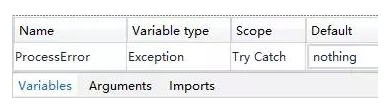
在眾多的程式語言中，開發程序都離不開Try Catch，UiPath中也有Try Catch。  
程序框架中依靠Try Catch來保障程序在運行出錯時，不僅可以獲知運行錯誤，還能確保其他程序正常運行。

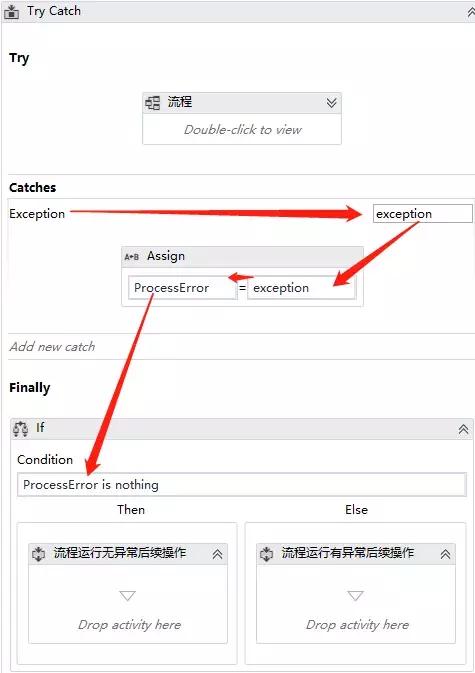
在今天的手把手教程系列，我們帶你詳細學習下Try Catch。

一 1

**用法（下面有截圖）**

1. 將開發好的流程放入Try Catch的Try中；
2. 在Try Catch範圍內，定義一個名為ProcessError的變量，類型是System.Exception；
3. 
4. 然後，在Catches中定義一個名為exception異常變量（用於在出現異常時，獲取異常信息），並通過Assign，將exception賦值給ProcessError；

最後，在Finally中放個if Activity，if Activity的Condition設置為：ProcessErroris nothing，then中放流程執行成功的操作，else中放執行失敗時要做的操作。

下圖是上面用法的Demo截圖，圖中用紅色箭頭標明了流程發生異常時，傳遞異常的路線：  
  
Try Catch

全稱：System.Activities.Statements.TryCatch

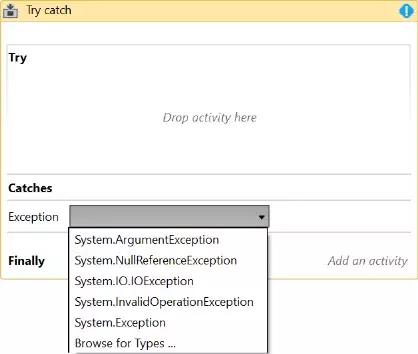
用途：捕獲你設計的流程中的指定異常類型，並顯示錯誤通知或將其解除並繼續執行。

外觀：

由三部分組成：

1.Try - 放編寫的流程（該流程有可能發生你未寫入流程邏輯的錯誤）。

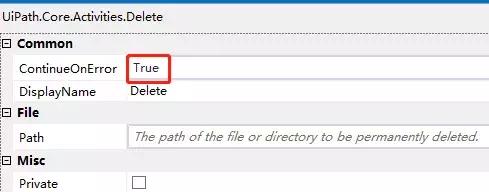
2.Catches - 發生錯誤時要執行的Activity或Activity集。異常 - 要查找的異常類型。請注意，您可以添加多個類型的異常。



3.Finally- Try Catch塊之後要執行的Activity或Activity集。只有在沒有拋出異常或發生錯誤並且在Catch部分中捕獲時，才會執行此部分。

注意：

如果Activities放在Try Catch的Try部分中，並且Continue On Error屬性的值為True，則在執行項目時不會捕獲任何錯誤；舉例如下：



