

xUnit Test Lab 2

Write the test stub for a class with dependency

說明：

很多時候，你正在實作的 class 必須依賴其他 class，object，library 的存在。但是這些 entities 有可能：

1. 還不存在（團隊合作的過程中，還在等你的隊友實作）
2. 就算存在，但是如果你要將他 include 到你的單元測試中，你的測試代價會非常的高
3. 雖然存在，但是如果你建立這些物件的代價非常高昂。

在單元測試中，為了解決這些問題，你通常必須在你的 class under test 進行必要的 refactor 來新增一些介面。（註：老師上課如果你有在聽，你應該沒有問題）。在這個練習中，你將練習如何為一個 class 撰寫 test double 中的 test stub。

你的任務：

1. 先按照規格(後面解說) 將你的 class under test 撰寫出來。
2. 撰寫你的單元測試
3. 然後你會發現有些外部物件 X 的建立符合說明中的三項條件，你的 test 以及 class under test 可能無法通過編譯來進行測試。
4. Refactor 你的 class under test（註：NO idea? 上課恍神? 請參考老師的投影片，或儘快請教同學了解背後的原理）
5. 撰寫一個 test stub for X
6. 讓測試可以通過編譯，並且讓你的測試可以執行?
7. 你的成績將由助教根據
 - 甲、是否成功地實作 class under test
 - 乙、是否撰寫了完整的 unit tests for class under test
 - 丙、是否撰寫了 test stubs，並且展示如何利用 test stub 來測試 class under test

提示：

1. 有兩個 test stub 需要撰寫
2. 你的 class under test 不應該知道 test stub 的存在，但是利用

interface 讓 class under test 藉由 test stub 的存在進行測試，而無需等待外部物件的建立

3. 寫完請專注撰寫完整的 unit tests

Class under test 的規格

你的團隊正在負責進行一個 **OoooCalendar** APP 的實作。**OoooCalendar** 若成功了將為公司帶來驚人的營收（好啦，有夢最美）。很幸運的，你的團隊是一個軟體工程非常成熟的團隊，你也一直非常自豪你是一個有紀律的 team player，而不是一個無法控制的瘋狂 programmer。

經過你的團隊的一番討論與設計，你需要負責實作一個與時鐘，時區的 class 叫做 **OoooSmartClock**。

這個 class 的 interfaces 以及細部規格描述如下：

```
class OoooSmartClock {

    public void setTimeZone() {
        // 當這個 method 被呼叫的時候會有一個 dialog 跳出來
        // 讓使用者選一個時區。這個 dialog 由你的同事來實作
        // 大致上是讓使用者從成千上百的城市選取，但是你的同事
        // 要負責幫你轉換成標準時區
        // 事先約定好的呼叫的方式如下：
        //     int zoneindex = new OoooSetTimeZoneDialog()
        // 全球的 time zone 共被劃分成 24 個時區，所以正常
        // 結果 OoooSetTimeZoneDialog() 會回傳 0-23
        // 0 代表換日線的最開始時區
        // 如果使用者沒有選擇，則會有個 exception NODATA 從
        // OoooSetTimeZoneDialog() 丟出來
    }

    public void setTimeZone(int index) {
        // nothing to explain, a simple setter
    }

    public string getCurrentTimeStamp() {
        // 當這個 method 被呼叫的時候請回傳
        // YYYY-MM-DD HH:MI:Sec 形式的 format
        // 例如 "2009-09-22 16:47:08"
        // (HINT:use "Date" and "SimpleDateFormat" from Java
    }
}
```

```
Public string getLocalCurrentTimeStamp() {  
    // 這個 method 被呼叫的時候會回傳比較人性的字串  
    // 1. 當時間介於 12:00:00-12:01:00 的時候會回傳  
    //     "2009-09-22 NOON"  
    // 2. 當時間介於 00:00:00-00:01:00 的時候會回傳  
    //     "2009-09-22 MIDNIGHT"  
    // 3. 當日期是國慶日的时候回傳  
    //     "2009-10-10 DOUBLE-TEN 16:37:08"  
    // 4. 當日期是父親節的时候回傳  
    //     "2009-08-08 FATHER'S DAY 16:37:08"  
    // 5. 當日期是聖誕節時回傳  
    //     "2009-12-25 X'MAS 16:37:08"  
    // 以上日期的變化與時間會互相結合  
    // 所以 "2009-12-25 X'MAS NOON" 是正確的字串如果時間  
    // 落在以上的條件的綜合  
  
}
```