

錄影帶出租系統

For Mob Programming

有一天九九走在路上看到一家白仕達錄影帶出租店的店長很帥，於是天雷勾動慾火，決定進去應徵工讀。實習第一天，店長 Jim 手把手教九九如何使用**錄影帶出租系統**。當然，白仕達租賃系統是一個 GUI 介面的程式。所有的動作都是在滑鼠與鍵盤的操作底下可以完成。

店長 Jim: 給你的第一個任務，剛剛倫倫來租了舊片 - "緋色的不在場證明"，他要租 3 天，你把資料輸入進去系統裡。

九九：歐虧 pui pui!

九九輸入完畢之後，系統顯示倫倫獲得常客積點 1 點，租片費用為 3.5 美元。所謂的常客積點是用來鼓勵常客多多來租片，累積過多了，未來總公司就會推行一些活動，讓常客可以使用這些點數來進行兌換。

隔天倫倫又回來租了新片 - "金剛大戰哥斯拉"，總共租了兩天。九九將這筆記錄輸入系統後，系統顯示倫倫獲得常客積點 2 點，租片費用是 6 美元

後來倫倫又回來租了兒童片 - "天線寶寶闖台北"，總共租了 7 天。九九將這筆記錄輸入系統後，系統顯示倫倫獲得常客積點 1 點，租片費用是 7.5 美元

打工的最後一天，倫倫來店裡想查詢租片記錄：

店長 Jim: 丸！把"倫倫"的出租紀錄報表印出來，倫倫想要參考一下。

九九：歐虧！Pui!

於是九九對系統輸入了"倫倫"，系統就跑出了倫倫的借閱紀錄、每一片花費的價錢，還有花費的總額，以及常客積點的點數總額。

經過了一番訓練，看到九九學得這麼快，店長便心滿意足地從他的一大包錢包掏出工讀獎金，當作九九的獎勵，九九看到那一大包興奮不已，他期待店長的那一大包能因為自己的努力而越來越大包。

九九在結束了這一場暑假打工之後，因為剛學會了物件導向程式語言 JAVA，所以九九決定來實作與重現白仕達的系統當作他的練習。但是他不想去實作 GUI 的部分，而是想單純的把基本 class 結構與 method 實作完成。所以，租片事實的建立在白仕達系統中原本是以 GUI 來建立，但是九九只想在 main() 以程式實作的方式完成就好了。

本租片系統僅負責將電影租給顧客，不負責顧客歸還的情況

Use Case 的標題敘述僅供參考，但必須實作出裡面 What to do 所提到的事情

Use Case A: 建立一個 Movie (name, type) 的 class ,並在 main()裡宣告 4 個 movies

Precondition:

該系統提供三種類型的電影讓店員新增，分別為新片、舊片及兒童片。

Definition:

店員透過輸入電影名稱及電影類型來新增電影。

What to do:

定義一個 Movie class。

分類為：新片(new_release), 舊片(regular), 兒童片(children)

建立四部電影，分別為

新片 - 我的名字

舊片 - K-0

舊片 - 涼宮春日的

兒童片 - GIVEN

Use Case B: 建立一個 Customer (name) 的 class ,並在 main()裡宣告 4 個 customers

Definition :

店員透過輸入顧客名稱來新增顧客。

What to do:

定義一個 Customer class。

建立四位顧客，分別為倫倫、六花醬、K 昂、史家瑩。

Use Case C: 建立一個 Rental (movie, daysRented) 的 class，並在 main() 裡宣告 7 個 rentals

Definition:

每借一部片就建立一筆租借資料，包含電影名稱和租借天數。

What to do:

定義一個 Rental class。

建立七筆租借資料，分別是

倫倫: GIVEN/8 天

倫倫: 我的名字/7 天

六花醬: 我的名字/8 天

六花醬: K-0/7 天

六花醬: 涼宮春日的/8 天

K 昂: K-0/7 天

史家瑩: GIVEN/8 天

註: 不用考慮電影被租完的問題

Use case D: 在 customer class 裡建立一個 method 叫做 statement()

Definition:

statement()的功能是要印出 customer 的所有租借紀錄。

What to do:

在 main()裡面呼叫 statement()，把四個顧客的租借紀錄印出來。

Use Case D.1: 在 statement() 中，計算每次租借的金額

Definition:

不同類型的電影會有不同的計價方式：

1. 新片： \$3/天
2. 舊片： \$2 可以租兩天，每多一天加 \$1.5
3. 兒童片： \$1.5 可以租三天，每多一天加 \$1.5

店員可以透過 statement() 來得知該顧客每筆租借紀錄的費用。

What to do:

店員呼叫顧客的 statement()。系統輸出該顧客每次租借的費用，以及所有租片費用的總額。

註：租片當下即決定租期天數並結帳，不會有延長租期的狀況發生。

Use Case D.2: 在 statement() 中，計算每次租借時顧客得到的積點

Definition:

為了鼓勵客戶消費，該系統可以提供常客優惠的措施，不同類型的電影會有不同的積點優惠：

1. 新片每租一片可獲得 2.0 常客積點
2. 舊片每租一片可獲得常客積點 1.0 常客積點
3. 兒童片每租一片可獲得常客積點 0.5 常客積點

店員可以透過 statement() 來得知該顧客每筆租借紀錄所獲的積點。

What to do:

店員呼叫顧客的 statement()。系統輸出該顧客每次租借得到的積點，以及顧客目前所持的積分總額。

Use Case E: 幾天之後，新片- 我的名字，在上檔了 2 個月之後，變成舊片。

What to do:

請在 main() 完成相關的程式來進行新片轉舊片的相關事宜。

註：

由於租借紀錄中沒有租借日期的 variable，

所以同學們並不需要將日期實作出來。

租片金額和點數一律根據租片當下該片的種類來做計算。

綜上所述，usecase E 的實作應該會像下列這樣：

```
main () {  
    /*其他 usecase*/  
    .  
    .  
    .  
    //假裝過了兩個月之後  
    /*usecase E 實作*/  
}
```

評分標準：

1. 實作三種類型的電影並建立 Use Case A 中的四部電影 (10%)
2. 實作 Use Case B 中提到的四位顧客 (10%)
3. 實作 Use Case C 中提到的七筆租借記錄 (10%)
4. 輸出指定顧客各筆租借記錄的費用和所獲積點 (15%)
5. 輸出指定顧客所有租借記錄的費用和積點總額 (5%)
6. 程式碼物件導向的程度 (20%)
7. 個人分數 (30%)

Use Case E 為此租片系統日後擴充的提示，因此沒有獨立的評分項目，該 Use Case 處理的好壞會影響到的評分項目為第 6 項。

本題目一定有需要使用多型的地方

請同學分析架構時仔細想想