

## Q：連連看- 例外發生的原因

- |                      |   |                     |
|----------------------|---|---------------------|
| 1) ZeroDivisionError | → | a) 運算的資料型態不一致       |
| 2) FileNotFoundError | → | b) 檔案發生輸出入錯誤        |
| 3) ValueError        | → | c) 索引超過範圍           |
| 4) TypeError         | → | d) 縮排不一致            |
| 5) IndexError        | → | e) int()函式傳入資料型態不合法 |
| 6) KeyError          | → | f) 變數名稱不存在          |
| 7) NameError         | → | g) 除法計算時除數為0        |
| 8) EOFError          | → | h) 字典Key不存在         |
| 9) IOError           | → | i) 檔案已達尾端           |
| 10) TabError         | → | j) 檔案找不到            |

## Q：下列關於例外處理的描述何者正確？

- a) try 結構中，try範圍內發生錯誤，一定會執行except區段內容
- b) try 結構中，try範圍內沒有錯誤，except區段內容不會執行
- c) try 結構中，不論有沒有發生錯誤，else區段內容一定會執行
- d) try 結構中，不論有沒有發生錯誤，finally區段內容一定會執行



Q：下列哪個選項可修正程式碼執行時發生的錯誤？

```
prices = [1.00, '3.50', 5.05]
total = 0

for item in prices:
    total += item

print(total)
```

- a) `total = total + item`
- b) `total += int(item)`
- c) `total += str(item)`
- d) `total += float(item)`



## Q：下列程式碼執行結果為何？

```
proLang = {1: 'Java', 2: 'Python', 3: 'C'}  
for i in range(0,3):  
    print(proLang[i])
```

- a) Java  
Python  
C
- b) 發生KeyError
- c) 發生NameError
- d) 發生IndexError

Q：程式碼中加入哪個選項可將補捉到的例外重拋？

```
try :
    x = 1/0
except Exception as e:
    print(e)
    # 加入程式
```

- a) raise as e
- b) **raise**
- c) raise ValueError(e)
- d) raise ValueError('Error Division by Zero')

## Q：哪個程式片段可在例外區段中捕捉多個例外？

```
def division(a, b):
    return a / b
try:
    division(4,0)
    division("3","4")
# 加入程式
    print("exception caught %s" %e)
```

- a) `except (ZeroDivisionError | TypeError) as e:`
- b) `except (ZeroDivisionError, TypeError) from e:`
- c) `except (ZeroDivisionError, TypeError) as e:`
- d) `except ZeroDivisionError, TypeError as e:`

# 練習：BMI 計算

- ◆ 修改 BMI 程式，加入例外處理機制避免下列問題造成程式中斷
  - ◆ 輸入非數值的身高體重
  - ◆ 身高為負數或0
  - ