

08 執行錯誤處理

黃彬華編撰

- ❖ 錯誤處理導論
- ❖ Exception 類別階層架構
- ❖ 執行錯誤處理機制
- ❖ 自訂執行錯誤類別
- ❖ 自動釋放資源

❖ 語法錯誤 (Syntax Error)

- 語法錯誤又稱剖析錯誤 (parsing error)
- 一旦語法發生錯誤，無法進入到執行階段

❖ 執行錯誤

- 執行時會檢查 app 對記憶體存取或邏輯上的正確性，如果錯誤會產生執行失敗，執行時檢測到的錯誤稱為例外 (exception)
- 執行不一定會產生錯誤，例如除法中的除數不輸入 0 就不會出錯
- 本章重點在於處理執行錯誤

範例

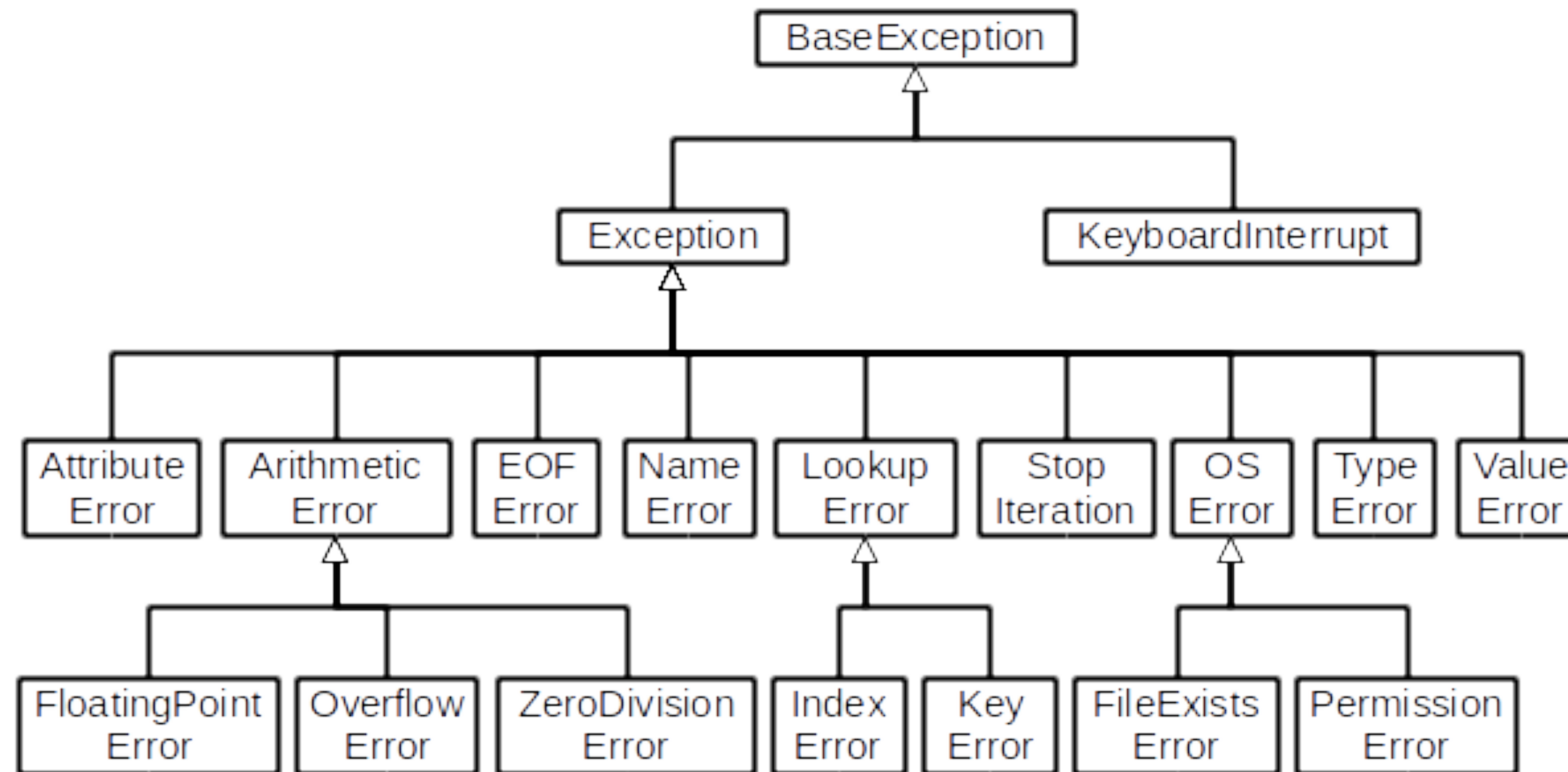
黃彬華編撰

❖ ExceptionDemo

Exception 類別階層架構

黃彬華編撰

❖ 官方 Exception 類別完整階層架構圖



執行錯誤處理機制 - 1

黃彬華編撰

- ❖ Exception 處理機制 (exception handling)
 - try
 - ✦ 把可能會產生 exception 的程式碼放在 try 子句
 - except
 - ✦ 將欲比對的錯誤放在 except 後方，並將處理結果放在 except 子句
 - ✦ 加上 as 可以取得觸發錯誤的 exception 物件，exception 有定義 `__str__()`
 - else
 - ✦ try 子句沒有發生 exception 時才需要執行的內容可以放在 else 子句
 - finally
 - ✦ 無論 try 子句是否發生 exception 都希望執行的內容 (例如釋放資源)，可放在 finally 子句

執行錯誤處理機制 - 2

黃彬華編撰

- ❖ 多個 except 子句各自處理的 exception 如果有繼承關係，則子型要排在前，父型要在後；沒有繼承關係，可以自由排列
 - 原因：try 子句發生 exception 後，會先比對第 1 個 except 子句的 exception 類型是否可以處理，如果是就不會再找下一個 except 子句。所以 exception 子型若排在後，處理內容就沒有被執行的機會

範例

黃彬華編撰

❖ ExceptionHandleDemo

