|  |
| --- |
| GTTlipse 使用說明 |
|  |
|  |
| GTTlipse 使用及操作指導手冊 |
|  |
| **Jung-Chi Wang** |
| **2010/02/04** |
|  |

目錄

[1. 建立 Java 專案 2](#_Toc253058821)

[2. 設定 GTTlipse 和該專案關聯 3](#_Toc253058822)

[3. 建立 GUI 測試程式 5](#_Toc253058823)

[4. 編輯 GUI Test Script 7](#_Toc253058824)

[5. 使用錄影方式產生 Script 15](#_Toc253058825)

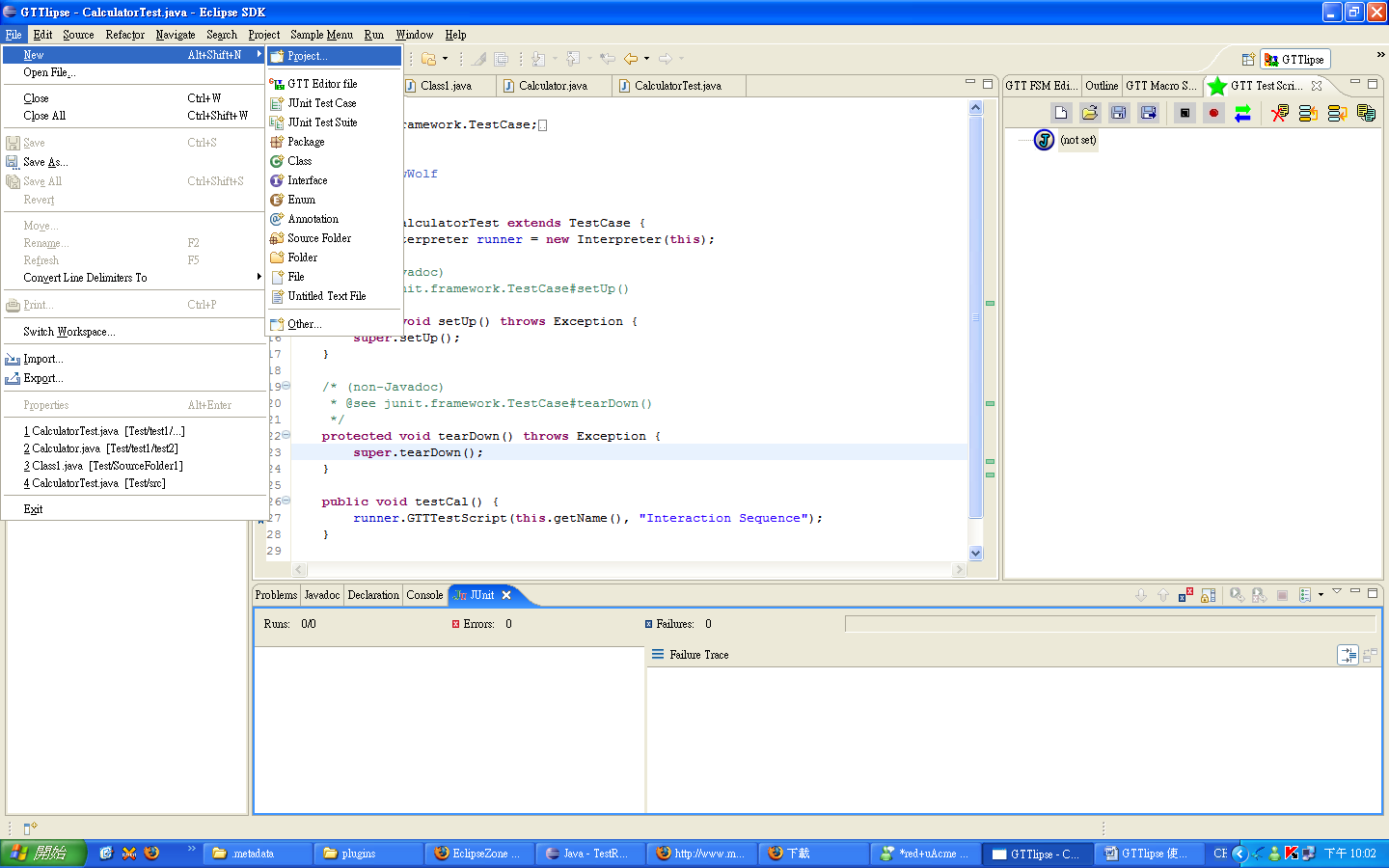
[6. 插入驗證節點 17](#_Toc253058826)

[7. 執行測試 19](#_Toc253058827)

