 `getCost()` 會在 `isInFrontOf()` 中被帶有 `const` 屬性的 `prod` 參數呼叫到, 所以依然為一個 `const member function`, 所以也需要在該函式的宣告後面加上 `const` 的關鍵字