**FEP專案翻寫錯誤案例**

2021/08/10

Copyright the SYSCOM Computer Engineering Co.2021, All Rights Reserved

Revision History

|  |  |  |
| --- | --- | --- |
| Date | Author | Remark |
| 2021/08/10 | YU, Richard | Initial Version |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Catalog

[Catalog 3](#_Toc106876825)

[Concept 5](#_Toc106876826)

[.NET中資料庫資料查詢方法 6](#_Toc106876827)

[案例1(FISC.vb) 6](#_Toc106876828)

[案例2(FISC.vb) 8](#_Toc106876829)

[案例3(FISC.vb) 10](#_Toc106876830)

[不要忽略Warning 12](#_Toc106876831)

[MS SQL Server和Oracle，rtrim(char)函數差異 14](#_Toc106876832)

[查詢資料只返回一筆 20](#_Toc106876833)

[FEPTXN表的操作 22](#_Toc106876834)

[日期format檢核 24](#_Toc106876835)

[case忘記break 25](#_Toc106876836)

[屬性(或者方法)是在子類中還是父類中 26](#_Toc106876837)

[大家來找茬 28](#_Toc106876838)

[字串直接用等號 29](#_Toc106876839)

[還是要看一下工具轉後的代碼 30](#_Toc106876840)

[枚舉使用錯誤 31](#_Toc106876841)

[StringUtils.deleteWhitespace使用錯誤 33](#_Toc106876842)

[Mapper中方法重複定義 34](#_Toc106876843)

[StringUtils.join使用錯誤 35](#_Toc106876844)

[沒有使用事務 36](#_Toc106876845)

[substring第二個參數錯誤 38](#_Toc106876846)

[使用完Ref後要記得把結果塞回去 39](#_Toc106876847)

[Mybatis Ext Mapper中不要改變model中返回欄位的類型 40](#_Toc106876848)

[Mybatis XML中傳入Order By的寫法 40](#_Toc106876849)

[Mybatis 查詢方法是否應該返回List<> 41](#_Toc106876850)

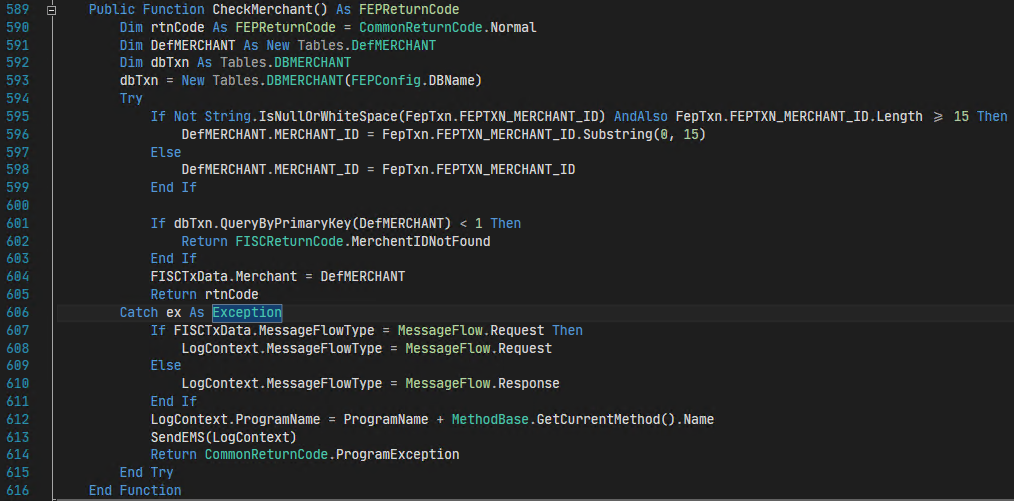
# Concept

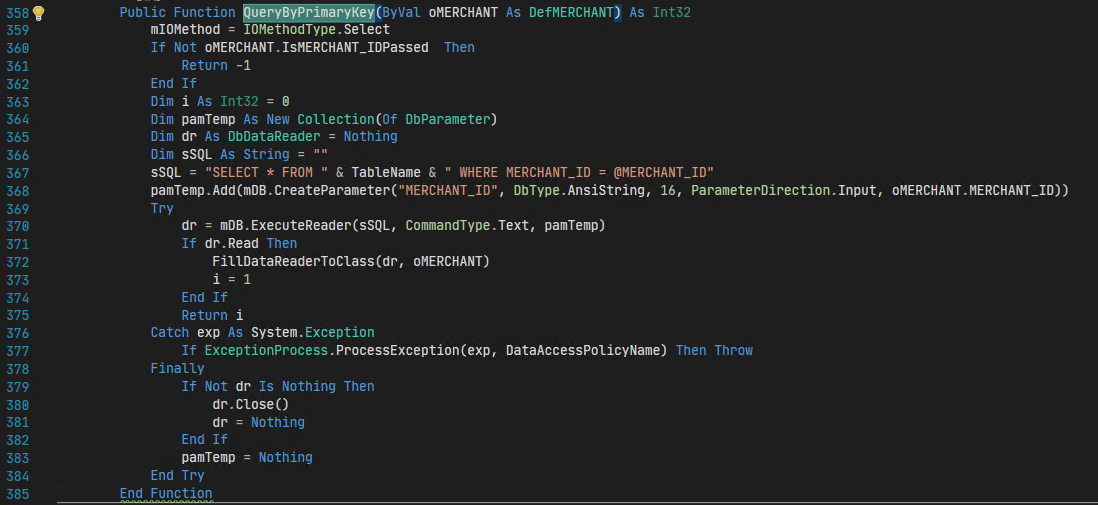
本文檔旨在闡述，本專案由.NET翻寫到Java出現的錯誤案例以及分析。

# .NET中資料庫資料查詢方法

## 案例1(FISC.vb)

* QueryByPrimaryKey傳入的是一個entity
* QueryByPrimaryKey傳入的entity按地址傳值，如果有查詢到資料會將查詢結果塞入entity
* QueryByPrimaryKey會返回查詢的資料筆數

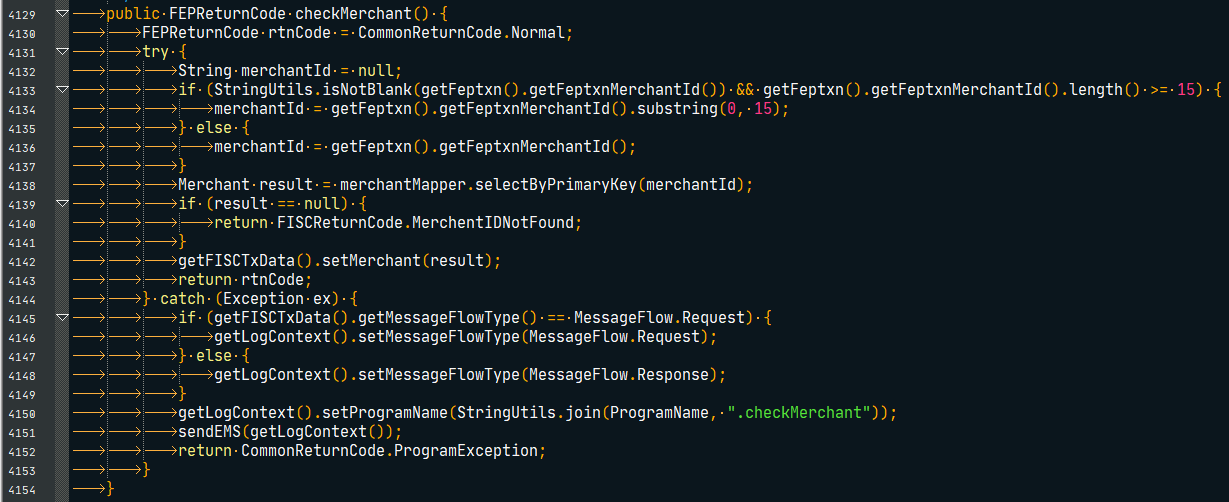




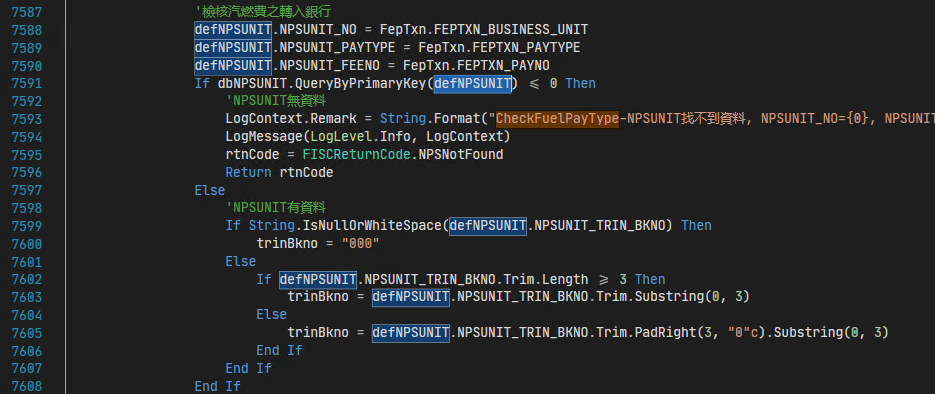
* 4131行不需要new entity。
* 4145行不需要getValue進行比較。