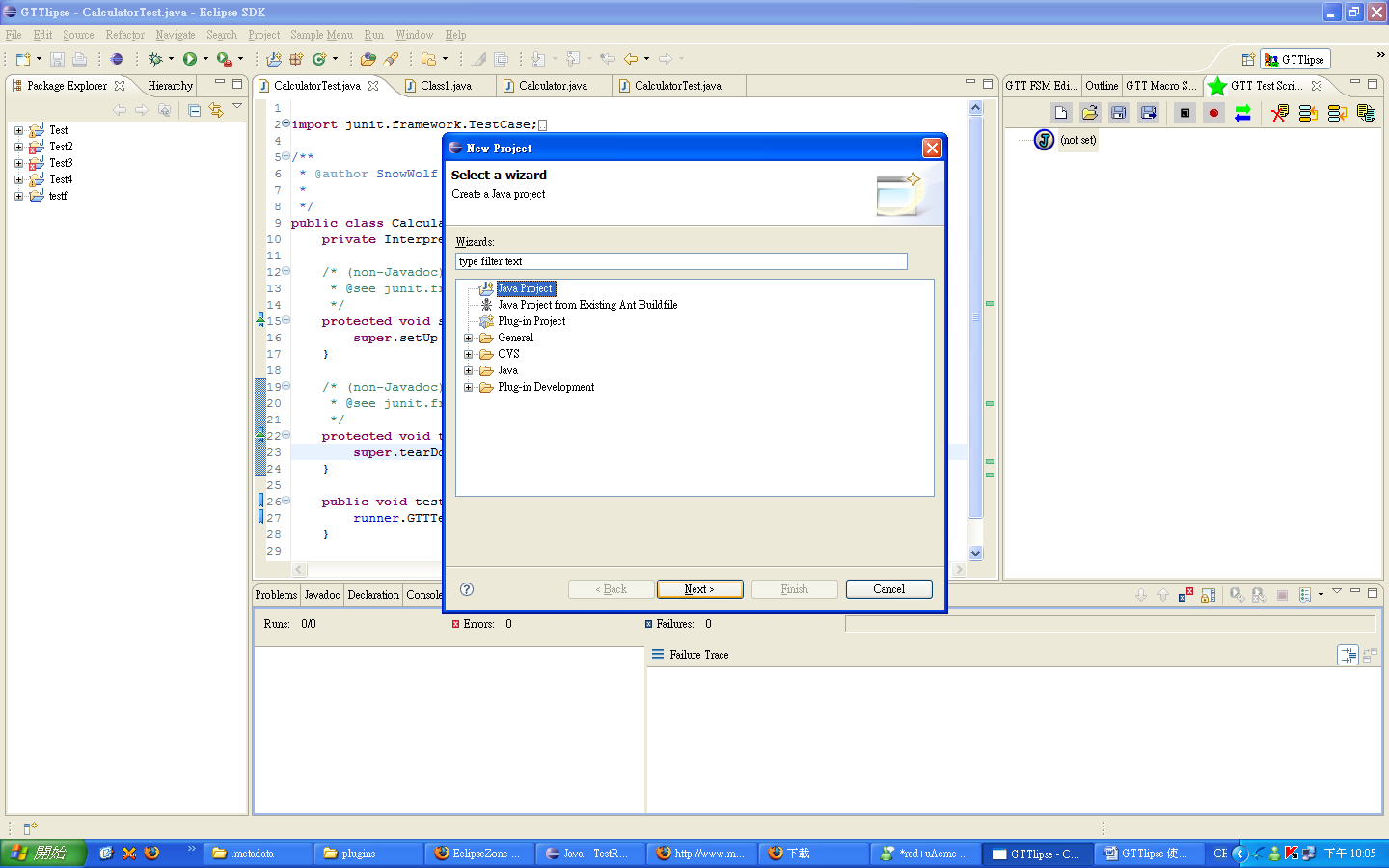
1. 建立 Java 專案

如果你已經有現成的專案可以使用，則你可以略過此一步驟。

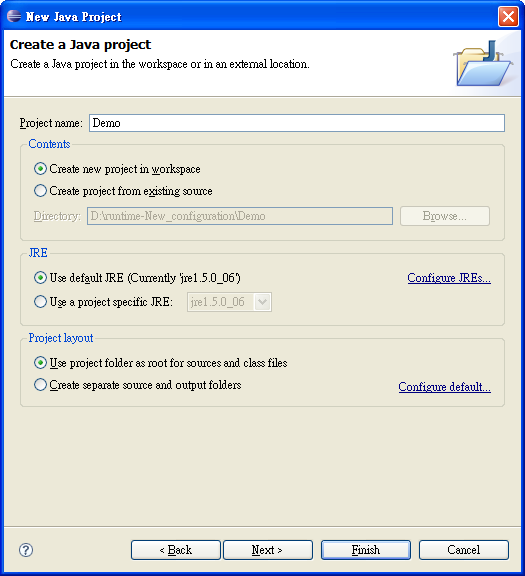
* 1. 開啟檔案後選擇 File=>New=>Project



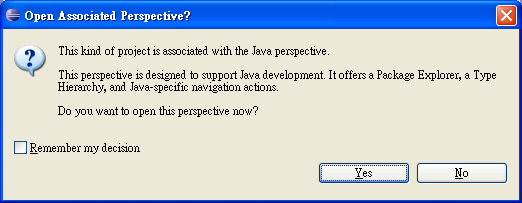
* 1. 選擇 Java Project 並按下 Next



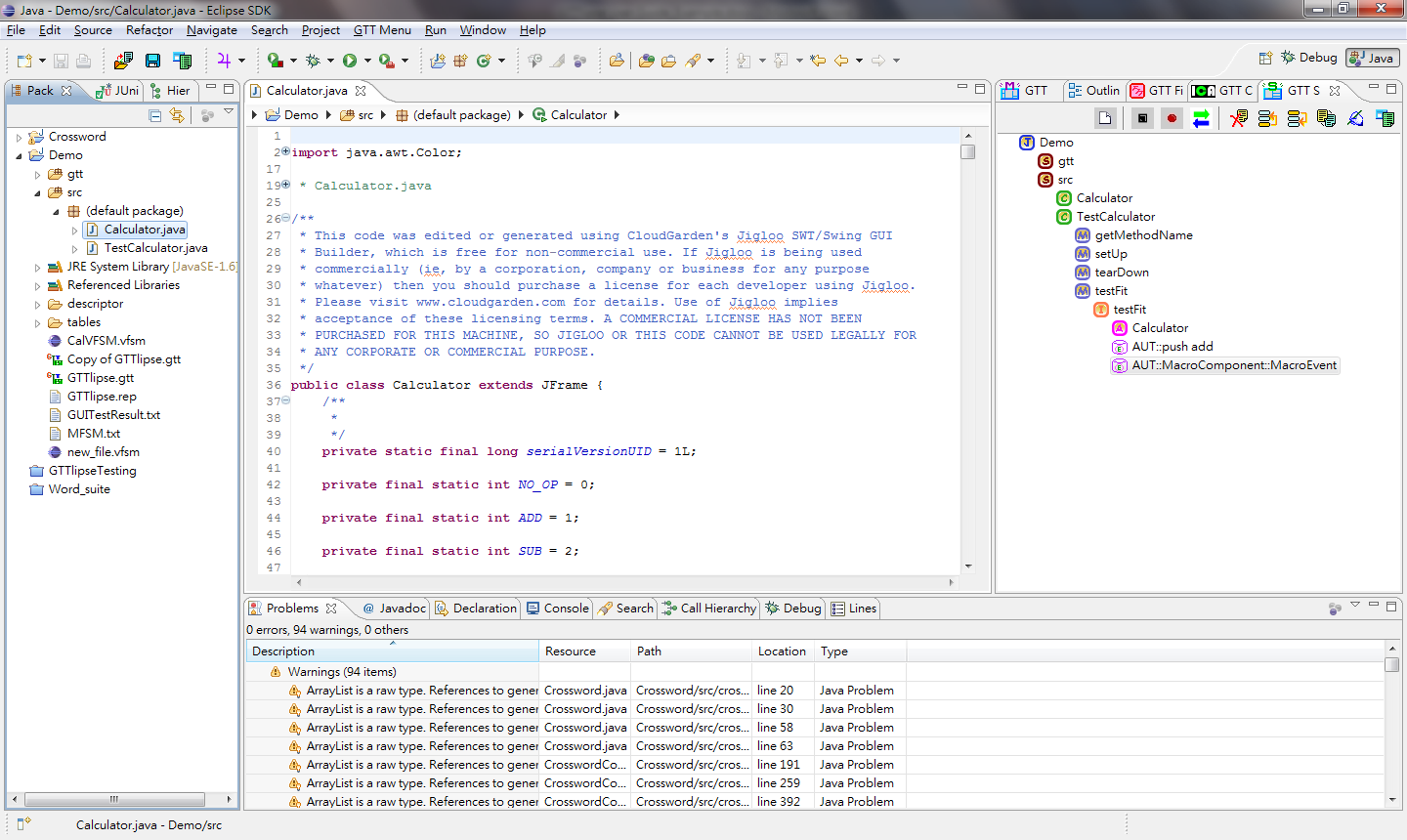
* 1. 輸入專案名稱 “Demo”，並按下 Finish 完成專案的建立



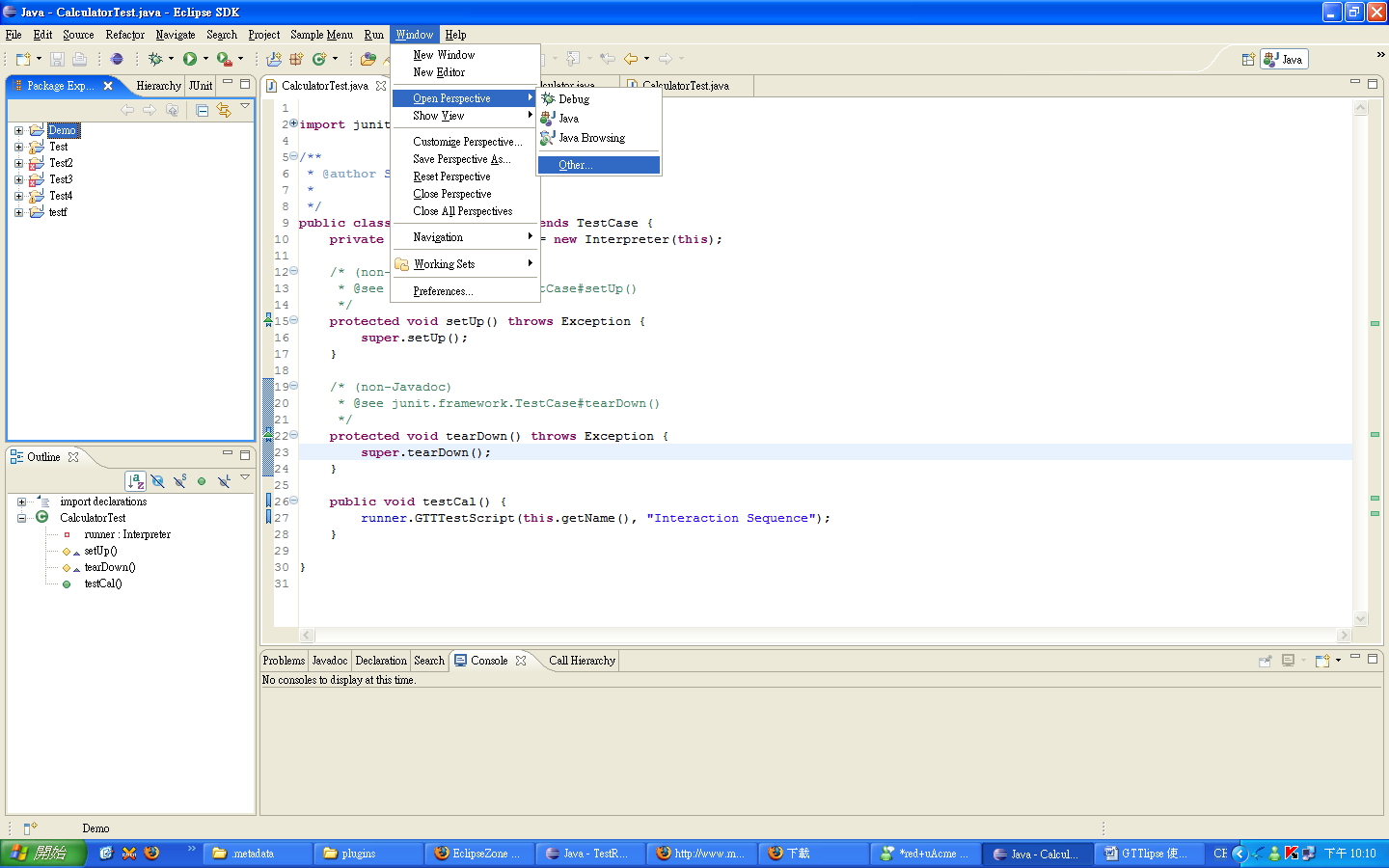
* 1. 切換到 Java 視景，選 Yes



* 1. 將小算盤的程式碼 Calculator.java 加入到專案中



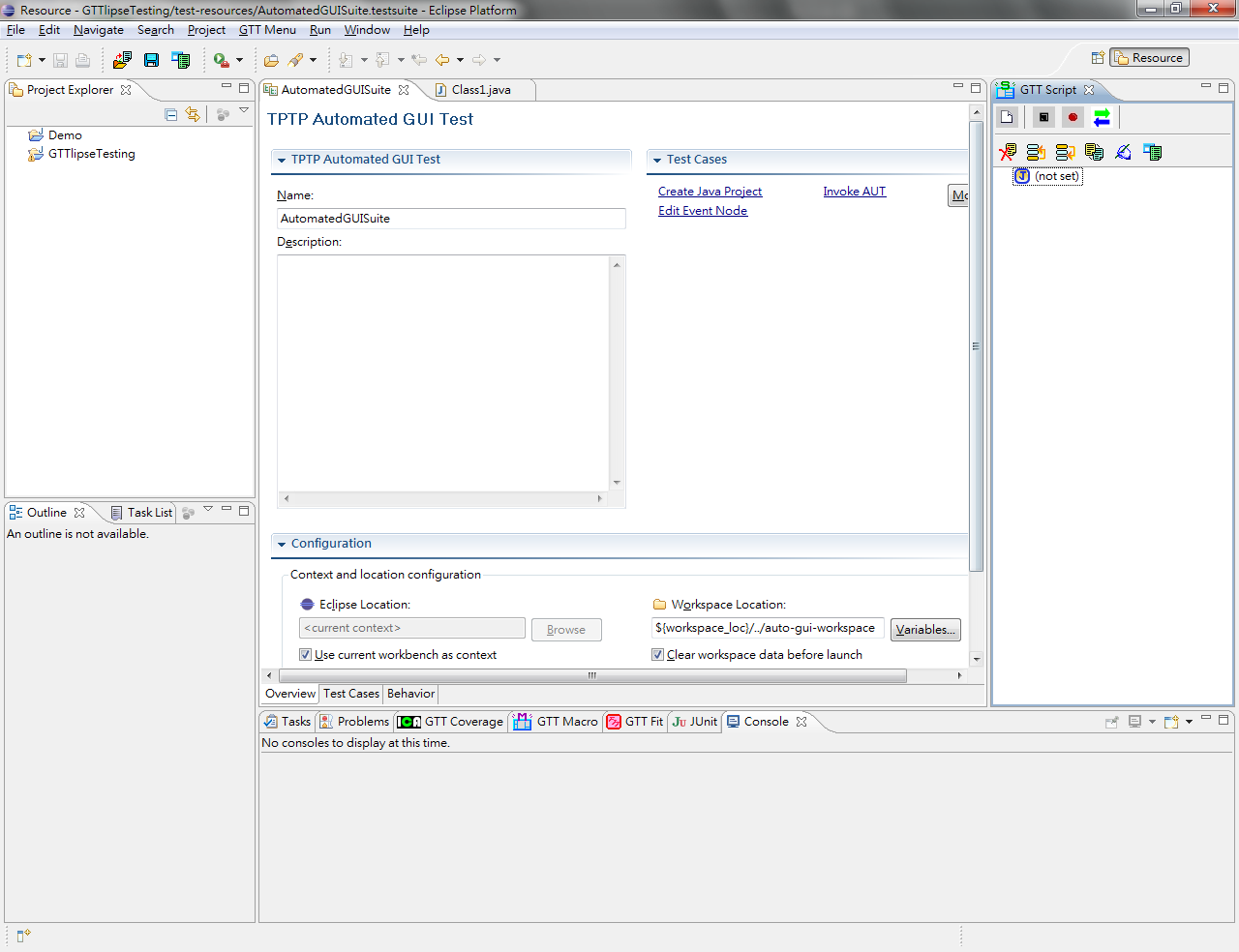
1. 設定 GTTlipse 和該專案關聯
   1. 選擇 Window => Open Perspective =>Other



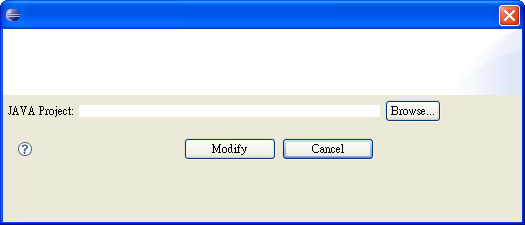
* 1. 選擇 GTTlipse 並按下 Ok 切換到 GTTlipse 視景



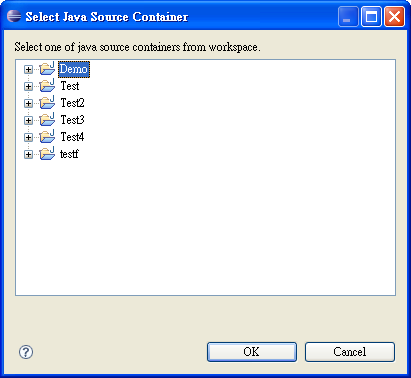
* 1. 接著可以在畫面的右上角看到 GTT Test Script Editor View，一開始只有一個顯示 “(not set)” 的 Node，點選該 Node 兩下，可選擇要和哪個專案進行關聯。



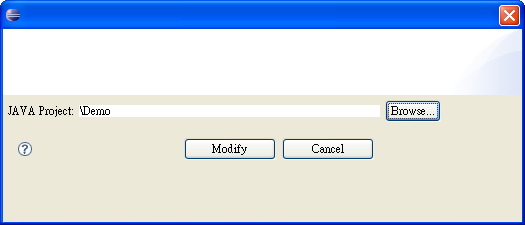
* 1. 選擇 Browse 以進行專案的選擇。



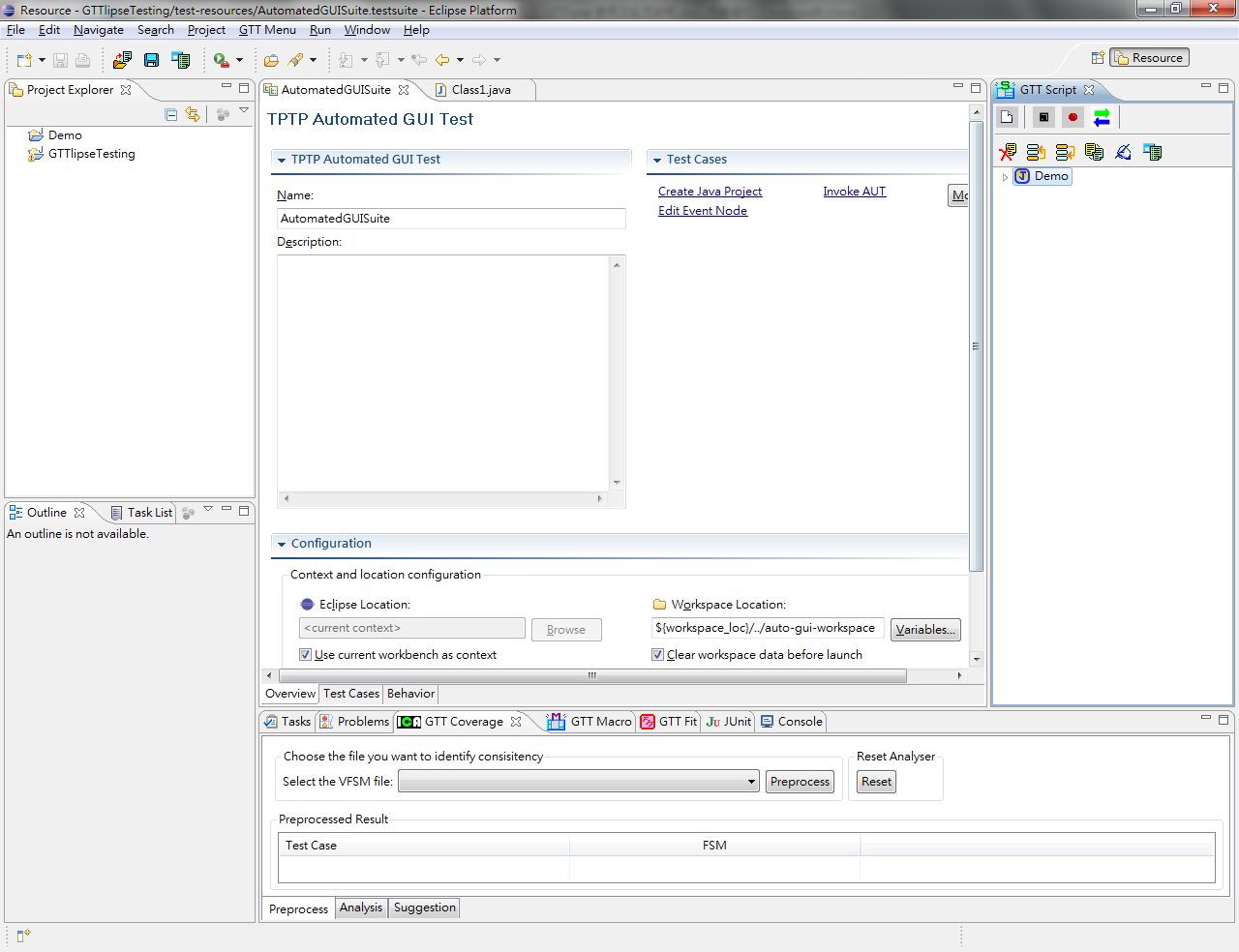
* 1. 選擇 Demo 專案並按下 OK



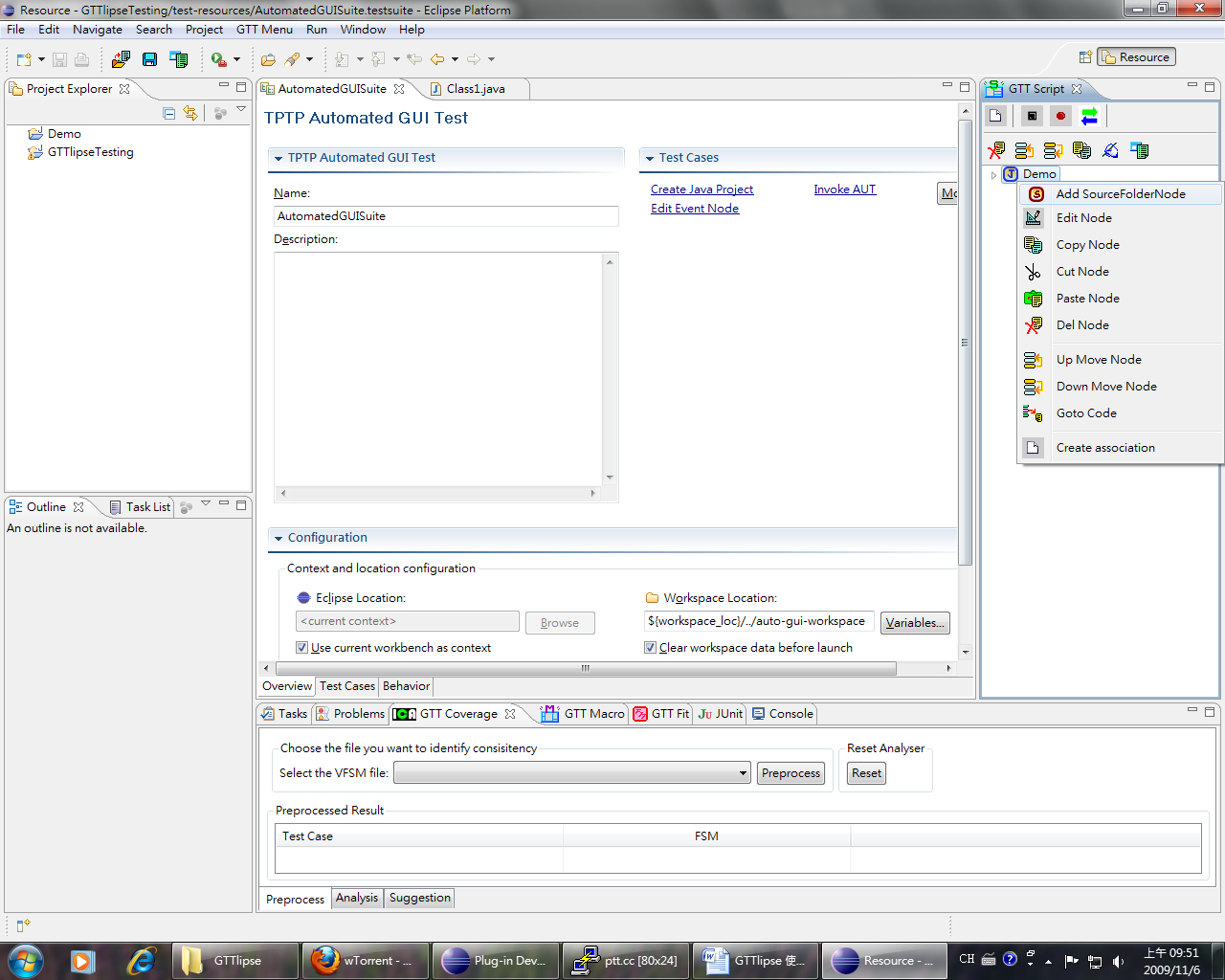
* 1. 按下 Modify 表示與該專案進行關聯，GTTlipse 會掃描該專案目前已有的檔案並建好表示專案檔案路徑的樹狀結構；若是之前已經使用 GTTlipse 編輯過的專案則會載入 Java Project 目錄下的 GTTlipse.gtt 檔案，該檔案除了會紀錄專案的目錄結構外還會額外紀錄 GUI 測試的 Script 資訊；GTTlipse 除了建立專案的目錄結構外還會一併載入 GTTlipse.gtt 中關於 GUI 測試的資訊，並設定進行 GUI 測試時所需的 Jar 檔位置以及其他相關檔案的位置。