練習 8-1

黃彬華編撰

輸入被除數(整數): a

被除數或除數格式錯誤，請重新輸入

輸入被除數(整數): 10

輸入除數(整數): 0

除數不可為0，請重新輸入

輸入被除數(整數): 10

輸入除數(整數): 3

$10 \div 3 = 3 \dots 1$

- ❖ 被除數與除數皆正確，顯示運算結果
- ❖ 被除數或除數格式錯誤，例如輸入 a，顯示「被除數或除數格式錯誤」
- ❖ 除數輸入 0，顯示「除數不可為0，請再重新輸入」

自訂執行錯誤類別

黃彬華編撰

- ❖ 開發者認為的錯誤，不一定可以在官方函式庫內找到適當的類型。有時必須自訂執行錯誤類別來描述，不過該類別仍需繼承官方 exception 類別，方能成為執行錯誤類別，也才能套用官方提供的執行錯誤處理機制
- ❖ 自訂執行錯誤類別步驟
 - 自訂類別並繼承官方 exception 類別
 - 符合自訂錯誤發生情況時，使用 "raise" 產生執行錯誤

範例

黃彬華編撰

❖ CustomExceptionDemo

練習 8-2

黃彬華編撰

立方體邊長: 0

CubeError: 正立方體邊長不可為0或負數，請重新輸入

立方體邊長: a

數字格式不正確，請重新輸入

立方體邊長: 3

邊長: 3.0, 體積: 27.0

- ❖ 定義正立方體 (Cube) 類別，內有邊長屬性 (length) 並用 private 封裝，有建構式與 setter、getter
- ❖ 呼叫建構式或 setter 設定邊長時，如果傳入的值為 0 或負數，會拋出自定的 CubeError，並顯示「正立方體邊長不可為0或負數」
- ❖ 使用者輸入的邊長必須為數字，如果為錯誤格式，顯示「數字格式不正確！」
- ❖ 輸入的邊長如果正確，會顯示體積

自動釋放資源

黃彬華編撰

- ❖ 使用資源時，無論程式處理過程成功或失敗，都應該關閉以釋放 (clean up) 該資源
 - 無論是否發生 exception，都要釋放該資源的程式碼，需要放在 finally 子句
 - with 敘述句能讓資源物件 (例如檔案) 在使用時，無論是否發生 exception，都能自動釋放該資源；減少撰寫 finally 子句麻煩

範例

黃彬華編撰

❖ CleanupDemo