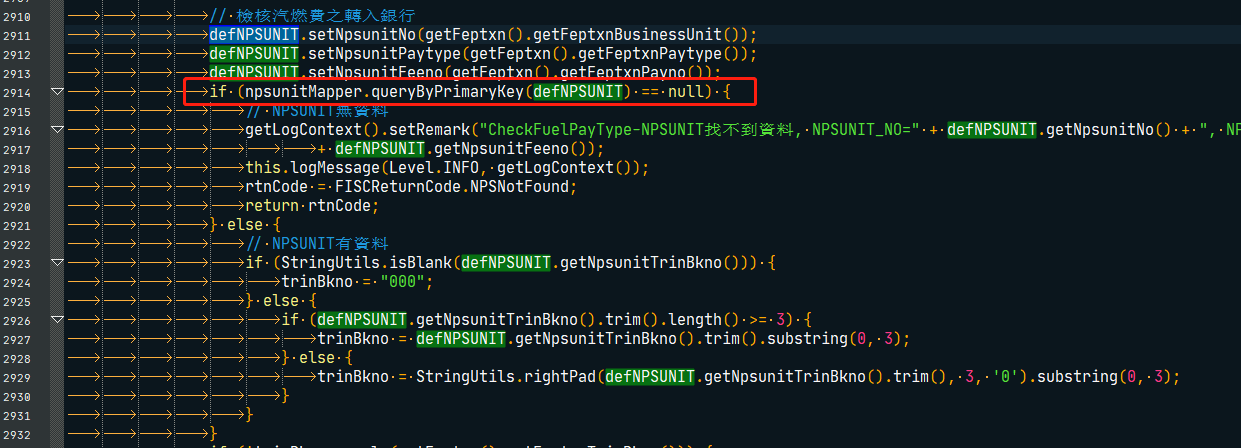
* 修改後的程式



## 案例2(FISC.vb)



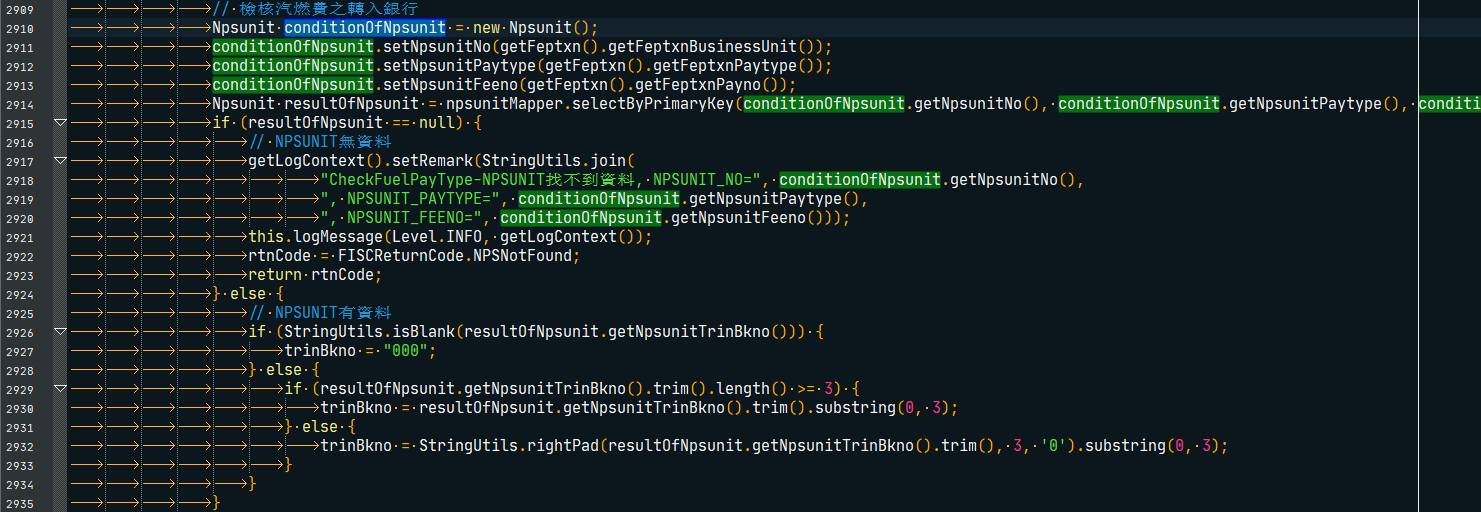
* 完全仿照，2914行傳入的entity會得到查詢的結果嗎？（應該是查了一個寂寞）