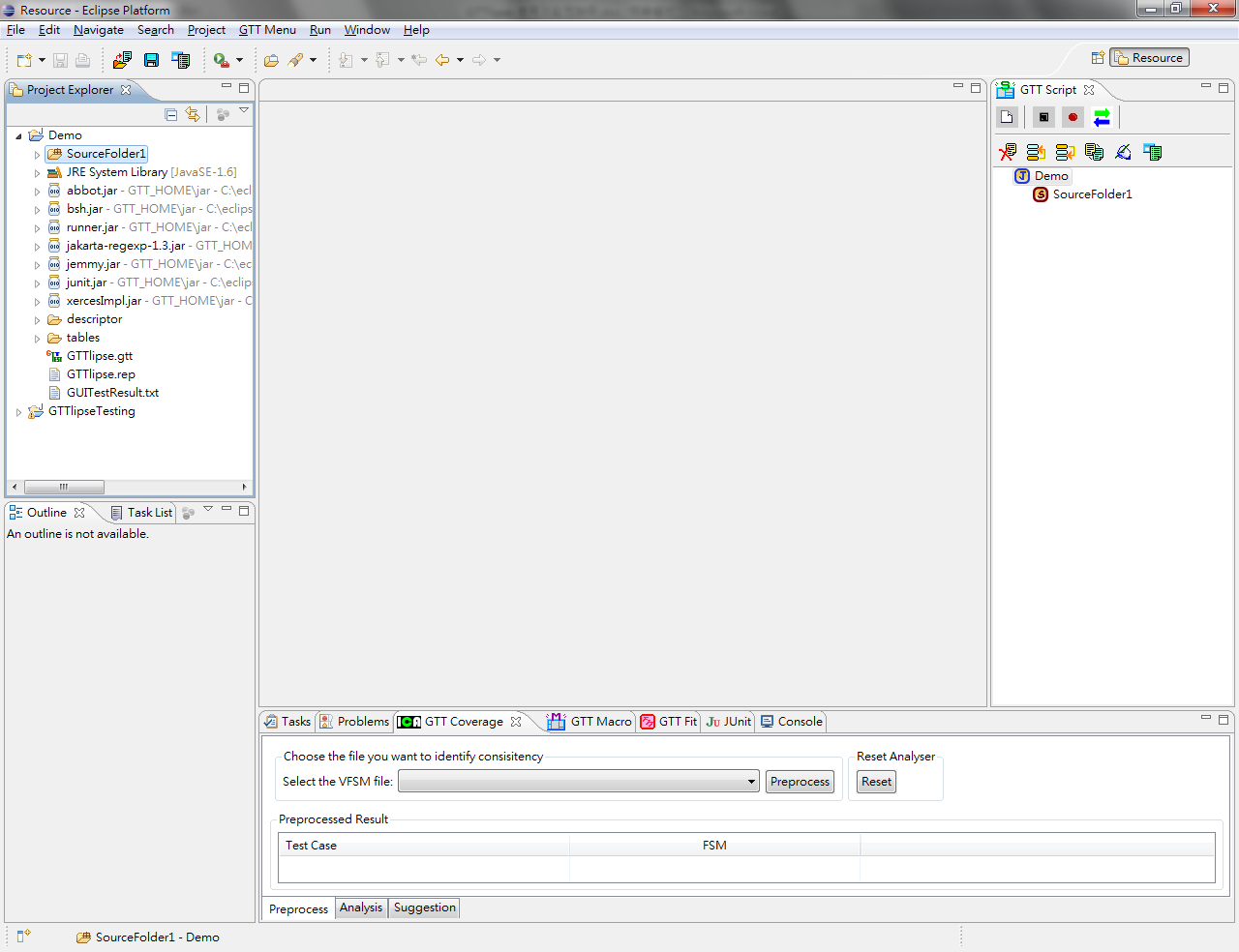
* 1. 完成關聯設定後該 Node 會顯示專案的名稱表示目前是和該專案關聯



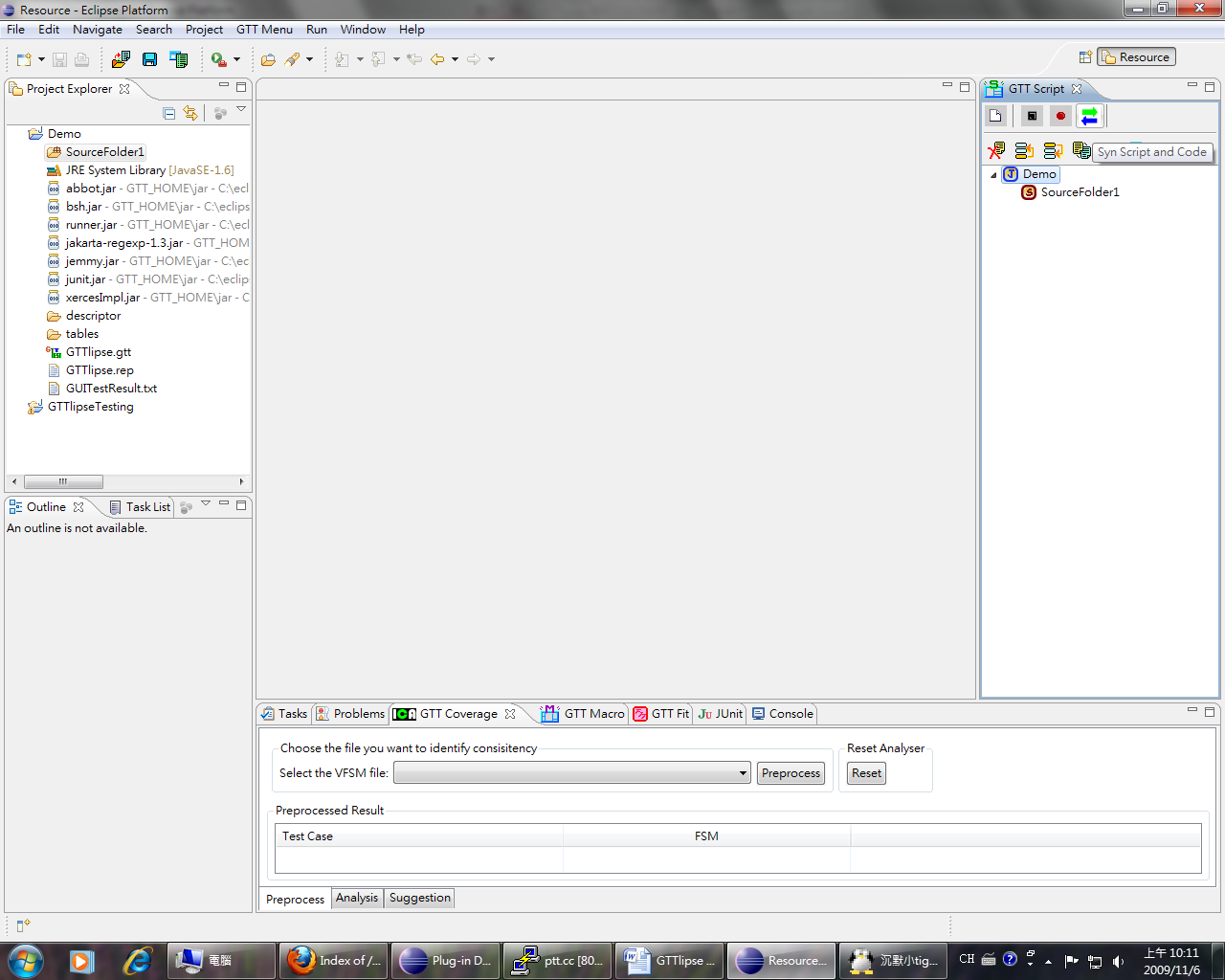
1. 建立 GUI 測試程式
   1. 對該 Node 按滑鼠右鍵選擇 Add SourceFolderNode以建立 Source Folder



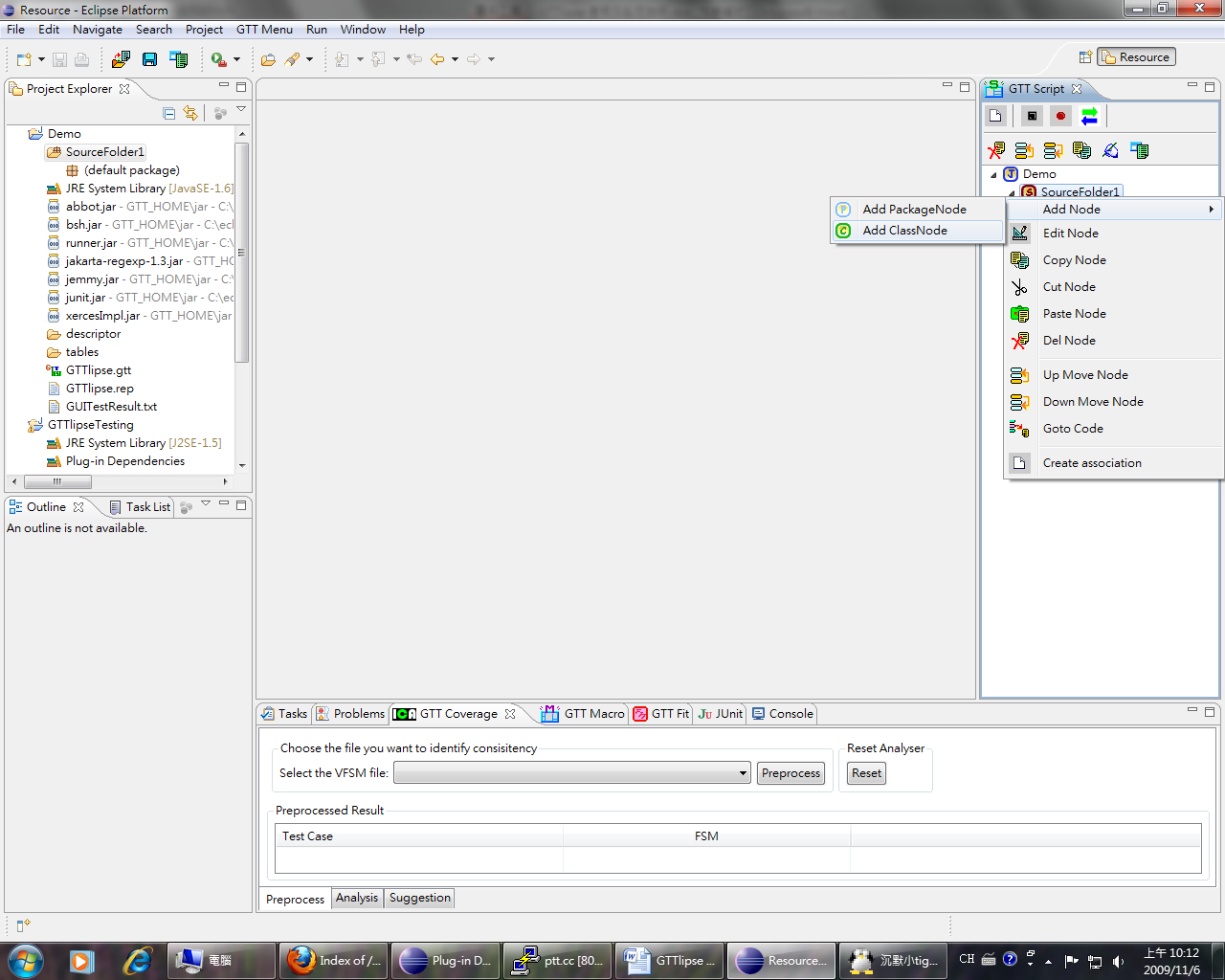
* 1. 完成後可發現 Demo 下面多了個子節點，並且在左邊的Java Project下也同時加入了一個 SourceFolder



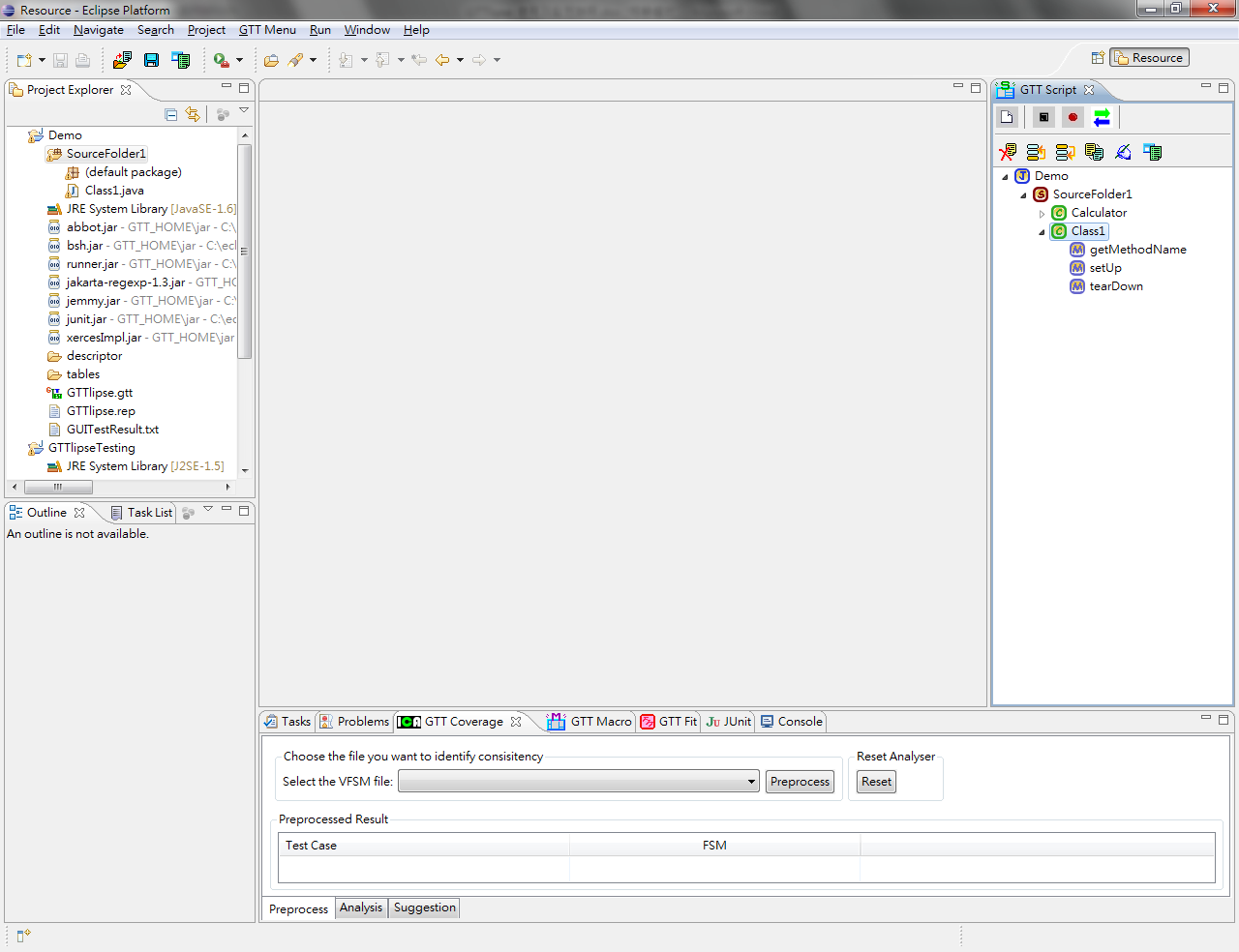
* 1. 接著開始撰寫 AUT 的 SourceCode，或者在原本的專案中載入之前寫好的 AUT 程式，假設寫好了一個小算盤的程式，並計畫對該程式進行 GUI 測試。
  2. 當撰寫程式時 GTT Test Script Editor View 也會加入與其相對應的 Node，如果發生 Script 和 Code 不對應的情況可以按下強制同步，使 Script和 Code 同步。



