|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Code Review紀錄內容 | | | | 更新日期 : 2024/08/09 |
| 日期 | CR人員 | 題號 | 備註 | |
| 2024/08/09 | 林亞青 | 第1題 | 建立九九乘法表 | |
| 1. 修改條件式 | * 條件式中要判斷隨機數字是否是兩位數，因為範圍小，不需要另建一個方法判斷。 * 直接判斷隨機數字是否大於9即可。   修改前      修改後 | | | |
| 1. 使用StringBuilder | * Printf相加字串的過程會建立新的記憶體空間 * 3個字串相加要用StringBuilder，提升效能   修改前    修改後 | | | |
| 2024/08/09 | 林亞青 | 第2題 | 隨機生成六個樂透數字並排序 | |
| 1. 修改判斷式 | |  | | --- | | * 判斷式如果是比大小，可以減少=的使用，閱讀時會比較直觀   修改前：    修改後： | | | | |
| 1. 使用Hashset改寫 | * Hashset中相同的元素只會有一筆，改用hashset就不必另寫判斷式。   修改前：    修改後： | | | |
| 2024/08/09 | 林亞青 | 第3題 | 建立一個計算員工薪水的薪資系統 | |
| 1. 用StringBuilder，並加入salary欄位 | * 3個字串相加時使用StringBuilder。 * Sales和supervisor都有使用到getSalary( )，可以直接將salary欄位都寫在employee的類別中就好。   修改前：    修改後： | | | |
| 1. 修改setBouns( ) | * setBouns( )已經是賦值-> setter( ) * 不需要再使用 \* = 重複賦值。   修改前：    修改後： | | | |
| 2024/08/09 | 林亞青 | 第4題 | 使用BufferedWriter輸出員工資料為csv | |
| 1. try-with-resources Statement | * 使用 try-with-resources Statement，確保每個資源結束後自動關閉   修改前：    修改後： | | | |
| 1. 調整命名方式 | * eachList調整成employee比較好直接閱讀。   修改前：    修改後： | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Code Review紀錄內容 | | | | 更新日期 : 2024/08/14 |
| 日期 | CR人員 | 題號 | 備註 | |
| 2024/08/13 | 林亞青 | 第五題 | 列印出今年 m 月的月曆 | |
| 1. 使用try…with…resource | * 可以使用try…with的方式關閉資源 * 使用後就可以不用寫monthScanner.close( )   修改前：    修改後： | | | |
| 1. StringBuilder命名 | * Str通常指字串，命名stringbuilder時，通常會用sb   修改前：    修改後： | | | |
| 2024/08/13 | 林亞青 | 第6-1題 | 讀取csv，按price排序後再輸出新csv | |
| 1. 使用相對路徑 | * 避免其他開發人員讀取不到檔案，可使用相對路徑代替絕對路徑  |  | | --- | | 修改前：    修改後： | | | | |
| 1. Try…catch | * Try…catch的範圍太大，應該要縮小使用   修改前：    修改後： | | | |
| 1. 使用Big Decimal | * 跟數字相關的欄位，在排序處理時用Dig Decimal   修改前：    修改後： | | | |
| 2024/08/13 | 林亞青 | 第6-2題 | 針對 Manufacturer將資料分組，計算 MinPrice ,Price的小計、合計 | |
| 1. append修改 | * String.format可以傳入多個參數，不需要重複使用append   修改前：          修改後： | | | |
| 2024/08/13 | 林亞青 | 第7題 | 資料庫 JDBC | |
| 1. SQL指令改寫 | * 將所需的欄位完整列出（不使用），後續資料庫新增其他欄位時，避免取得其他不需要的欄位資料   修改前：    修改後： | | | |
| 1. 常數：連線資料  * Account * Password | * 帳戶名稱和密碼設定成常數避免打字錯誤及後續修改   修改前：    修改後： | | | |
| 1. 常數：主要欄位 | * 將使用頻率高的四個欄位定義成常數，避免打字錯誤及後續修改   修改前：      修改後： | | | |
| 1. doUpdate（）修正 | * 原本的方法中沒有使用connection.commit和rollback的指令，所以導致方法錯誤 * 在原有的catch區塊中再加入rollback指令 * 透過pstmt.executeUpdate(　)的回傳值，判斷是否更新成功   修改前：    修改後： | | | |