* 2914行，將查詢的結果又賦值給defNPSUNIT，那麼2917行會有NullPointException。

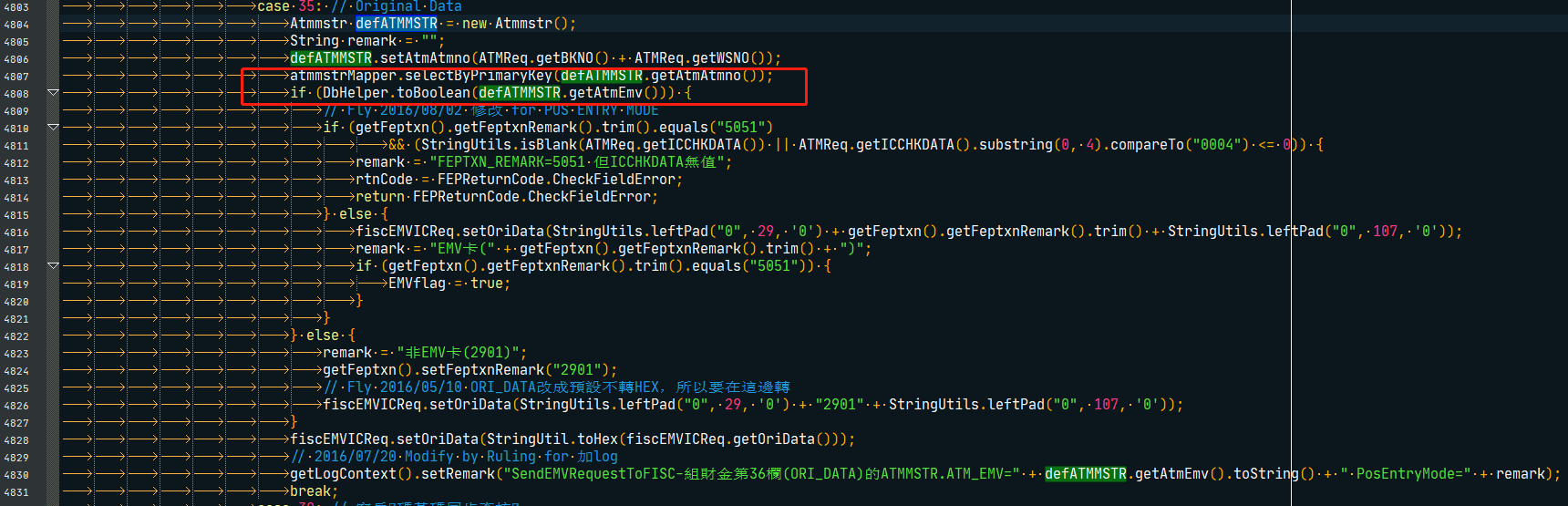


* 修改後的程式



## 案例3(FISC.vb)

* 與案例2是一樣的問題，下圖4808行的if永遠都進不去。

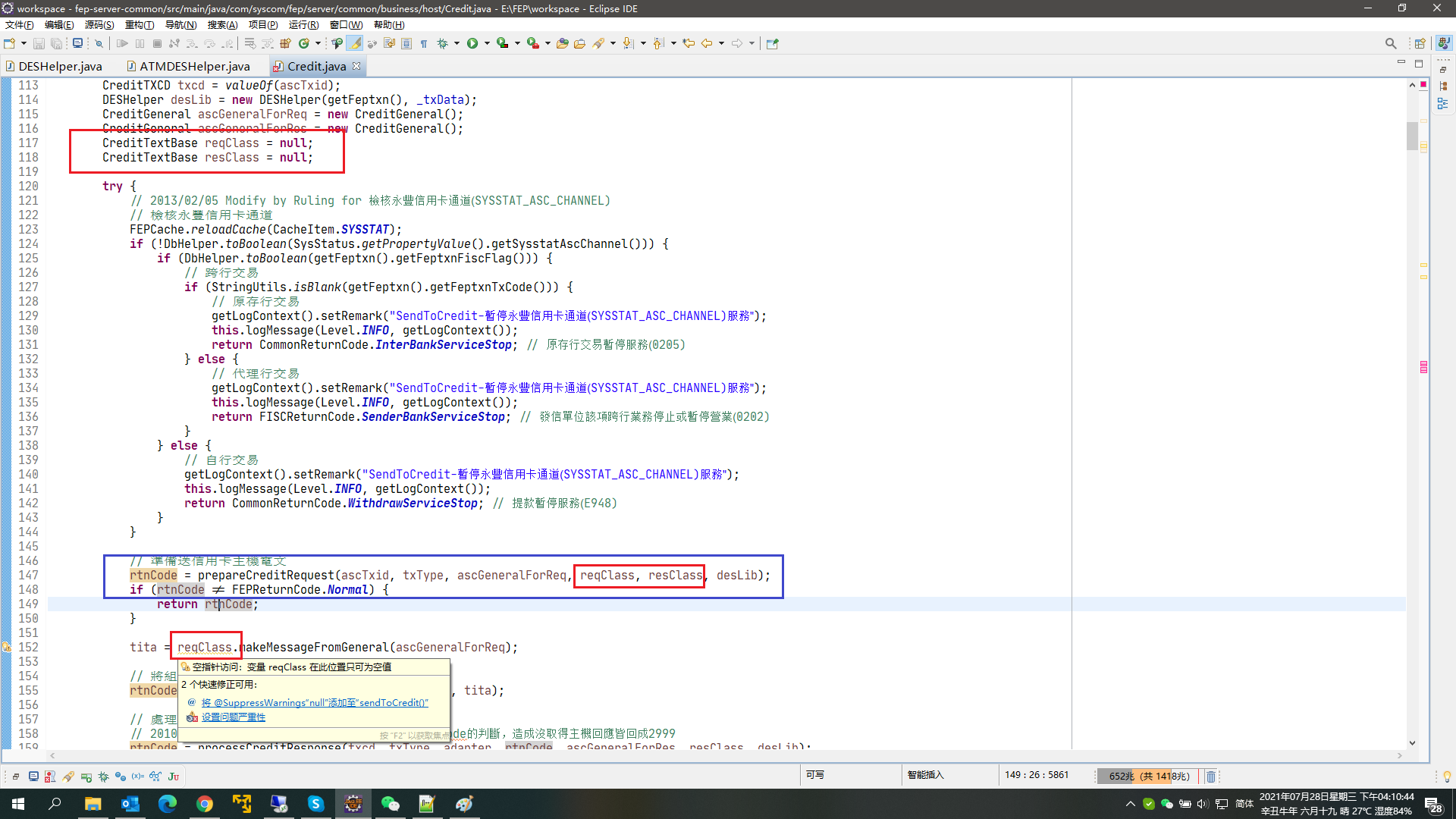


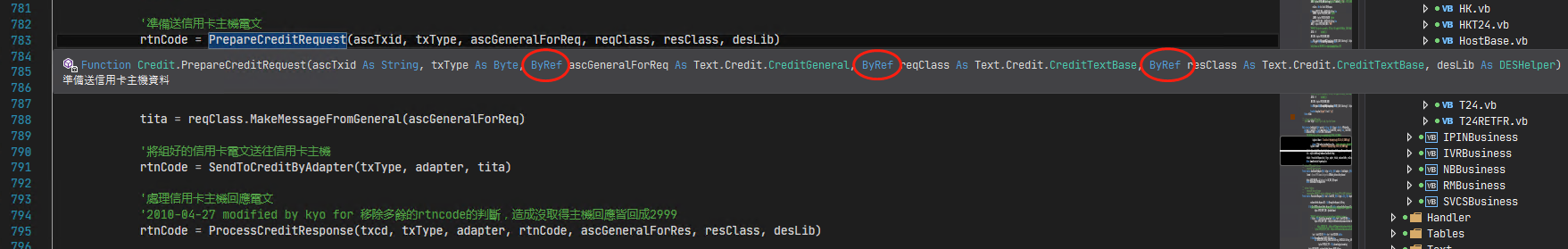
* 修改後的程式
  + 保險起見，4805行加入是否為null的判斷
  + 4809行feptxnRemark單獨提出來並加入是否為空串的判斷，避免直接trim出現NullPointException
  + 4810行的equals將常量放在左邊
  + 4831行原本entity.getAtmEmv().toString()改為DbHelper.toBoolean(resultOfAtmmstr.getAtmEmv())，同樣是為了避免toString出現NullPointException，同時也符合原本.NET程式需要列印布林結果的要求。



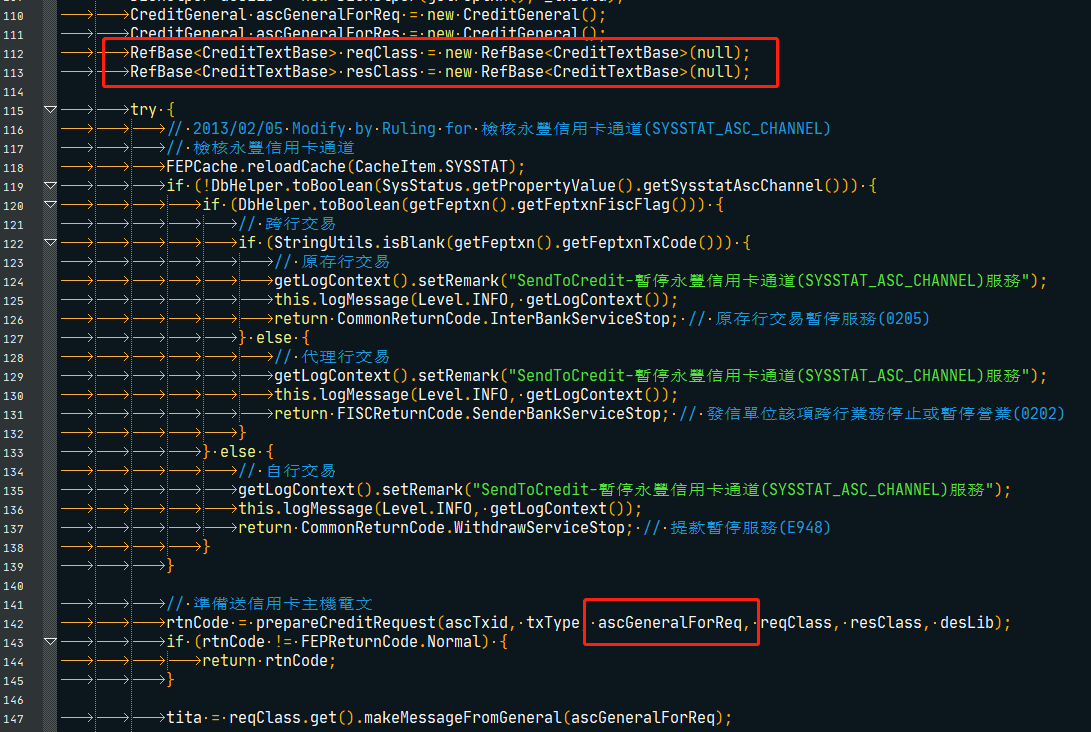
# 不要忽略Warning

* 下圖152行提示，變數只能是空值，會引發NullPointException，原因就是117和118行定義的兩個變數聲明為null。
* 這種情況要注意看一下，經過分析只能是147行傳入的這兩個變數，會在prepareCreditRequest方法中進行賦值，故這兩個參數在prepareCreditRequest肯定是按地址傳值。



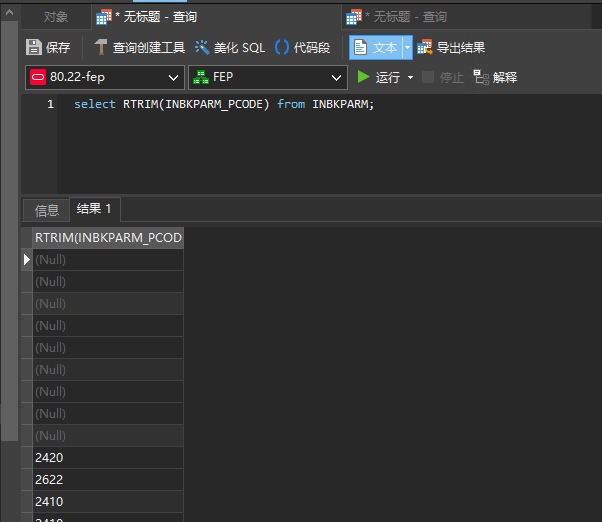


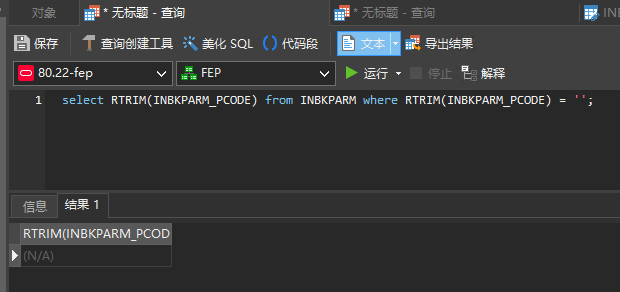
* 修改後的程式
  + 將reqClass和resClass包成RefBase物件，傳入方法中。
  + 不需要將ascGeneralForReq包成RefBase物件，因為110行已經有產生實體，在Java中物件本來就是按照位址傳值的。但是，必須在傳入方法前先產生實體，再傳入方法中。

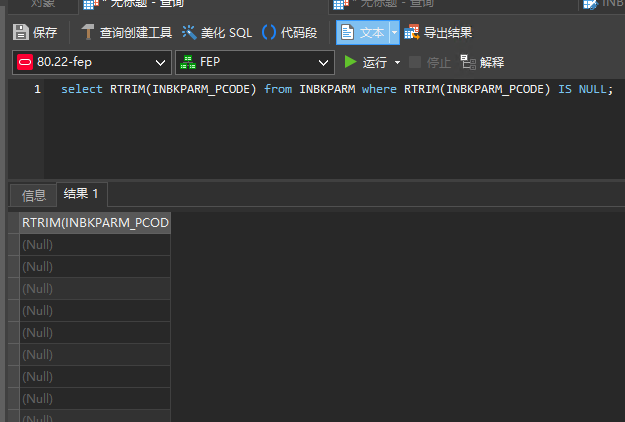
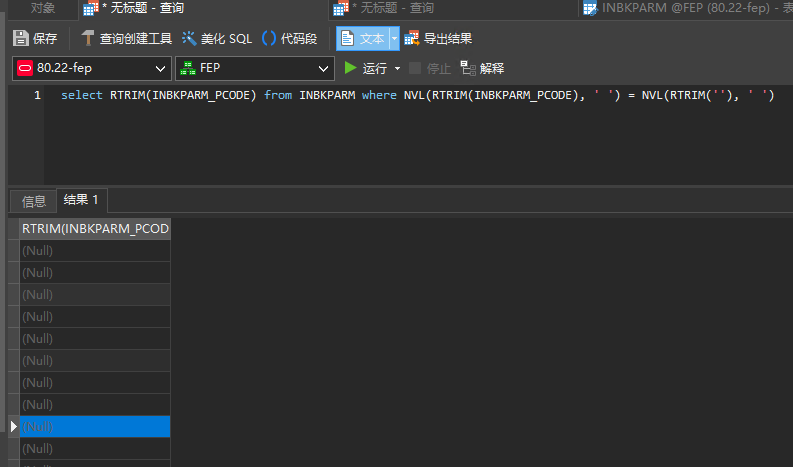