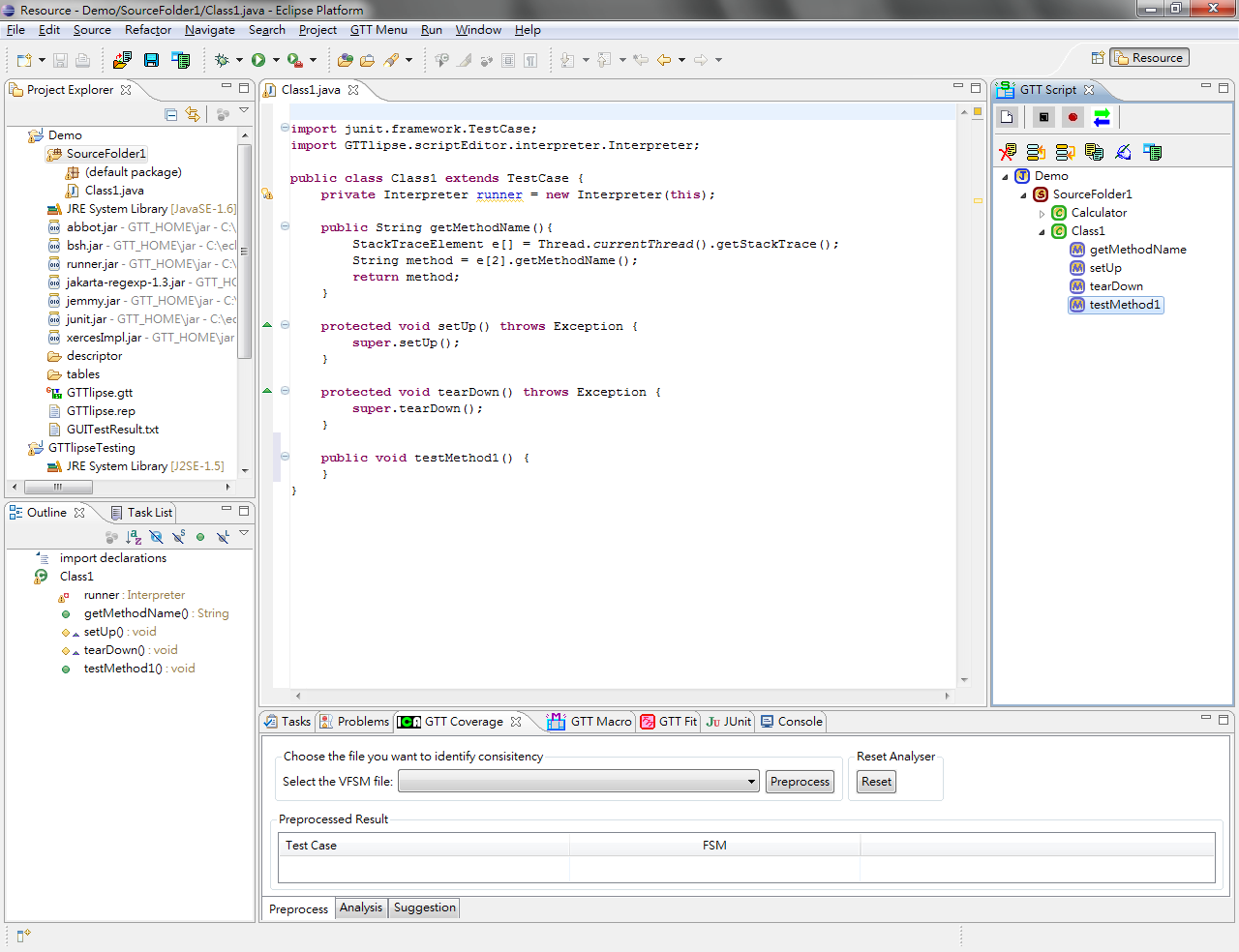
* 1. 接著在欲加入 GUI 測試程式的 Package 或是，Source Folder 上按滑鼠右鍵選擇 Add Node=>Add ClassNode 以加入 GUI 測試的程式



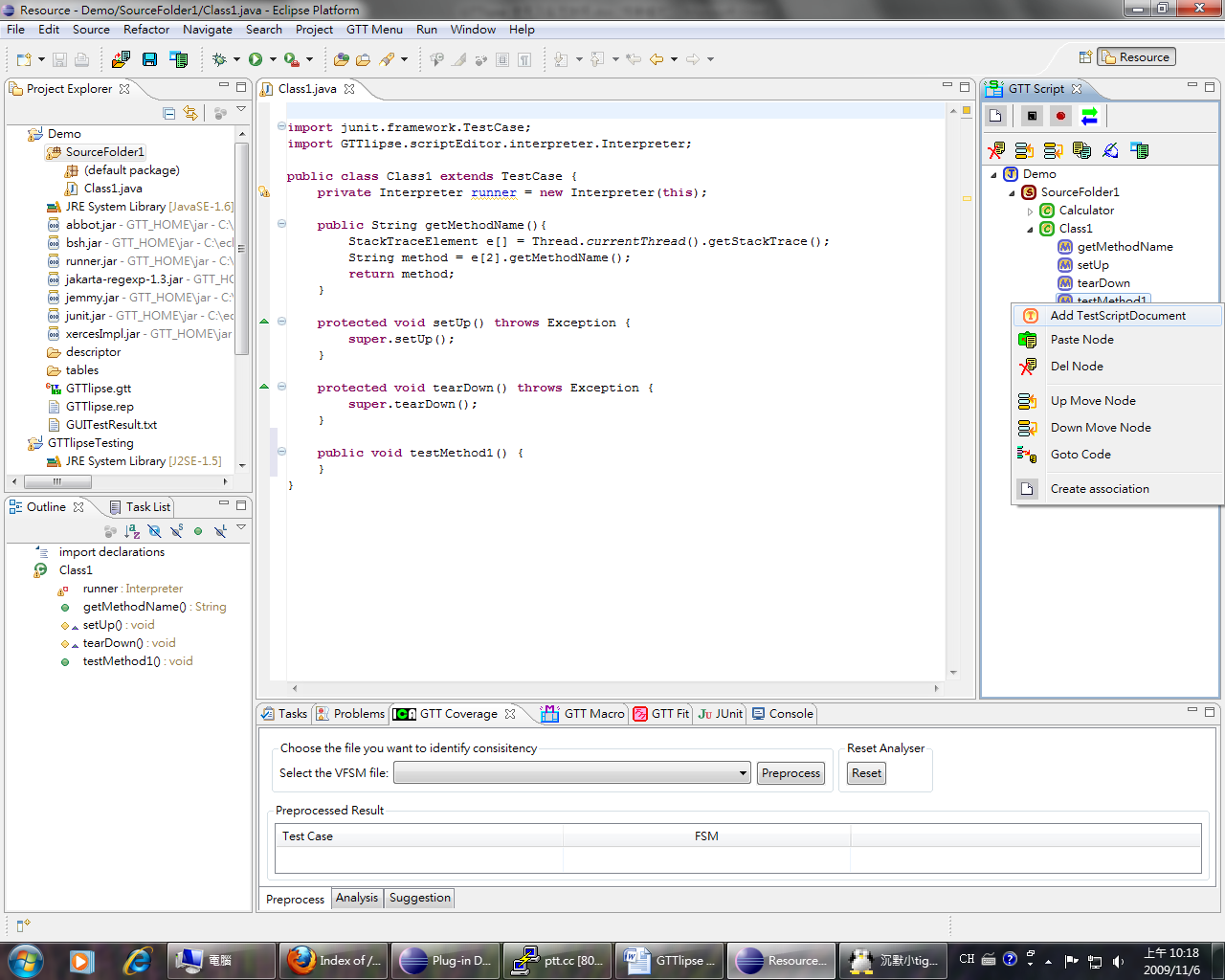
* 1. 該檔案會預先加入 setup、tearDown和getMethodName 三個 Method



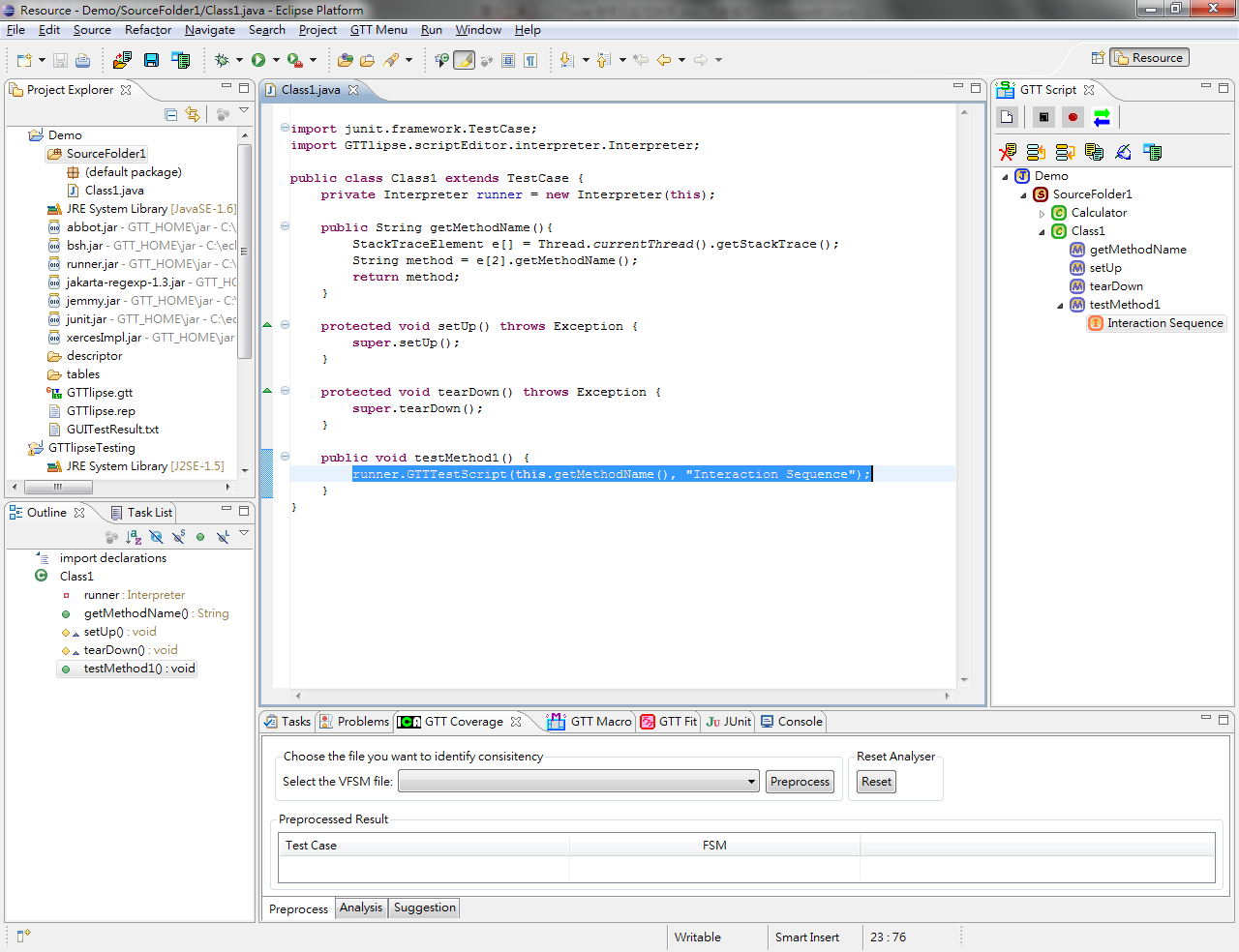
* 1. 接著對該檔案按右鍵加入一個新的 test method



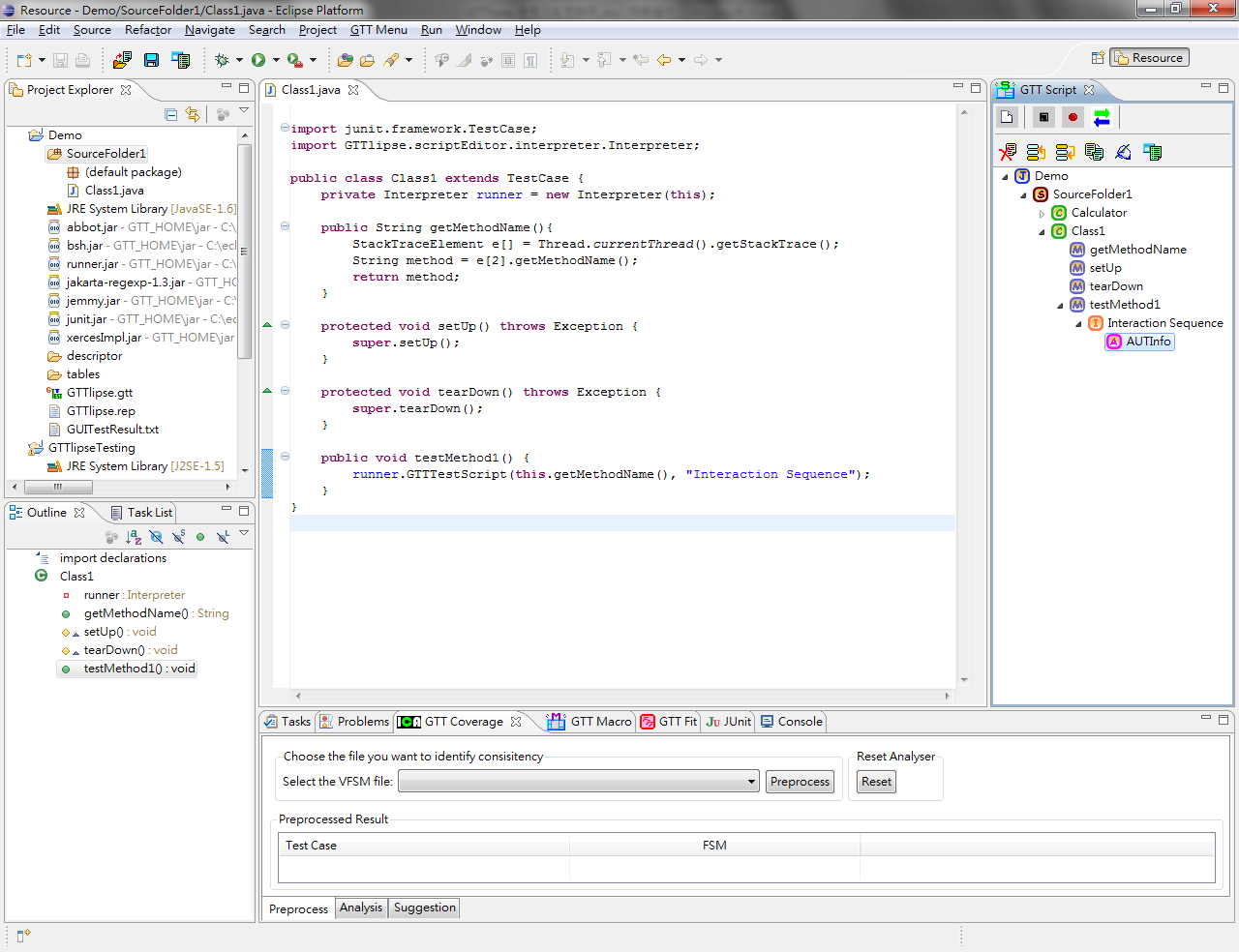
* 1. 接著在對該 test method 按右鍵加入一個新的 Interaction Sequence