屬性

DisplayName - Activity的顯示名稱。Private - 如果選中，則不再以Verbose級別記錄變量和參數的值。

技巧：按快捷鍵「Ctrl + T」，將所選的activity置於Try Catch的Try部分內。

為了更好地理解Try Catch的用法，我們創建了一個自動化程序，它從隨機名稱生成器網站收集多個名稱並將它們寫入Excel電子表格中。

點擊UiPath英文網站的這裡，可以下載初始工作流程。(運行會出程序異常)

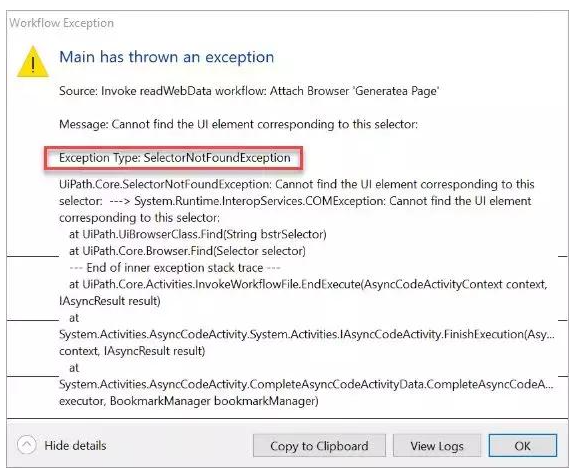
Build DataTable 創建一個用於存儲所收集名稱的表。

調用另一個工作流來讀取Web數據。

最後，Excel application scope用於在Excel文件中寫入收集的信息。

首先，讓我們運行自動化來檢查是否有任何錯誤。

請注意，將顯示「工作流異常」介面：

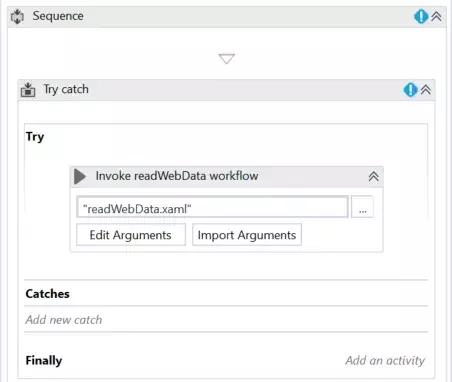


該異常類型欄位告訴我們是什麼問題，在Try Catch的Catches部分中，用作在工作流程執行期間查找的異常類型。  
正如您在上面的螢幕截圖中看到的，在運行示例工作流時，AttachBrowser容器選擇器似乎存在問題。問題是選擇器無法使用「生成隨機名稱 - 偽名稱生成器」名稱來標識瀏覽器窗口。

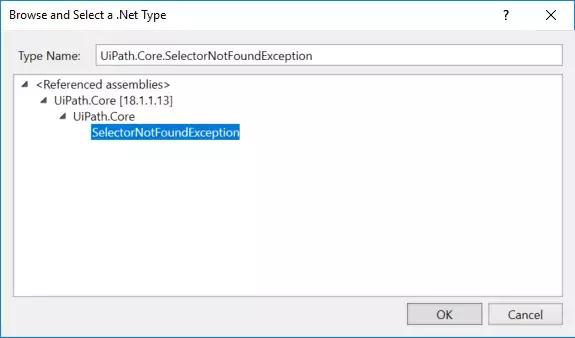
要捕獲這個異常，需要進行以下操作：

1.從Activities面板中拖一個「Try Catch」到流程編輯區。

2.在Try Catch的Try部分中放一個Invoke（引用了一個文件叫readWebData.xaml）。Try會監視Invoke，當它出現異常時進行捕捉

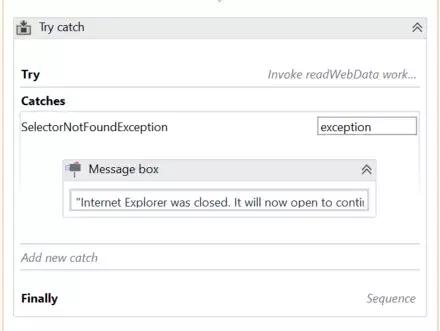
****

3.在「Catches」部分中，從下拉列表中選擇UiPath.Core.SelectorNotFoundException。如果沒有找到這個異常類型，點擊「Browse and Select a .Net Type」，在彈出窗口中找到它。