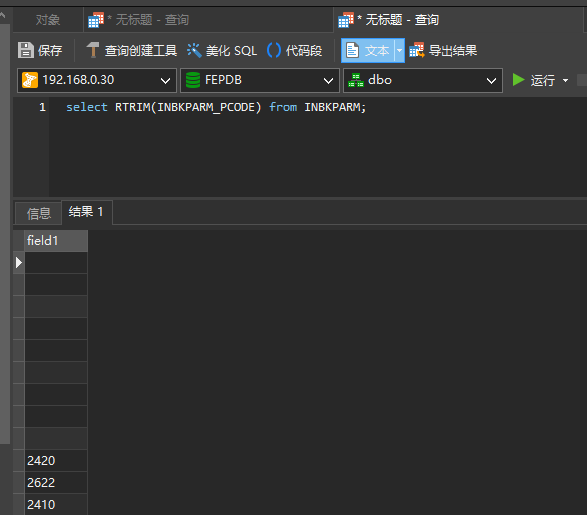
# MS SQL Server和Oracle，rtrim(char)函數差異

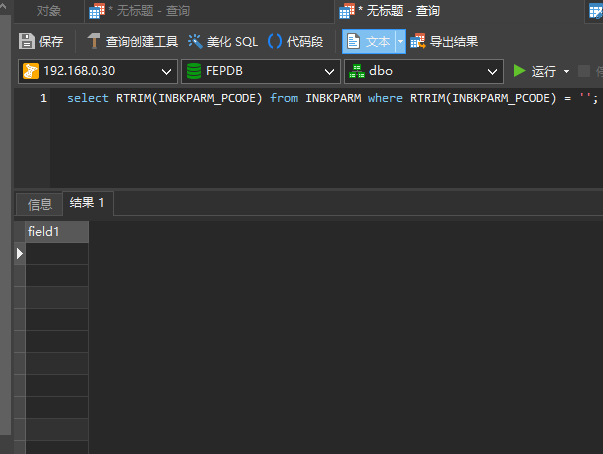




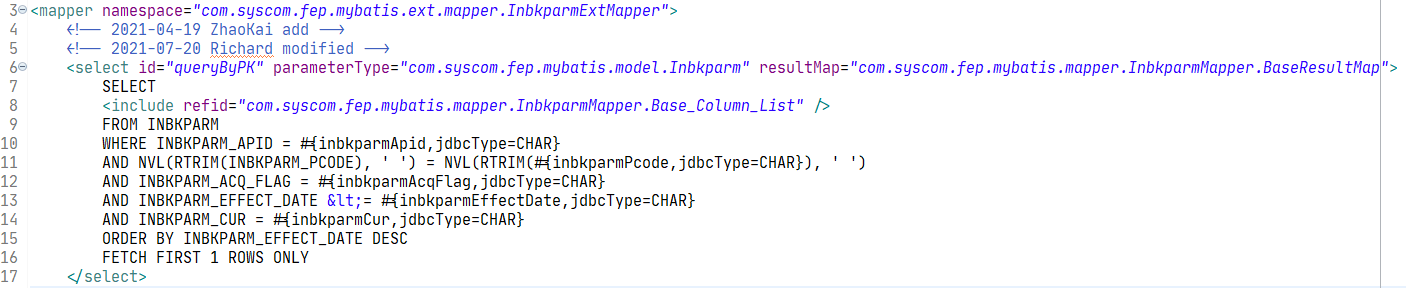






* 如果欄位是char，值為含有空格的或者是空白字串，oracle中，rtrim(char)，則返回Null，但是sqlserver返回是空白字串。
* 另外， 在oracle中，如果是select rtrim(char)，得到的是Null，沒有問題，但是在where中，如果rtrim(char) = rtrim('')，也就是Null = Null是不成立的，只能寫成rtrim(char) IS NULL。
* 請大家後面翻寫程式時，一定注意sql中rtrim的欄位是char類型的，建議大家可以寫成NVL( RTRIM( char欄位 ), ' ' ) = NVL( RTRIM( 傳入的字串參數 ), ' ' )