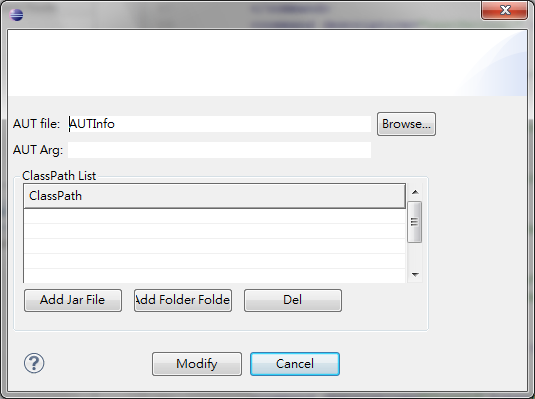
* 1. 完成後可以看到在 test method node 下增加了一個 Interaction Sequence，而在檔案的原始碼中則增加了一行 GUI 測試的程式碼。



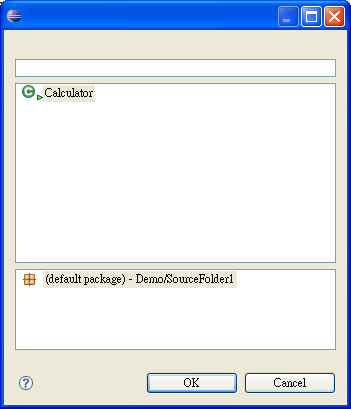
1. 編輯 GUI Test Script
   1. 首先在 Interaction Sequence 按右鍵選擇加入 AUTInfoNode ，以載入要進行測試的 AUT。



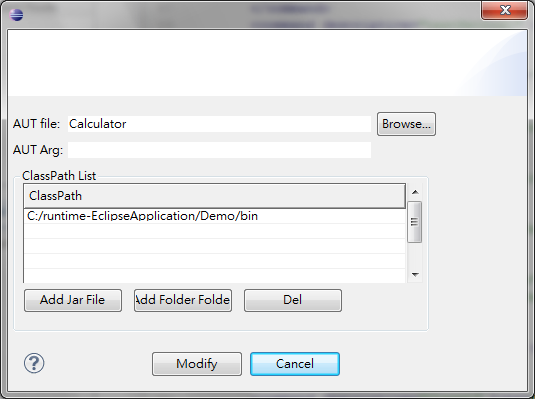
* 1. 對AUTInfoNode 進行Double Click 的動作以開啟 AUT 的設定視窗，點選 Browse 選擇欲載入的 AUT



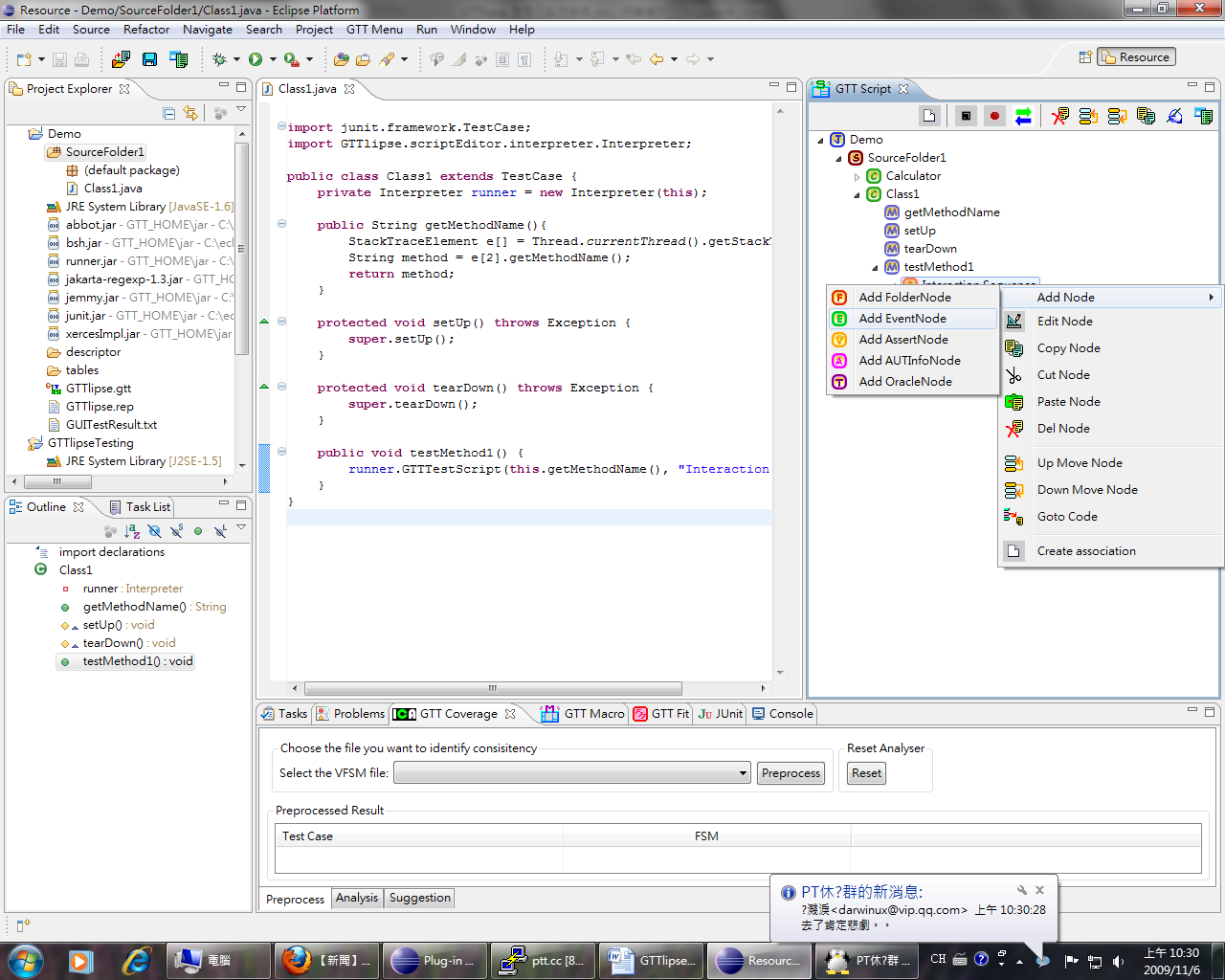
* 1. 選擇欲載入的 AUT 並按下 OK



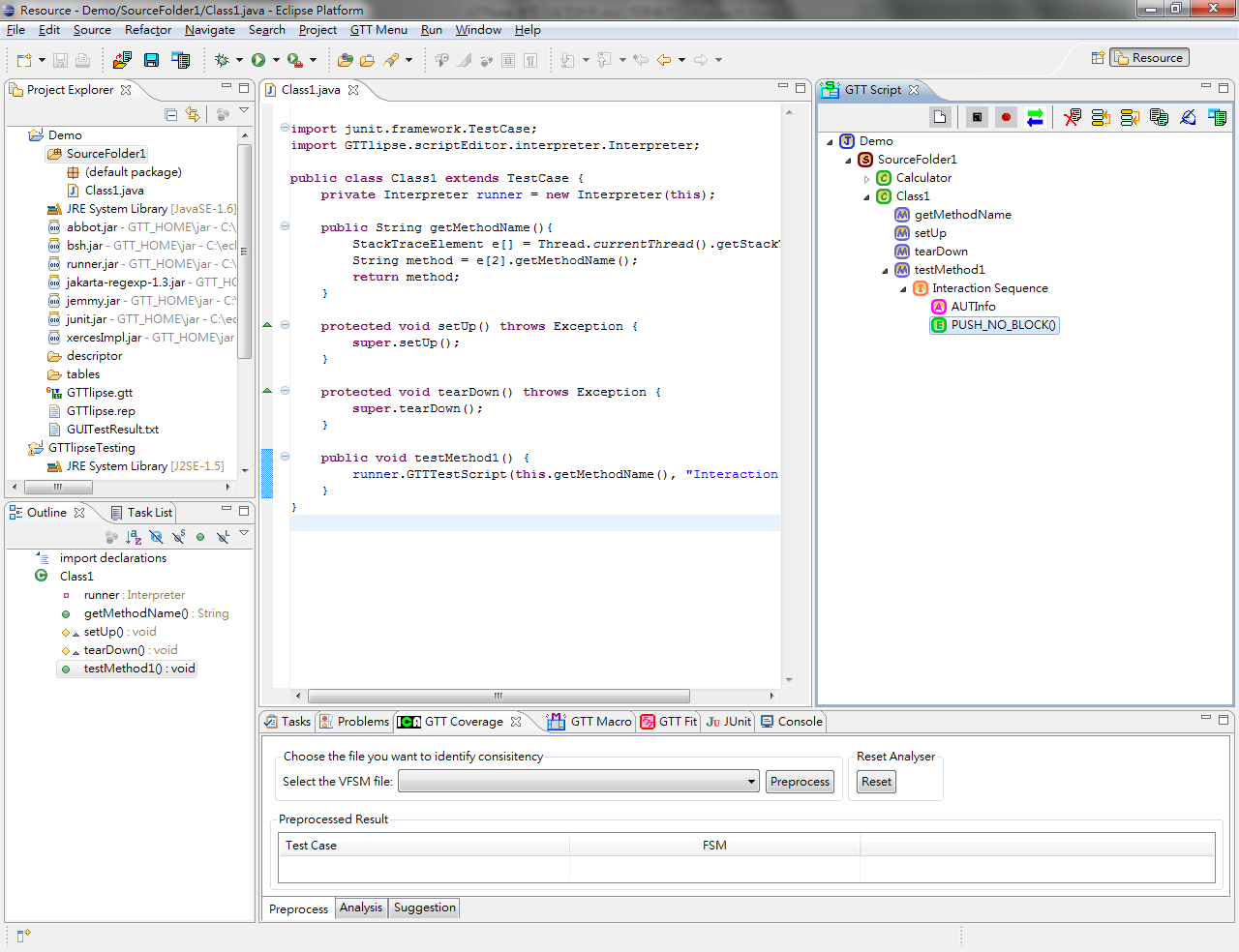
* 1. 接著按下 Modify



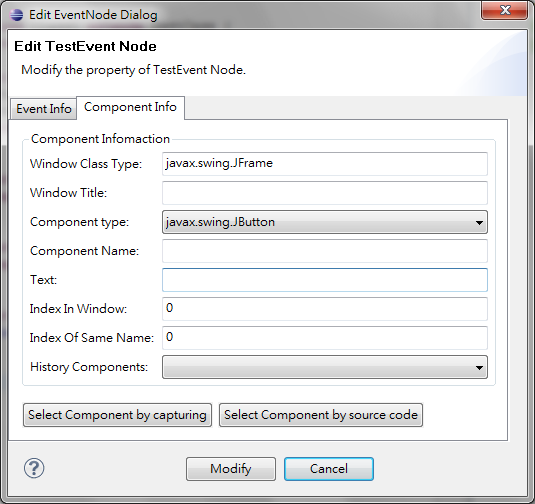
* 1. 接著加入 Event Node



* 1. 對 EventNode 進行 Double Click 的動作



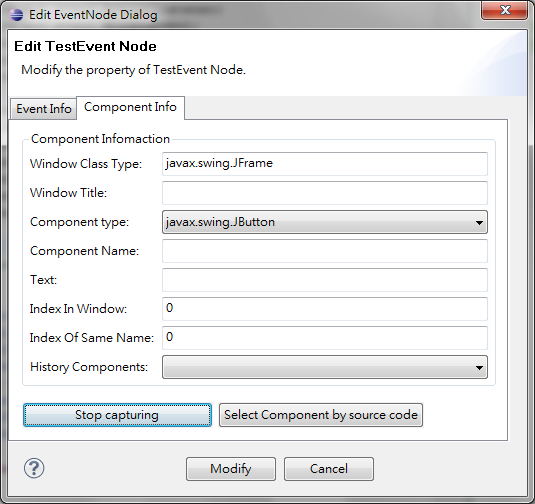
* 1. 接著在編輯 Event Node 的視窗中，設定要對特定的 Component 進行何種 Event 的操作，在設定 Component 資訊時，除了可以用手動輸入外，另外也提供了三種輔助輸入 Component 資訊︰（1）使用點選 AUT 來輔助輸入 Component 資訊，（2）由 Source Code 中取得 Component 的資訊，（3）由歷史紀錄中選取之前輸入的 Component 資訊。



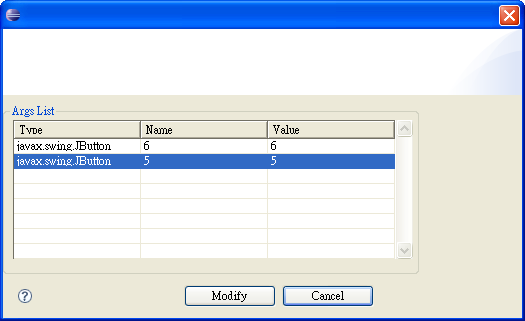
* 1. 假設使用由 AUT 取得 Component 資訊的功能，點選 Select Component by capturing
  2. 在 AUT 上點選欲對其進行操作的 Component



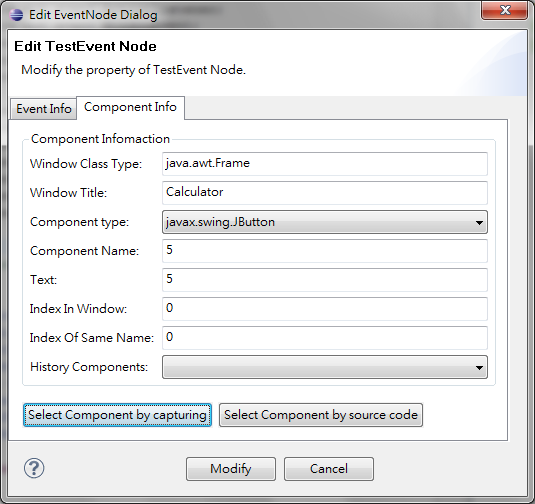
* 1. 點選 Stop capturing



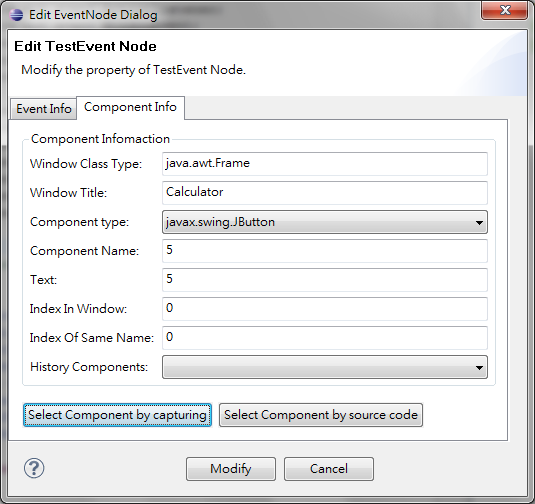
* 1. 接著 GTTlipse 會列出使用者所點選的 Component 供使用者選取，選擇 Component 並按下 Modify



