Template Method

- 定義演算邏輯的骨幹(skeleton)及步驟, 將該步驟定義在一般化類別中,細節 則由特殊化類別來完成。
- 可以稱之為軟體設計中「留白的藝術」。





問題描述

許多的程式以及邏輯,常常會有一些特 性,就是有部分的邏輯相同,但在細節 上又不大一樣。 如何經由抽象,把相同邏輯抽象出來, 並且留下適當的空白,讓未來的開發者 可以發揮更多想像的空間?



解決方案

在一般化類別中,建立一個Template Method,在該Method中,定義出邏輯運算,並且使用到一些目前還未實作的方法 (abstract method)。



結構



AbstractClass

- + TemplateMethod(): void
 // do something
 PrimitiveOperation1();
 // do something
 PrimitiveOperation2();
 // do something
- + PrimitiveOperation1(): void
- + PrimitiveOperation2(): void

ConcreteClass

- + PrimitiveOperation1(): void
- + PrimitiveOperation2(): void

::AbstractClass

- + TemplateMethod(): void
- + PrimitiveOperation1(): void
- + PrimitiveOperation2(): void



說明

Template Method指的是在一般化類別中的Concrete Method,在該Method中,會用到Abstract Method。Template Method會使用的時間點,通常是在系統開發進入某個時間點後,Developer或是Architect發現到某些邏輯重複出現,此時,通常會利用Refactoring的手法來將Method抽象化。

像是Word中的Template(.dot)檔,在新增一個Word文件 (.doc, 即特殊化類別)時,可以先選擇用某一個Template,他會建構出包括框架以及常用的



案例描述

假設某保險公司針對保單的線上變更業務,有幾種不同的變更方式。

一種是可以直接針對保單進行借款,一種是可以針對保單來修改被保險人的名字。

針對客戶的所有保單,如果要借款的話,其保單的狀態必須是正常繳費;而要更改姓名的話,保單的狀態則必須不是已理賠。

進行線上變更時,保單必須分成可異動以及不可異動的列表,要如何達成上述的需求呢?