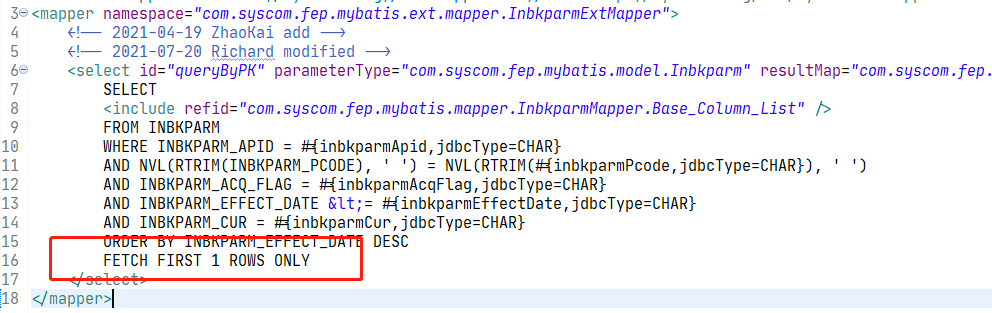


# 查詢資料只返回一筆

* 下圖紅框，if只是取第一筆資料。

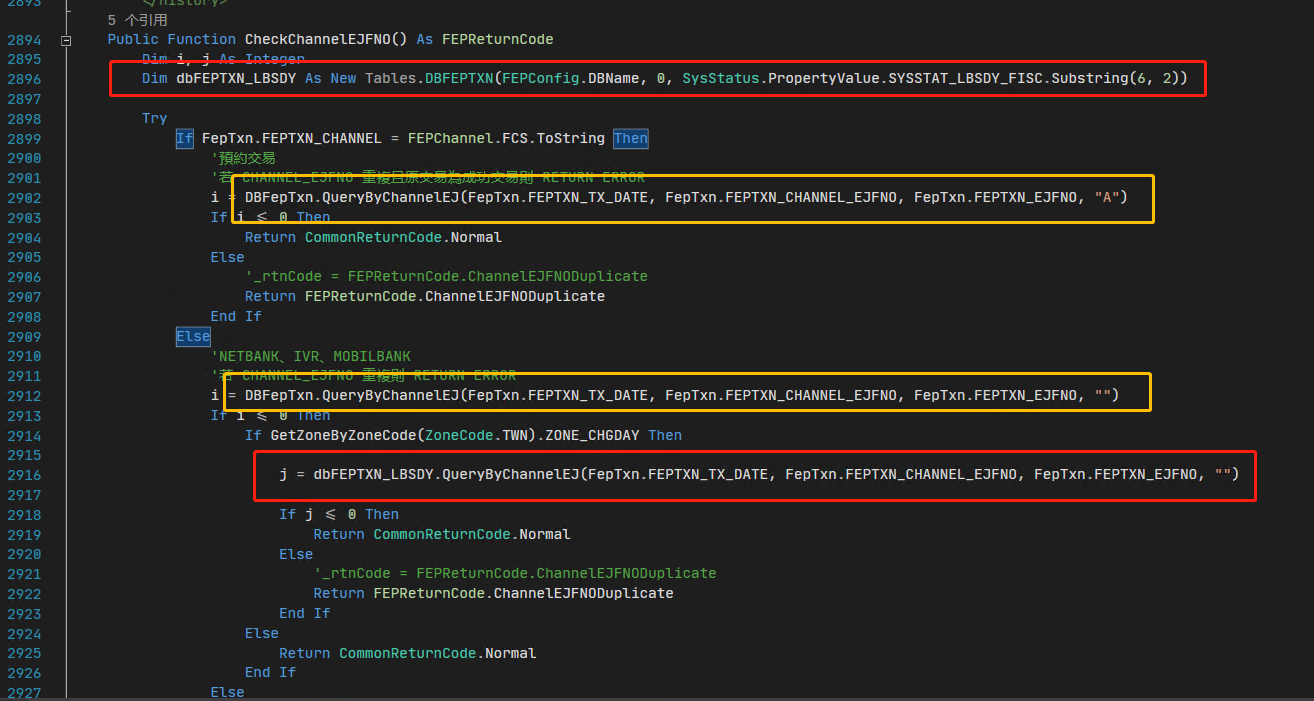


* 修改後的程式，SQL最後要加上FETCH FIRST 1 ROWS ONLY，並且mapper方法返回entity，而不是List<entity>。

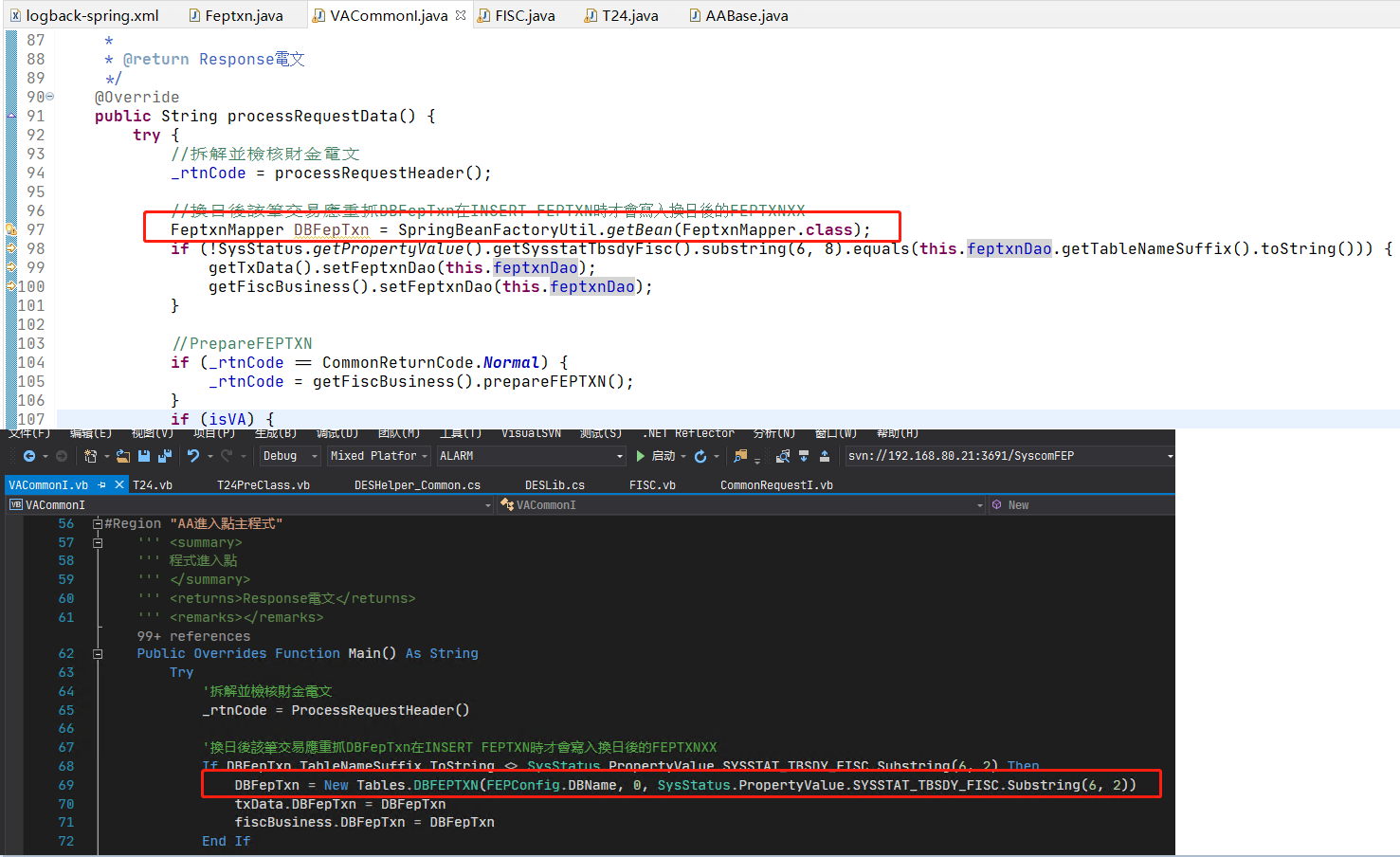


# FEPTXN表的操作

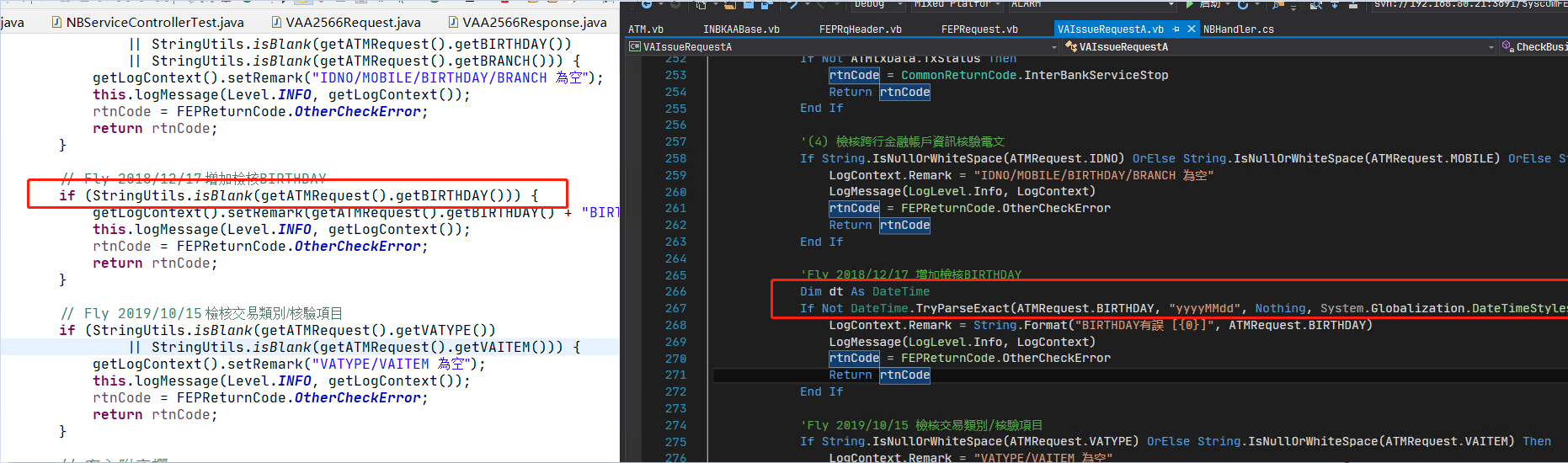
* FEPTXN表一共有31個，每天對應一個表，所以有時候根據業務需求，可能會查詢不同的表的，下圖一，紅色框和黃色框實際查的是不同的FEPTXN表，所以下圖一紅框中的代碼，對應則是下圖二紅框中的代碼。