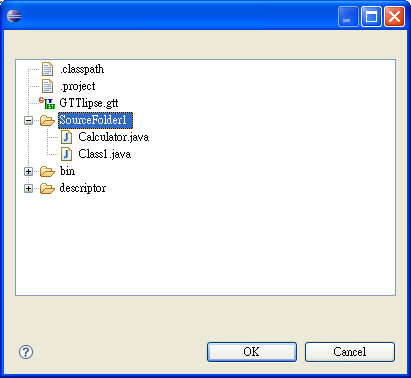
* 1. GTTlipse 會依照使用者所選的 Component 填入 Component Info編輯面板的欄位中。



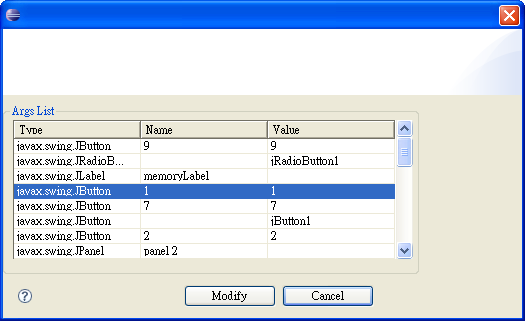
* 1. 同時也可以由 Source Code 取出 Component 的資訊點選 Select component by source code



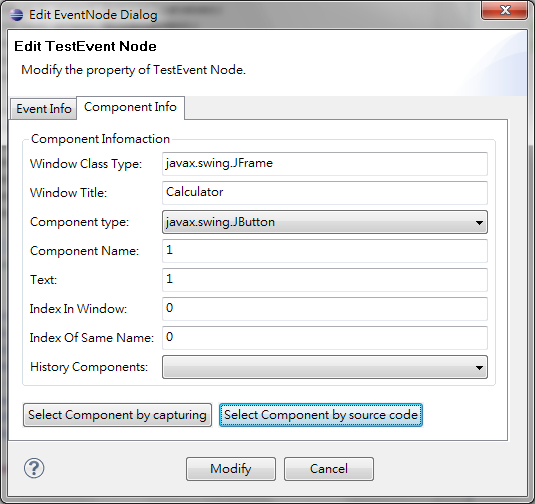
* 1. 點選 Source Code 所在的資料夾或是檔案，按住 Ctrl 可點選複數個檔案或資料夾，GTTlipse 會由這些使用者所選的檔案中分析出可能是 Componet 的資訊，選好檔案後按下 OK 進行檔案分析。



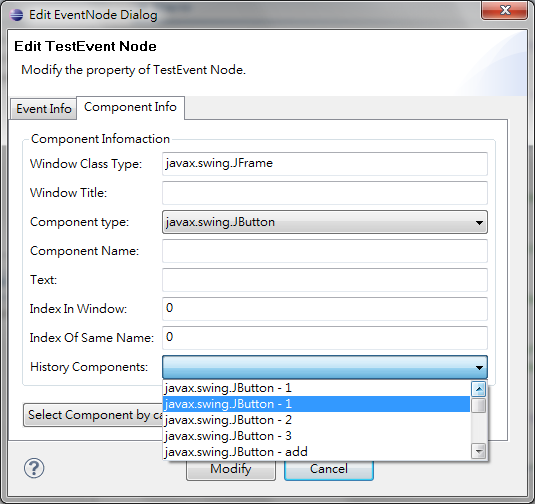
* 1. 接著點選 Component 並按下 Modify



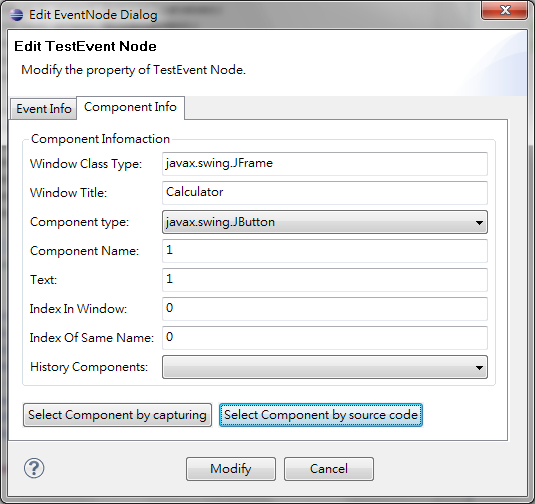
* 1. GTTlipse 會依照使用者所選的 Component 填入 Component Info編輯面板的欄位中。



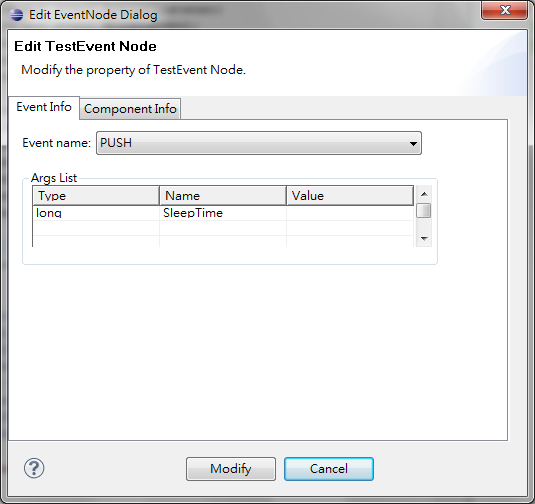
* 1. 若之前已經有輸入過相同或是類似的 Component 資訊也可以由 History Component 下拉選單中選取。



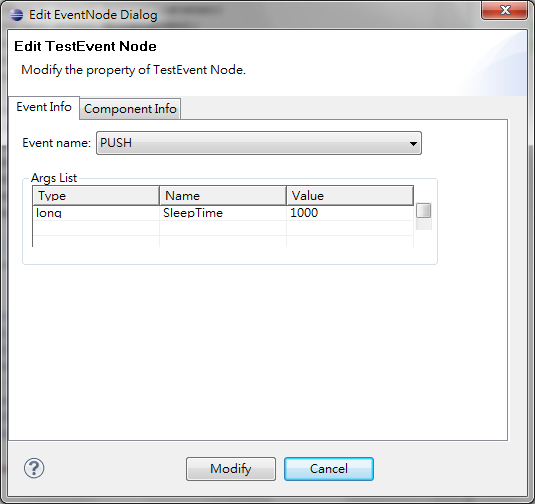
* 1. GTTlipse 會依照使用者所選的 Component 填入 Component Info編輯面板的欄位中。



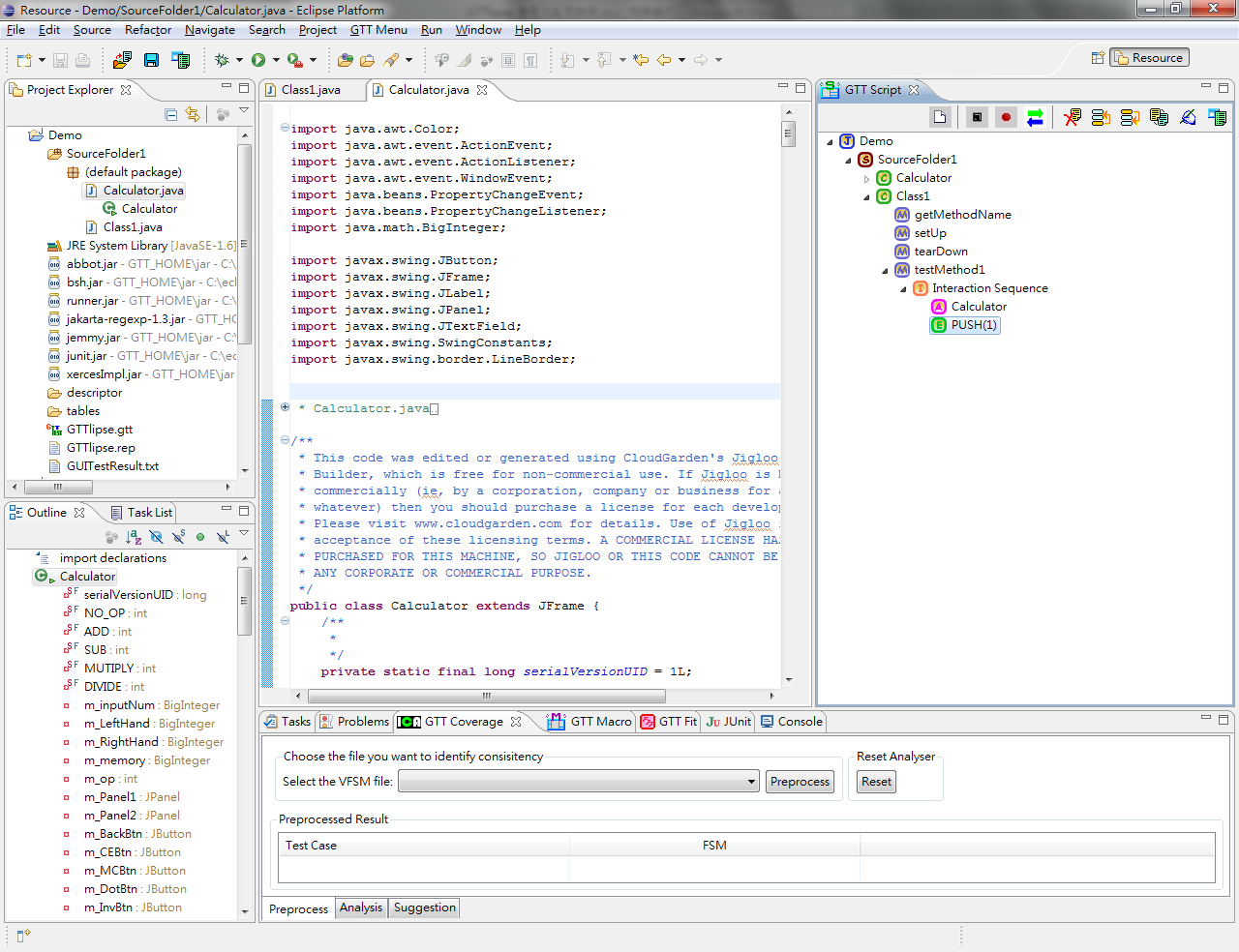
* 1. 接著點選 Event Info 切換到設定 Event 的畫面，在畫面中可以選擇要對該 Component 發的事件。