****

4.可選操作。您可以在「Catches」部分中添加一個「Message Box」Activity，在Message Box的「內容」欄位中填寫引號之間的信息性消息，在我們的示例中「Internet Explorer」已關閉。

它現在將打開以繼續工作流程執行。這意味著每當捕獲到異常時，都會顯示此消息框，以通知用戶瀏覽器即將打開，以便成功執行工作流。

****

5.拖個「Element Exists」Activity到「Finally」部分中。

「ElementExists」Activity用於檢查InternetExplorer是否打開了我們需要的網址https://www.fakenamegenerator.com。

6.打開Internet Explorer並訪問前面提到的網址。

7.使用Indicate on screen功能選擇Internet Explorer窗口。

8.選擇「Element Exists」Activity並編輯其選擇器，選擇器內容是這樣<wnd app='iexplore.exe' title='Generate a Random Name - Fake NameGenerator - Internet Explorer' />。

此選擇器確保「Element Exists」Activity僅查找上述頁面打開的InternetExplorer窗口。

9.在Output屬性中，創建一個具有相關名稱的變量，例如browser。這是一個布爾變量，可幫助您確定Internet Explorer是否在指示的頁面上處於打開狀態。

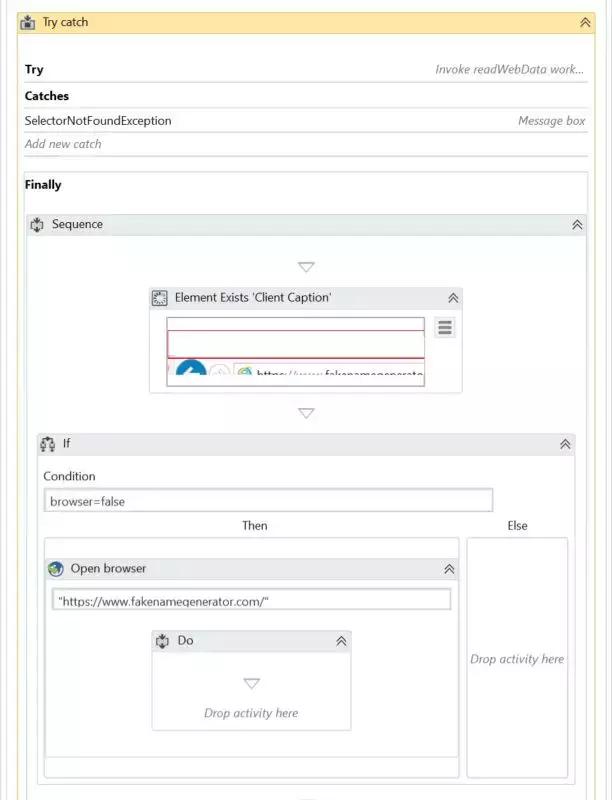
10.在「Element Exists」Activity下添加If Activity。如果Internet Explorer已關閉（通過browser的值來判斷），則打開Internet Explorer，否則繼續工作流。

11.在If Activity的Condition中，寫入browser=false。此條件用於驗證瀏覽器是否已打開，並根據其值執行其他操作。

12.在Then部分中拖入OpenBrowser Activity。如果滿足條件（瀏覽器已關閉），則使用「Open Browser」Activity打開它，從而不影響工作流程。

13.在「Open Browser」的Url欄位中鍵入https://www.fakenamegenerator.com。

14.Else部分留空，以便如果已在Internet Explorer打開指定的網站，則工作流按預期繼續。

****

15. 運行工作流並注意以下其中一項：

* 如果Internet Explorer已關閉 - 將通知用戶Internet Explorer即將打開，以便工作流程可以繼續。瀏覽器打開，所有預期的數據都被收集並寫入Excel文件。
* 如果Internet Explorer已打開 - 工作流程按預期執行。