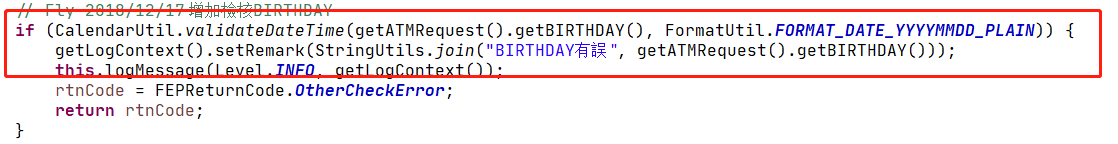




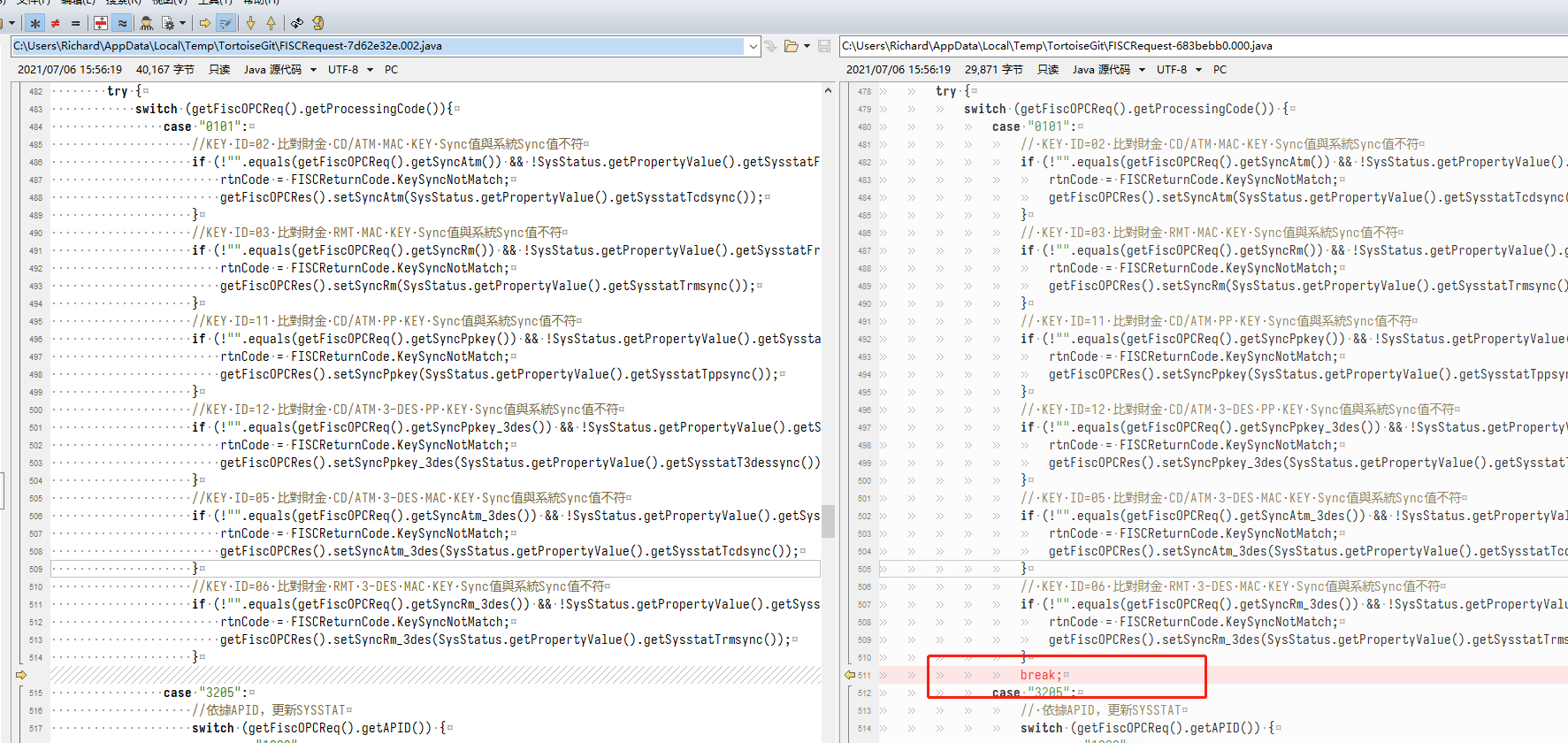


# 日期format檢核



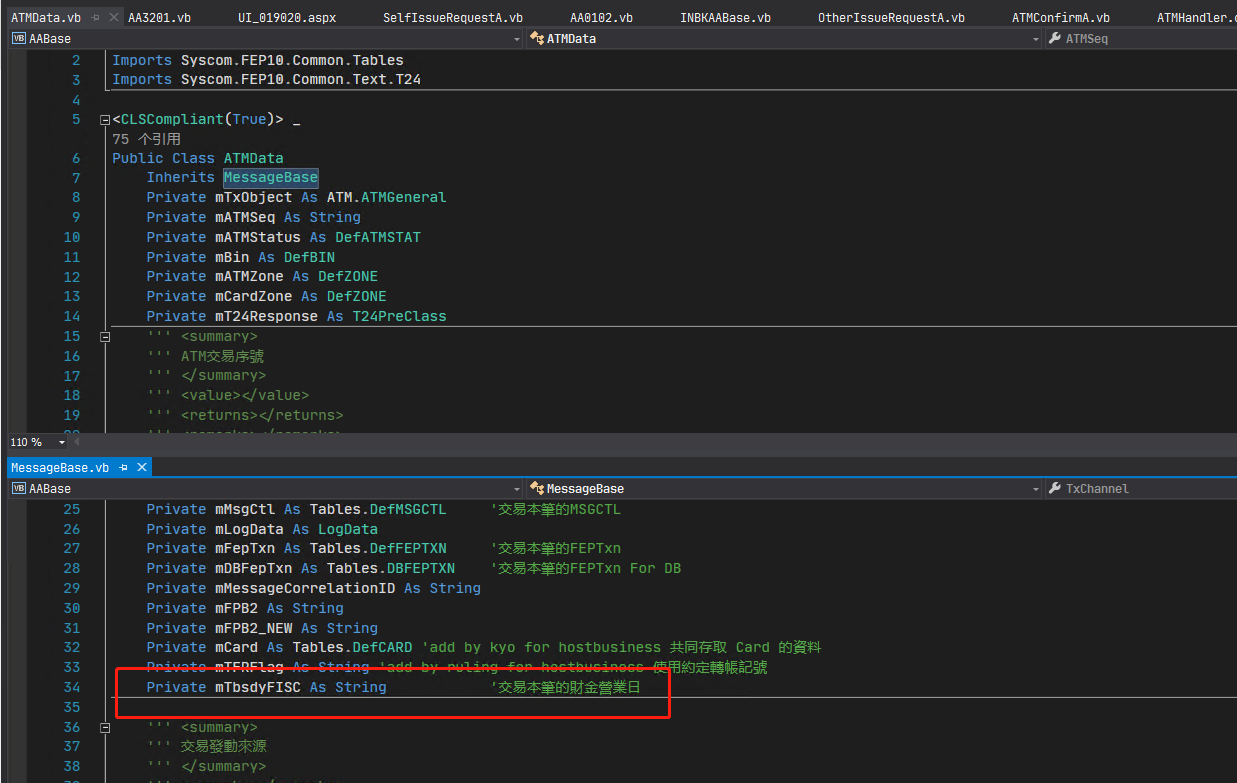


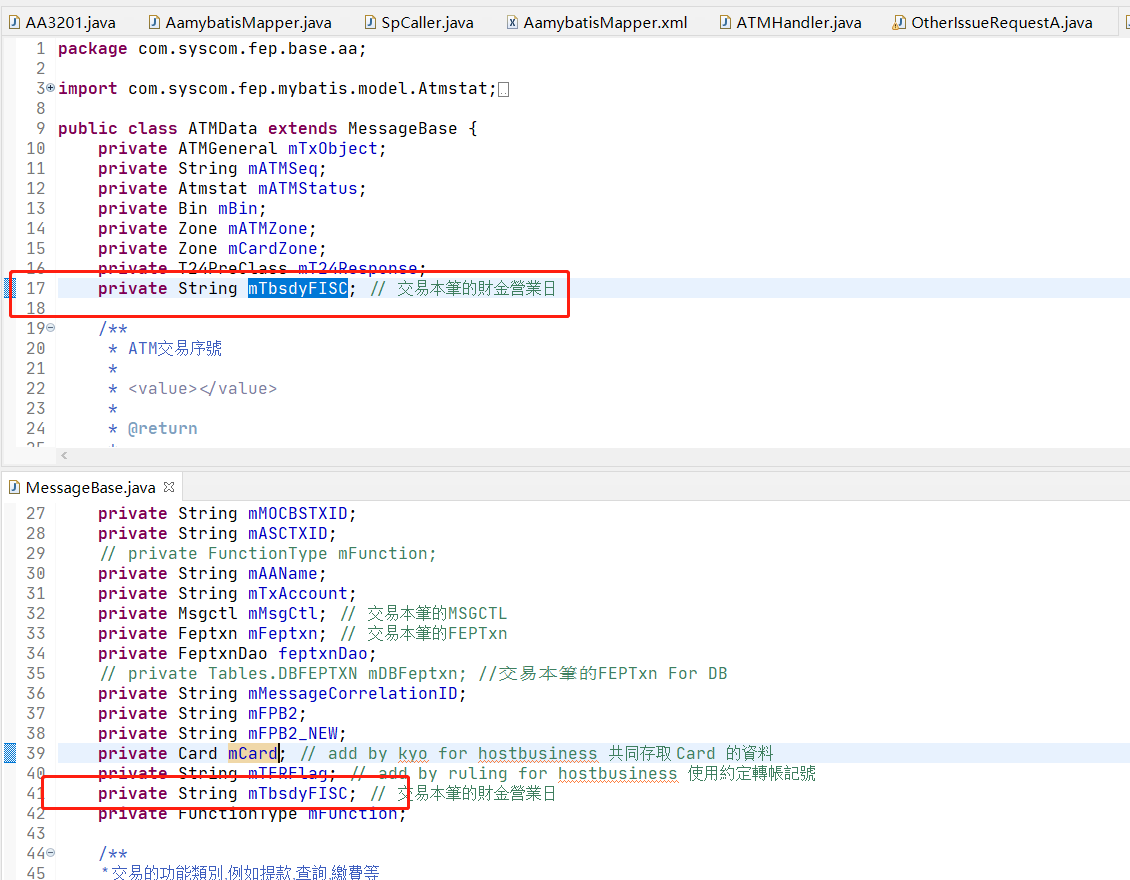
# case忘記break



# 屬性(或者方法)是在子類中還是父類中

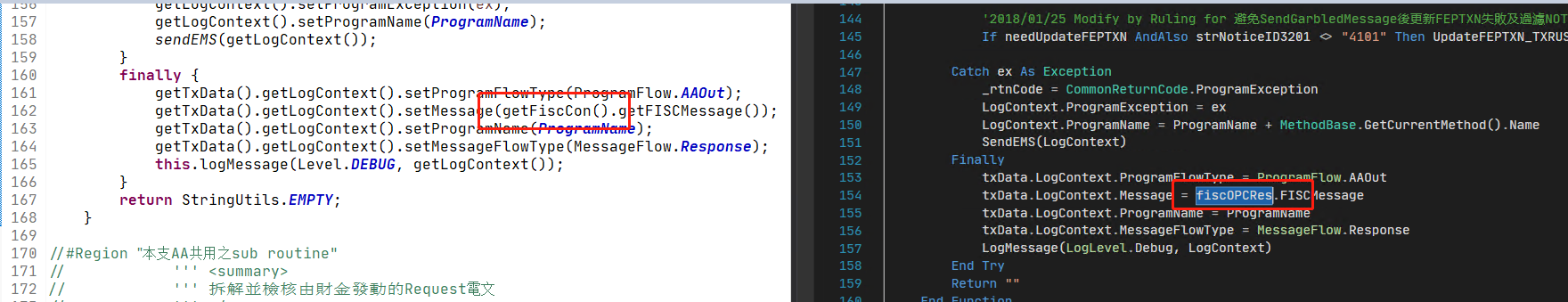
* 類中的屬性具體是在哪個類中，是在當前類(子類)中，還是在向上的哪一個父類中，不要像下圖這樣，重複了，後面就用錯了