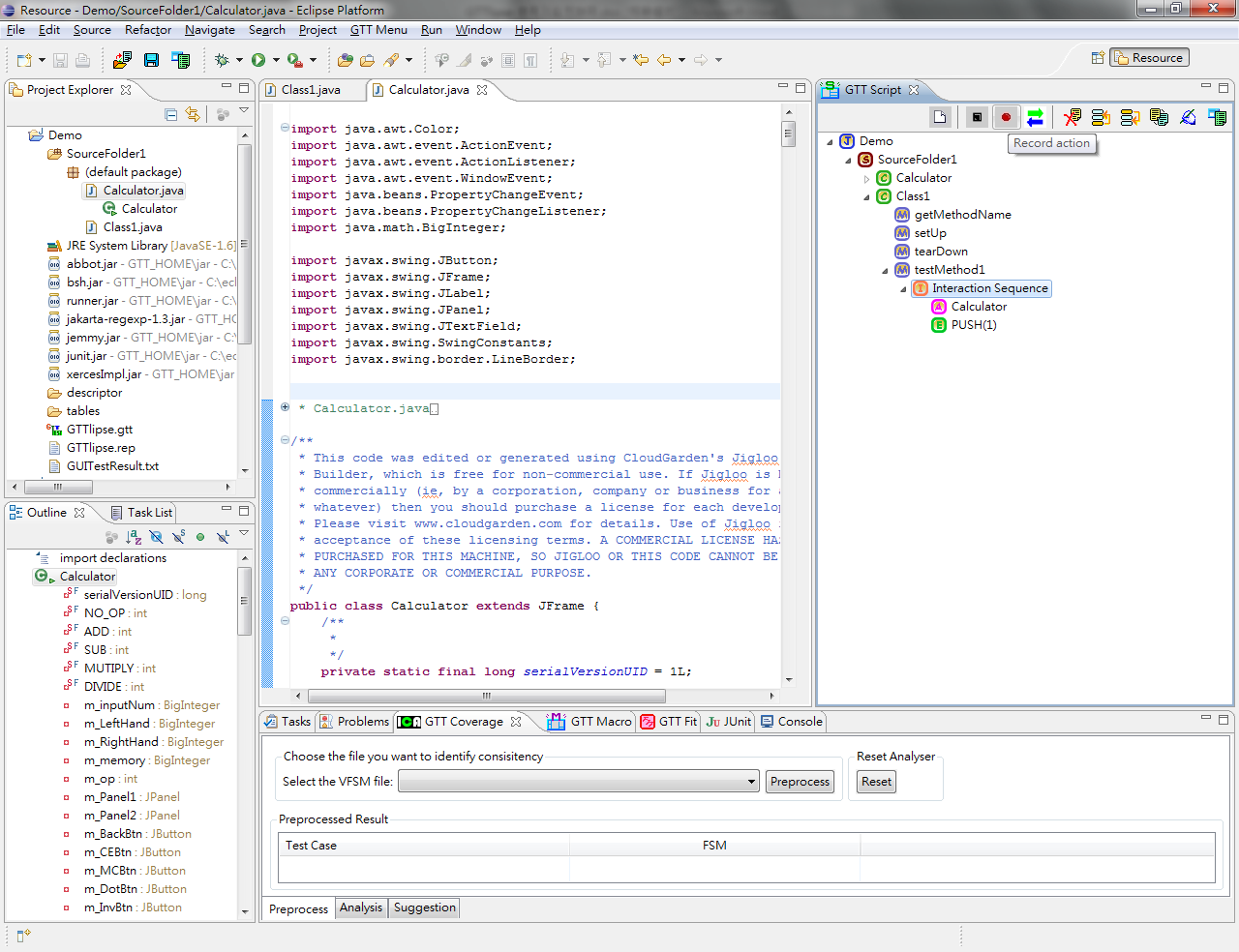
* 1. 接著設定該 Event 所需要的參數



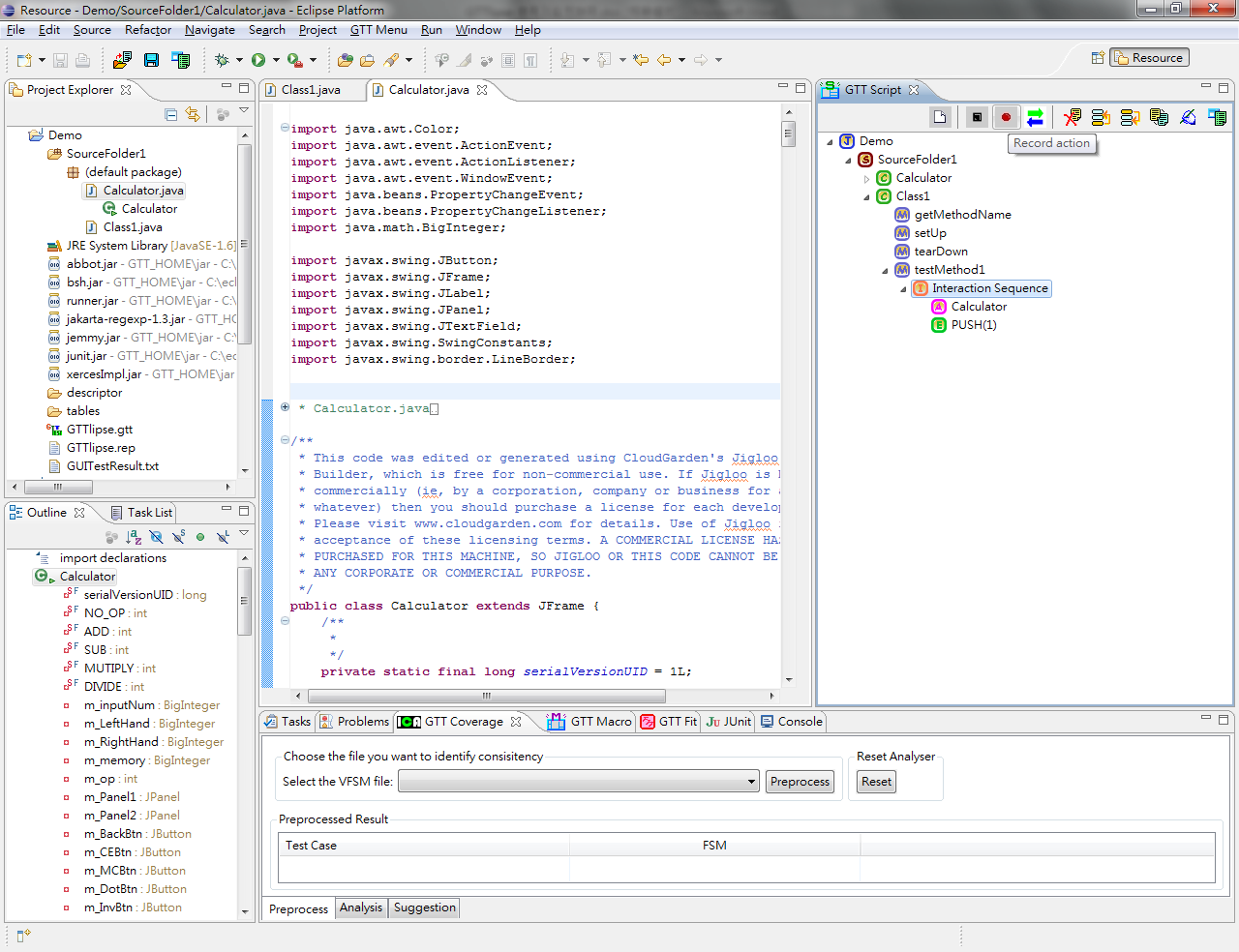
* 1. 最後按下 Modify 關閉 Event Node 設定視窗，如此一來便完成 Event Node 的設定。



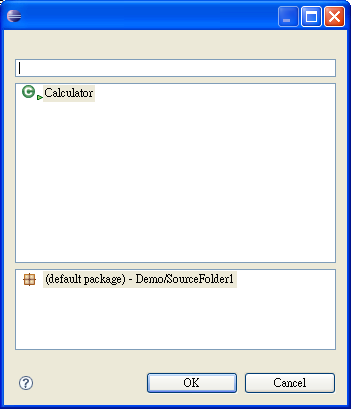
1. 使用錄影方式產生 Script
   1. 除了可以用手動的方式進行 Test Script 的編輯外也可以使用錄影的方式進行 Test Script 的編輯，首先點選Interaction Sequence，或是某個 Folder Node。



