Coding 習慣

好的寫 code 習慣能讓程式在開發 / 維護 / 觀看上更方便也更快速 , 不一定要去模仿寫 code 大神的寫法,可以根據開發團隊的討論來決定 。 但下列幾點是建議各位要遵守的

1. 命名方式:

命名要清楚,讓人一眼就知道這個是什麼(例如:使用者名稱 username),不要自創縮寫(如果真的需要就打上註解),要讓後面維護或一起開發的人有辦法一眼就看出這個是什麼。

2. 檔案放置:

專案裡面會有很多檔案, 我們叫它 layout 。

舉例來說 RESTful API 的檔案就會固定放在 RESTful API 的資料夾中, 如果每個人都隨便亂放那後面要修改內容的時候就會找不到或者少改某些地方的 code。

放檔案位置這部分可以團隊們統一就好,畢竟後面開發也是團隊自己內部使用。

3. 排版:

排版對於除錯跟開發都有很大的幫助。

```
int main() {
    printf("Hello World");
    return 0;
}
```

你們覺得上面哪邊的程式比較容易閱讀呢?

而且這只有一段 Function 當你程式碼越開發越多,就會越來越可怕跟難 以閱讀。

4. 註解:

註解對後期維護與新開發者接手都有很大的幫助,我們會將這段程式出處/哪邊可能目前還有問題未處理/參考哪邊的資料/這段程式是哪天是誰開發寫進註解讓大家有跡可循。

但是請注意!不要寫一些只有你理解的註解,請讓大家一看就明白你在說什麼,否則就喪失了寫註解的用意了。

5. 盡量避免重複程式碼:

當你發現你的 code 中有越來越多一模一樣或者 8~9 成像的程式碼出現 , 你的 code 就已經是個不健康的 code 了。

當你發現這個方法很多地方都會使用到,而你每次都是回上一頁複製,請你把那個方法抽取出來變成共用方法,而不是每次都複製貼上等到將來需要修改時就會有很高的機率改 A 忘記改 B 而導致系統出錯。

分享結束