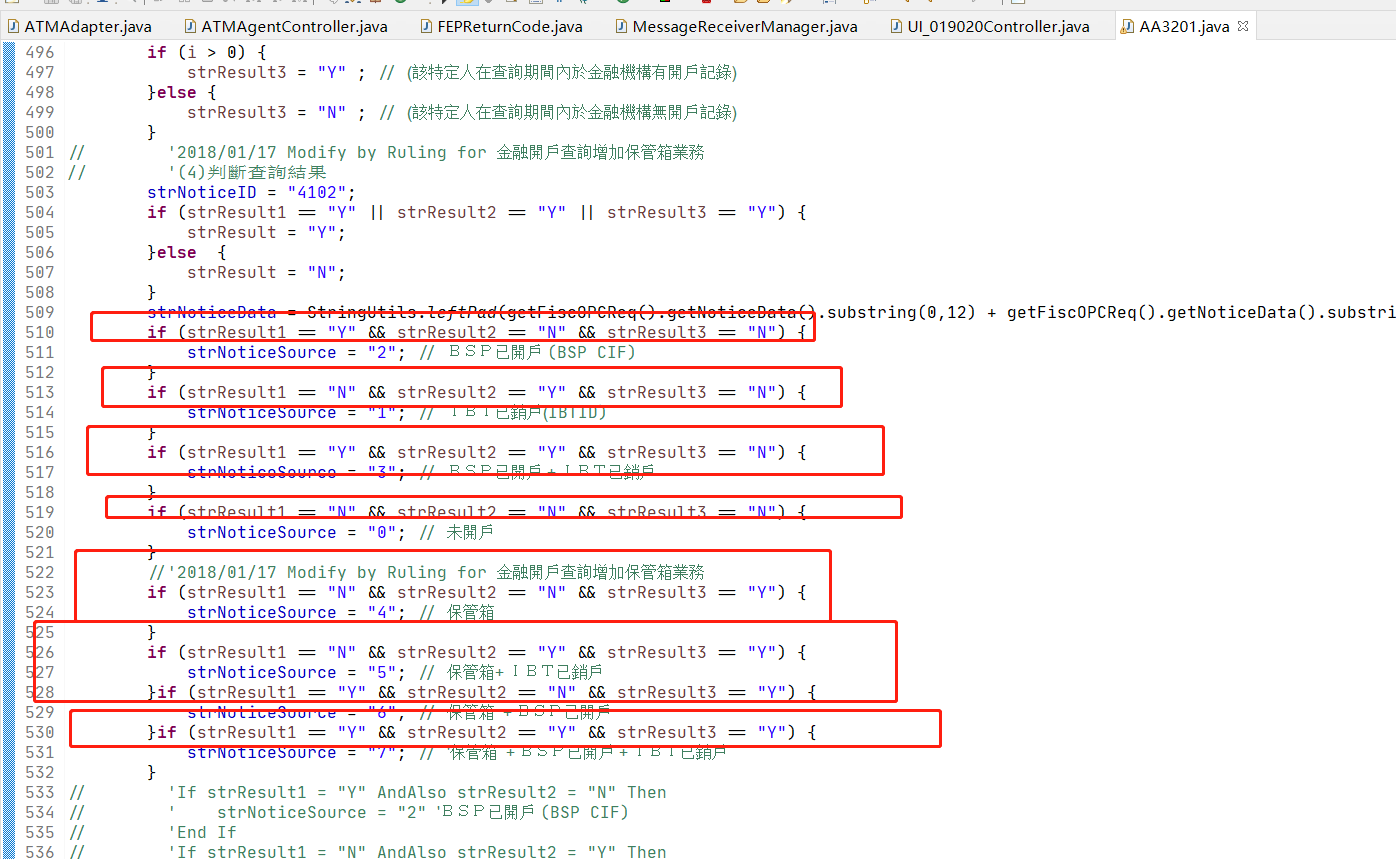


# 大家來找茬

* 下圖左右不仔細看，還真是看不出來，但是程式一跑，NullPointException就冒出來了。

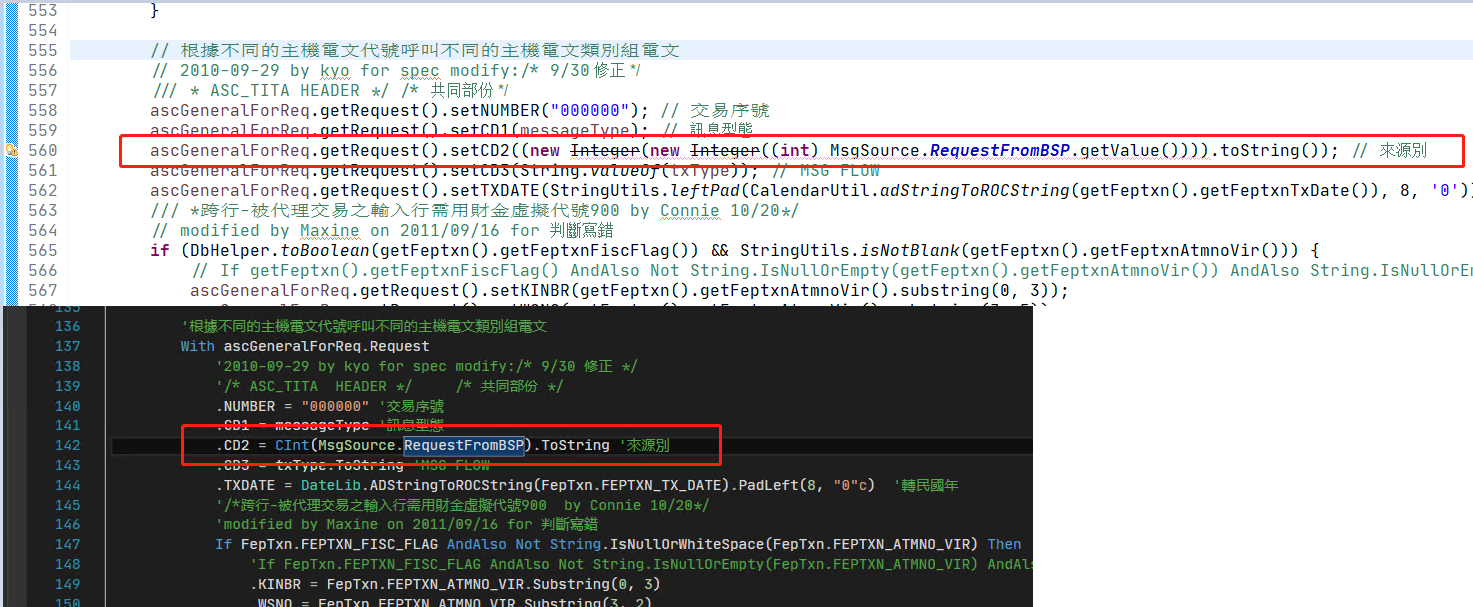


# 字串直接用等號



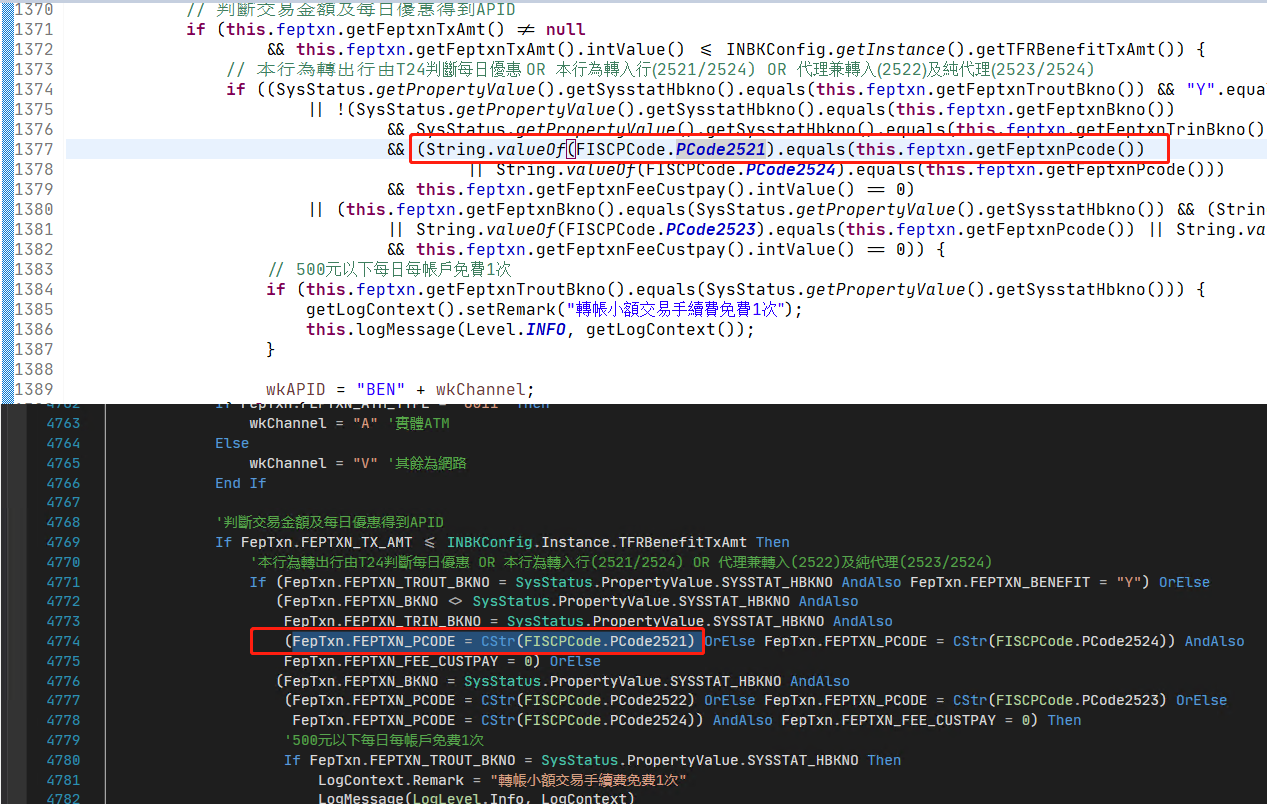
# 還是要看一下工具轉後的代碼

* 又是轉int，又是new Integer的，又是toString的，getValue已經是int了，然後再String.value()就可以了

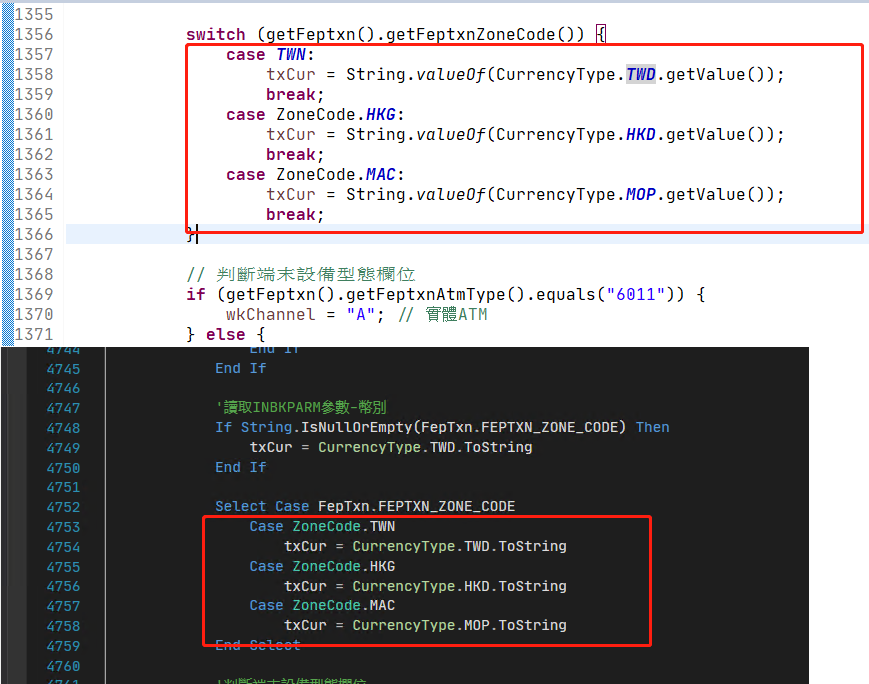


# 枚舉使用錯誤

* 如下圖，右邊.NET的CStr(FISCPCode.PCode2521)，實際就是枚舉的int數值轉為string，也就是等於2521，而右邊java的寫法String.valueOf(FISCPCode.PCode2521)，實際就是枚舉類型的toString()方法，也就是枚舉類型的name()方法，那麼就等於PCode2521，顯然是錯誤的，應該寫成String.valueOf(FISCPCode.PCode2521.getValue())，也就是使用getValue()取枚舉的int數值，再String.valueOf

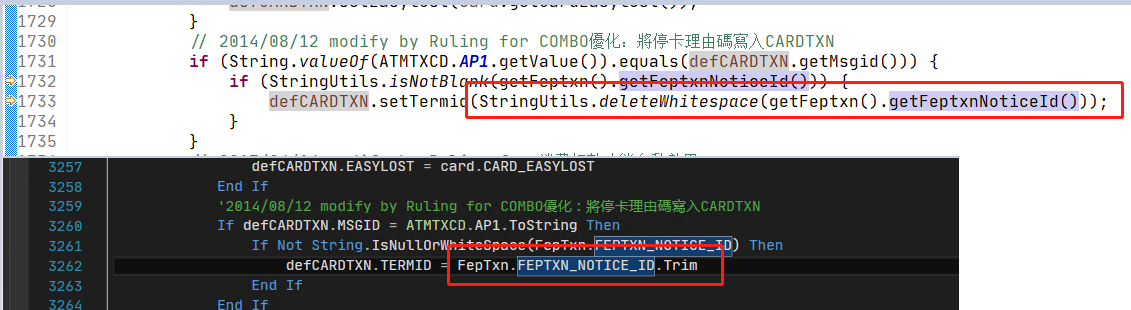


* .NET中枚舉類型ToString方法，類似java中枚舉的name()方法



# StringUtils.deleteWhitespace使用錯誤

* 這裡僅需要trim就可以了，不要使用StringUtils.deleteWhitespace，這個方法會Deletes all whitespaces from a String as defined by Character.isWhitespace(char).