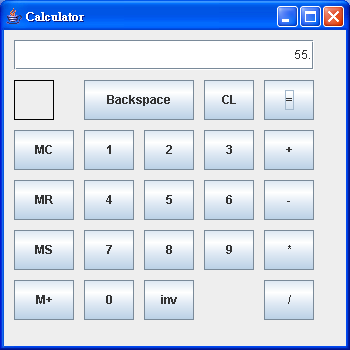
* 1. 接著點選 tool bar 上面的 Record Action



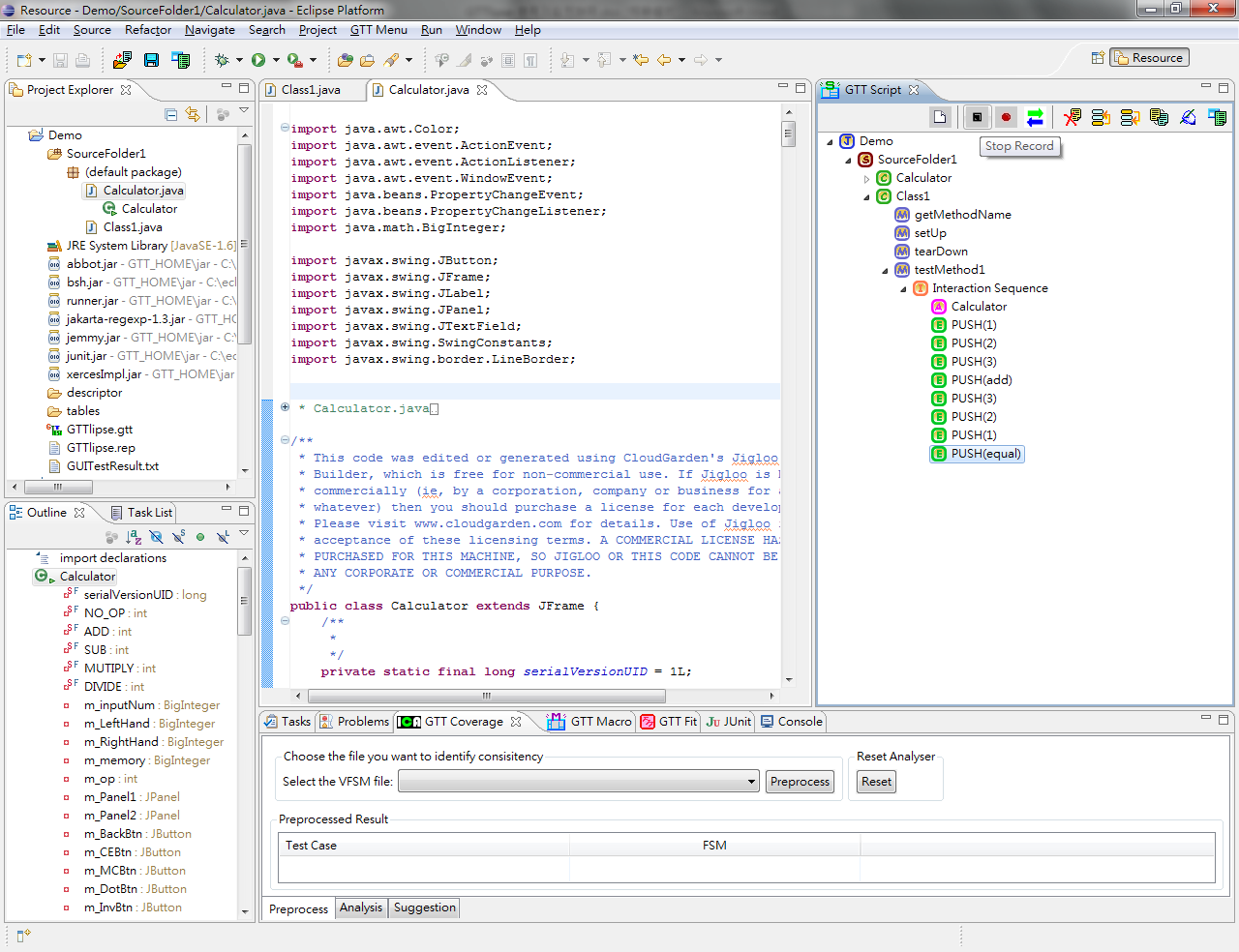
* 1. 點選欲測試的 AUT Class 並按下 OK



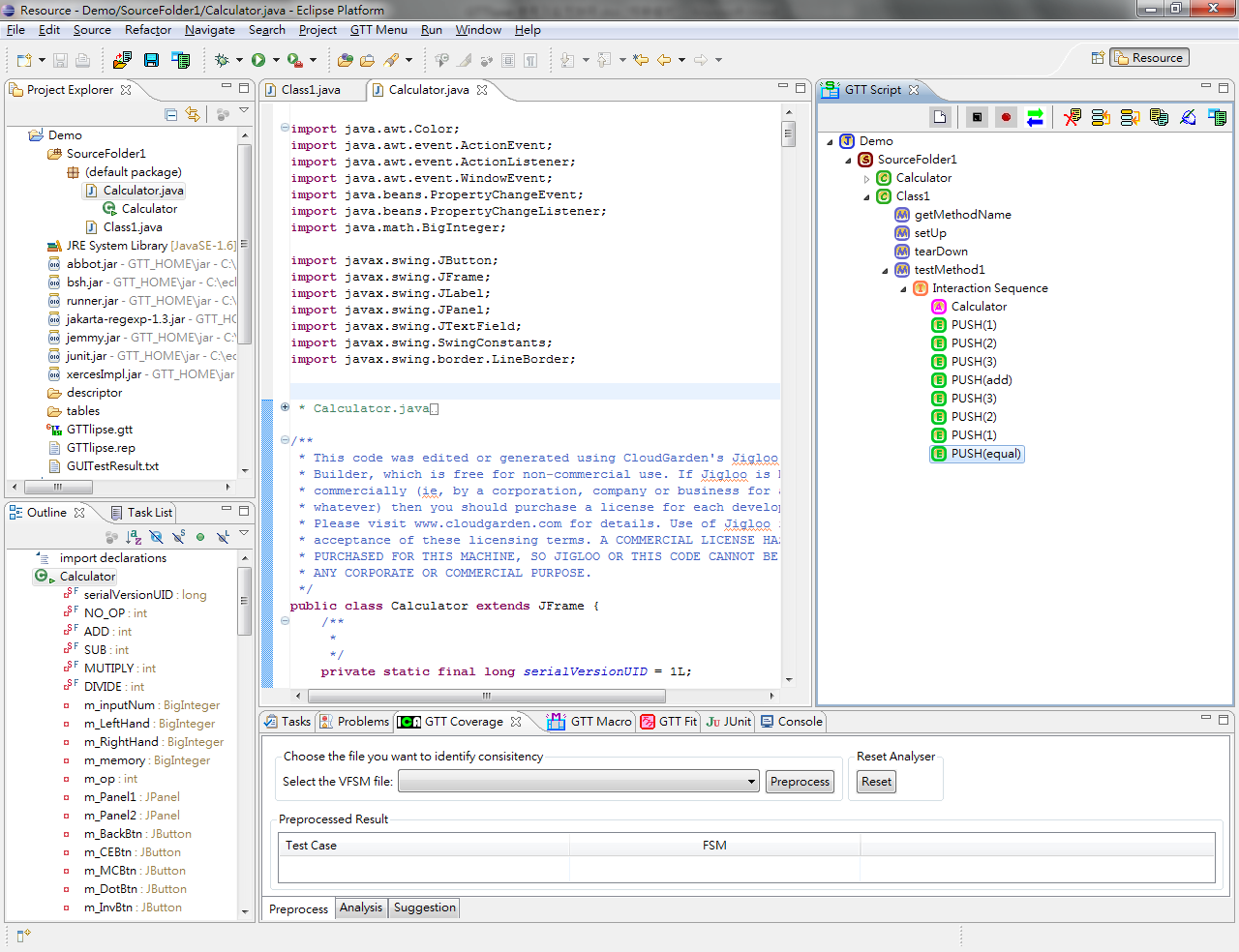
* 1. 在 AUT 上進操作



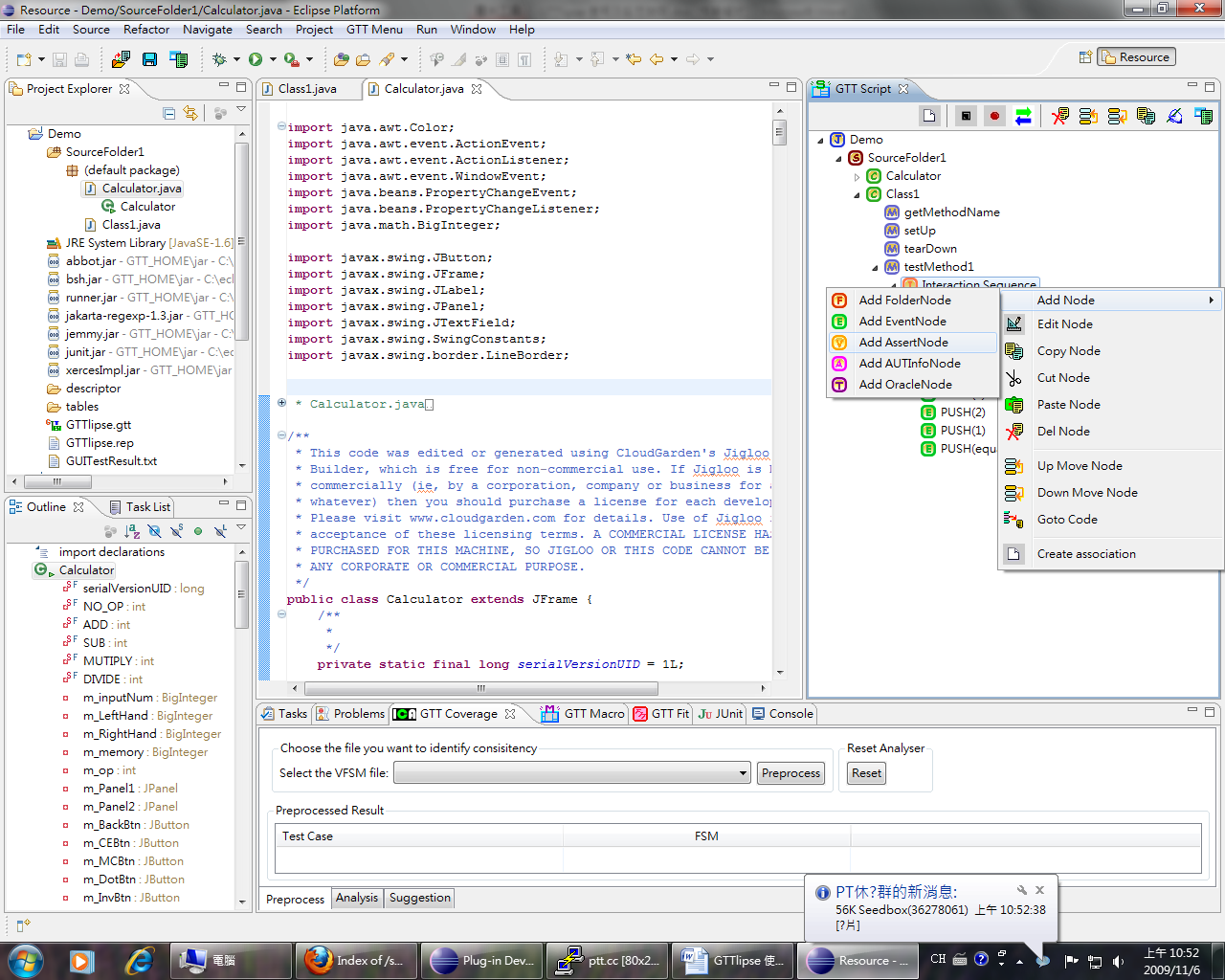
* 1. 按下 tool bar 上的 Stop Action 結束錄影



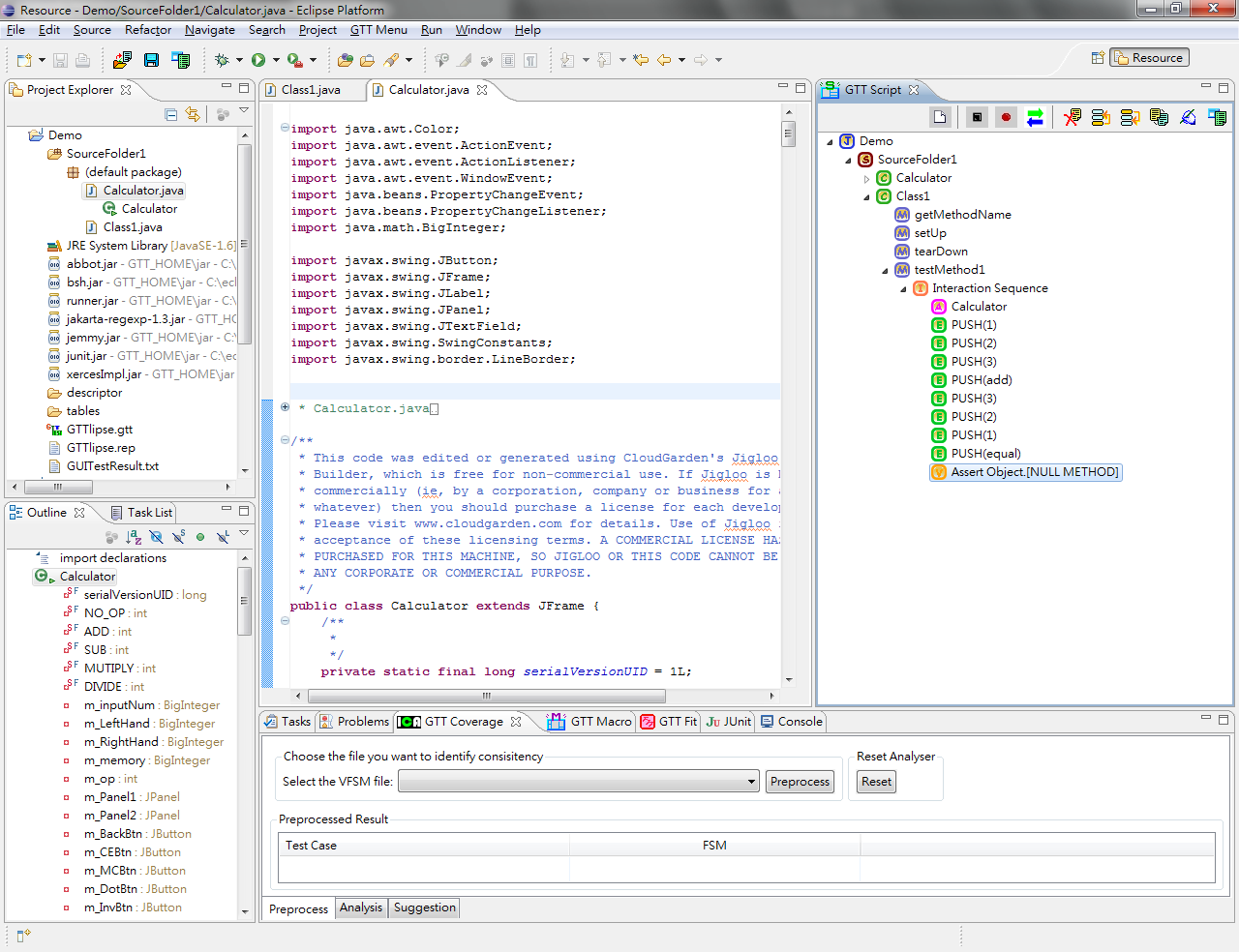
* 1. 則使用者的操作會轉換為一連串的 Event Node 並增加在 Interaction Sequence 下。



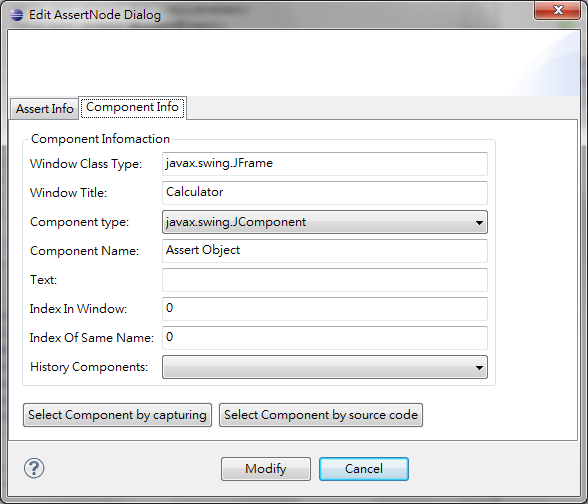
1. 插入驗證節點
   1. 點選 Interaction Sequence 並按下滑鼠右鍵選擇插入 Assert Node



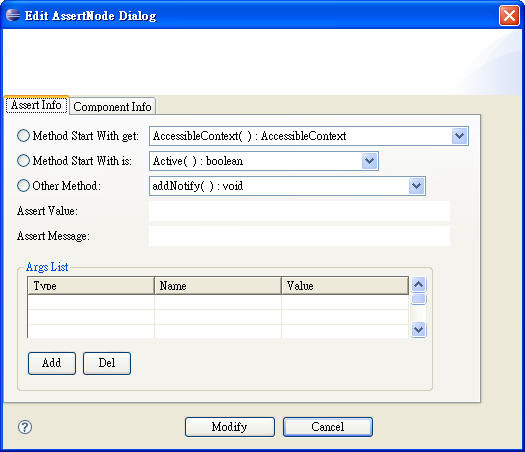
* 1. 對 Assert Node 進行 Double Click 的動作



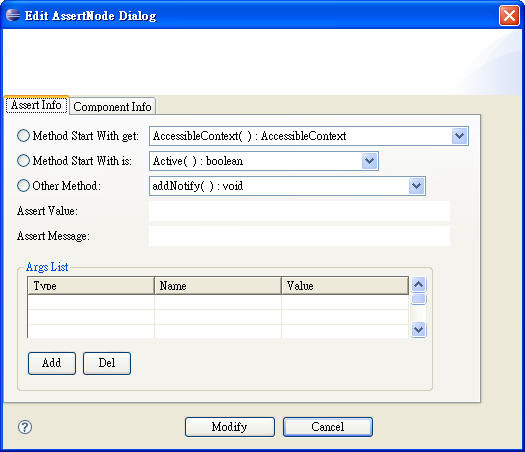
* 1. 首先如同編輯 Event Node 需要編輯 Component 和 Event 資訊一樣，Assert Node 也需要編輯 Component 和 Assert 資訊，Component 資訊的部分和 Event Node 完全相同故不多加贅述。



* 1. 在編輯 Assert Info 的部分，需要選擇透過哪個 Method 取得 Assert 所需的資訊，並輸入 Assert Value ，以及錯誤時所要顯示的 Assert Message，如果 Assert Method 需要傳入參數的話需額外設定。



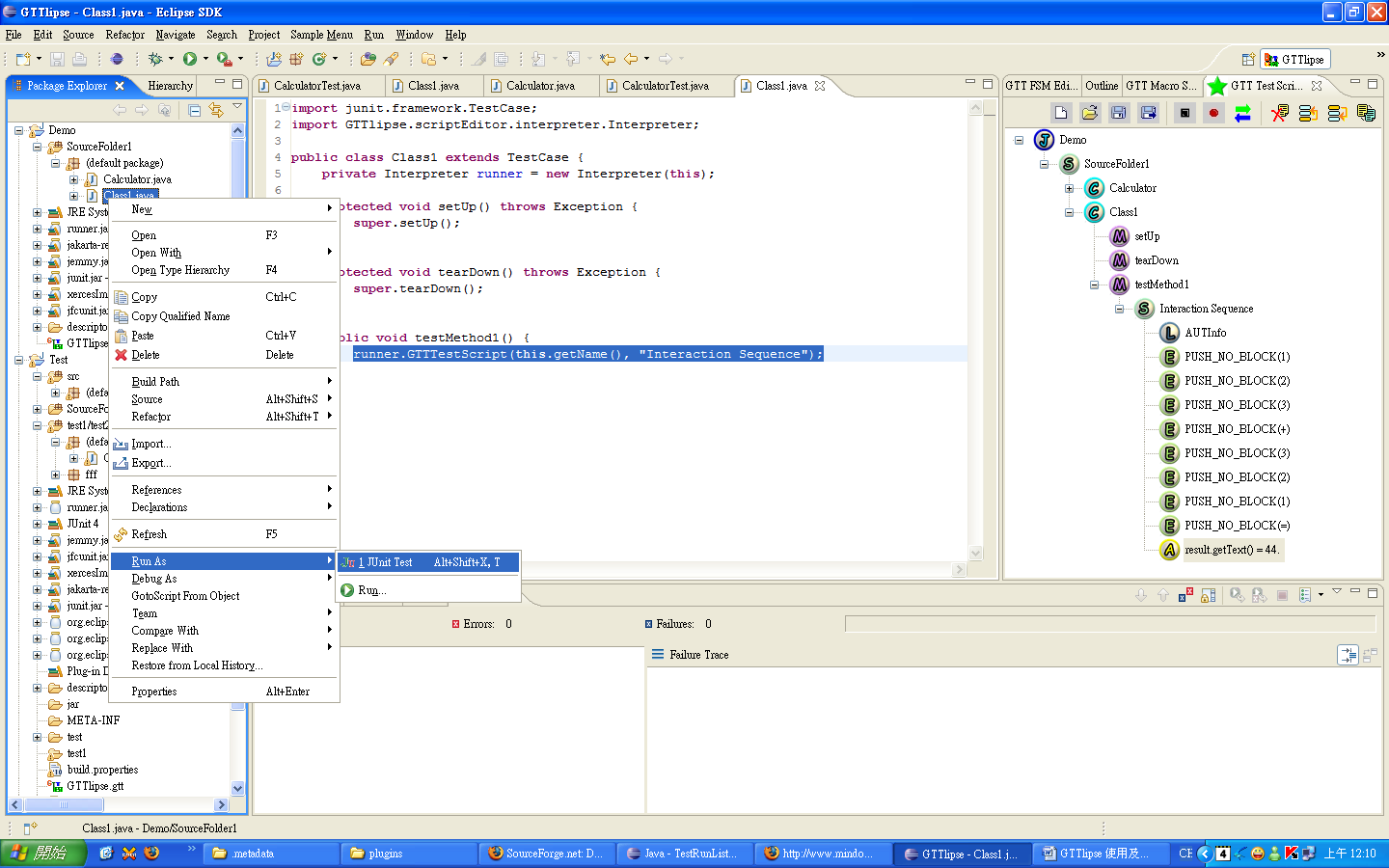
* 1. 設定完成後按下 Modify 即可關閉設定視窗，如此便完成插入 Assert Node 的動作。



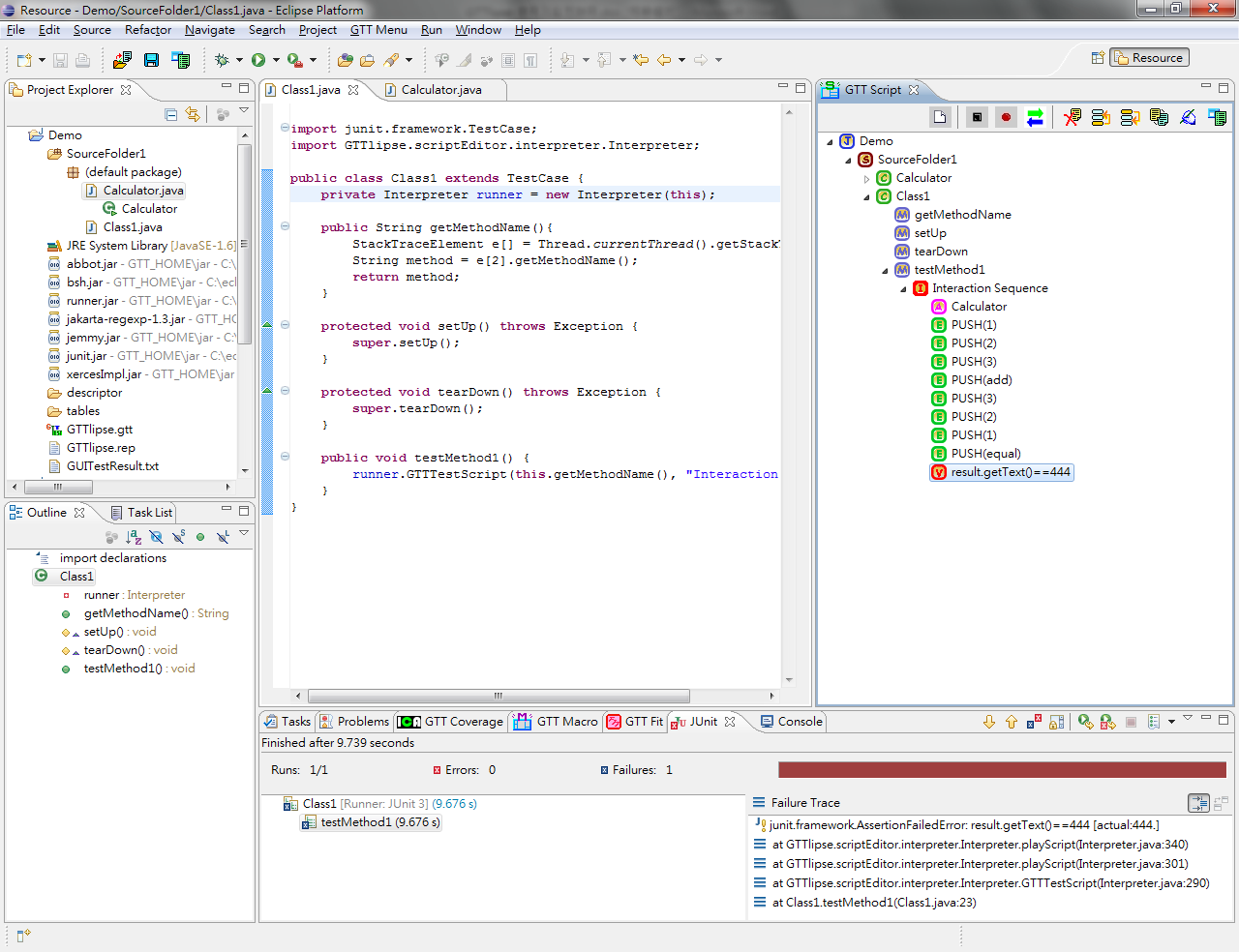
1. 執行測試
   1. 當編輯好 Test Script 後，按下 tool bar 上的 Save GTT Script File 即可將 Script 進行儲存。



* 1. 接著在左邊的 Package Explorer 中點選測試程式，並按下滑鼠右鍵選擇 Run As=>Junit Test 即可進行測試。



* 1. 若測試結果和預期的值不相等則會在JUnit View顯示錯誤訊息，並在 Test Script View 上將錯誤的節點以紅色顯示。



* 1. 若測試通過則顯示完成測試的畫面