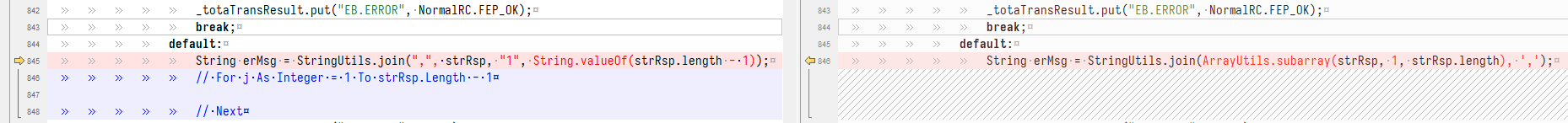
# Mapper中方法重複定義

* 已經有的方法，就不需要再特別添加了，另外既然是按照pk查詢，直接返回一個entity就可以了，不用返回List

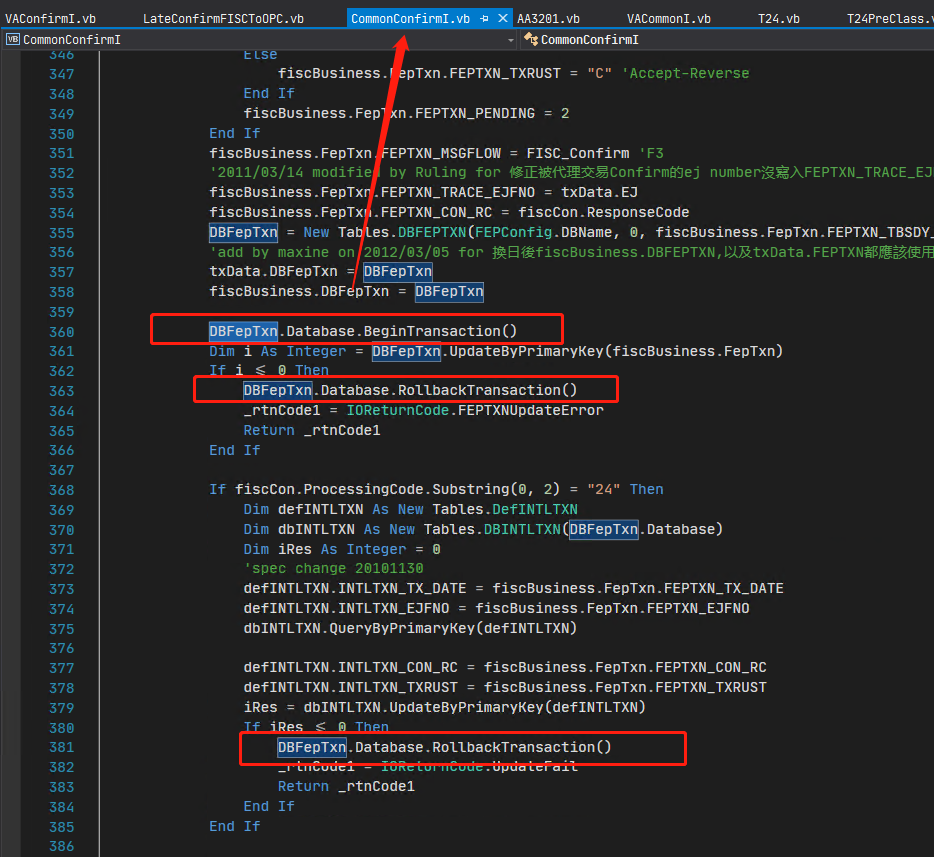


# StringUtils.join使用錯誤



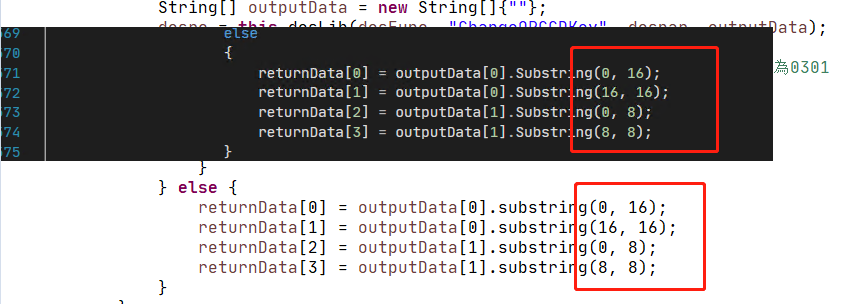


# 沒有使用事務





# substring第二個參數錯誤

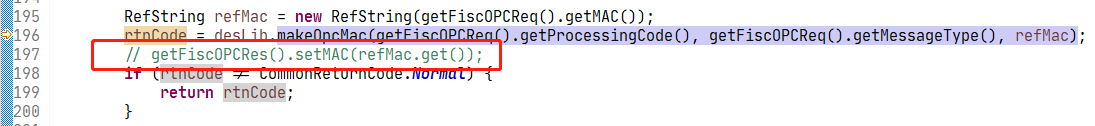


# 使用完Ref後要記得把結果塞回去

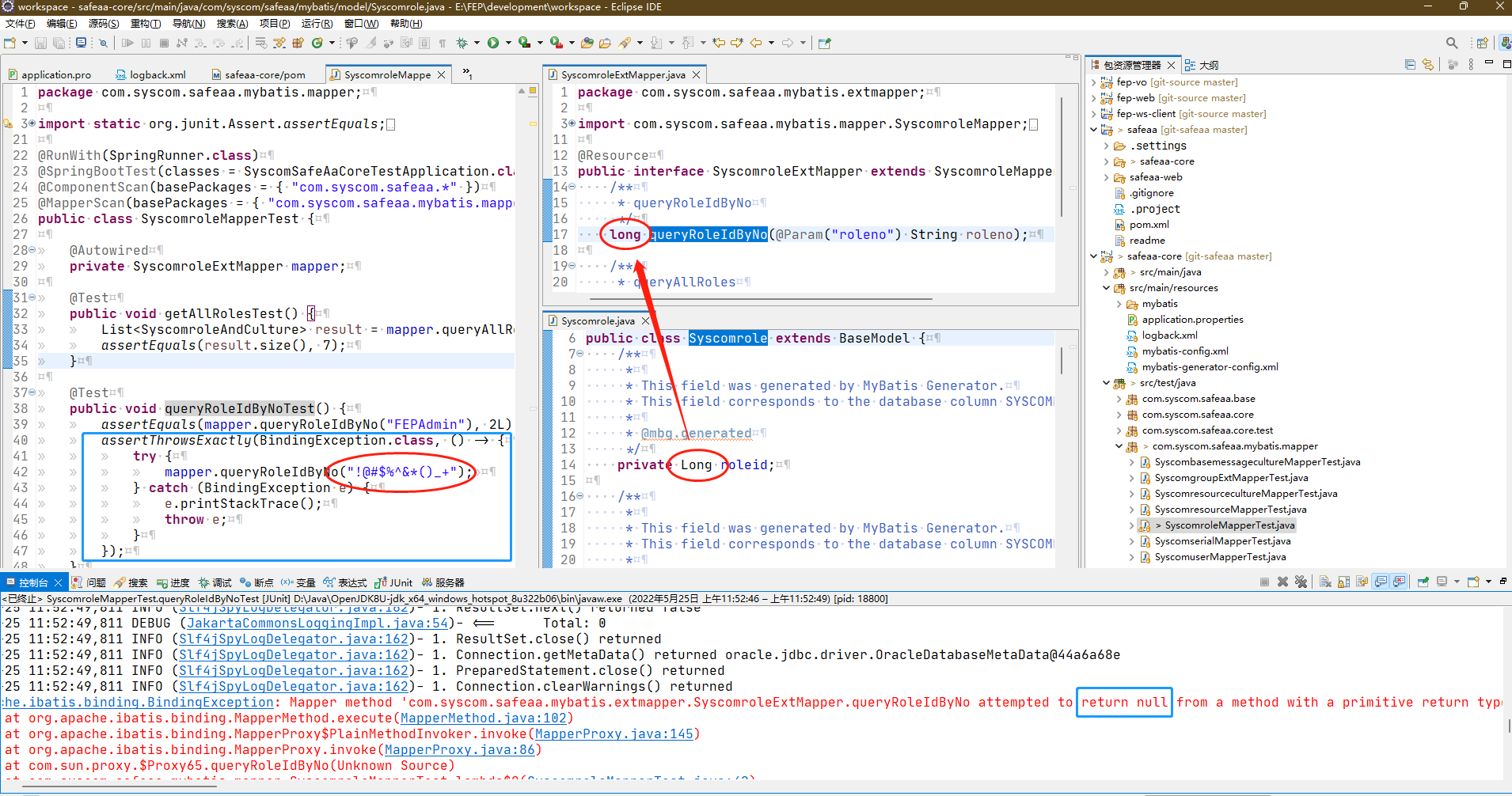
* mac來自fiscReq



* 下圖紅框不要mark，要把結果塞回給fiscReq
* 終於知道為啥mark掉了，因為翻寫錯了，應該是fiscOPCReq，卻寫成了fiscOPCRes，會報NullPointException，但實際上是翻寫錯了。



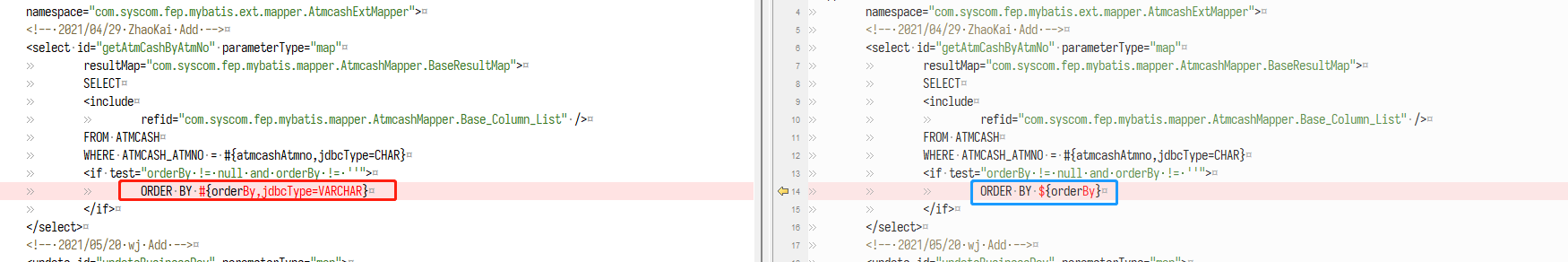
# Mybatis Ext Mapper中不要改變model中返回欄位的類型



* 以上圖舉例，原本RoleId是Long，那麼Ext Mapper中的查詢方法返回也要是Long，因為如果返回是long，一旦查不到資料，mybatis會自動將Null轉long，就會有異常。

# Mybatis XML中傳入Order By的寫法

* 如下圖左邊紅框是錯誤的寫法，應該是右邊藍框中的寫法，${orderBy}，不能寫成#符號，應該是$符號，並且不需要jdbcType。



# Mybatis 查詢方法是否應該返回List<>

* 如下圖的查詢方法，SQL中where條件有判斷，塞值的情況下才會有where語句，那麼這個方法按理返回List<Fepuser>才是正確的。

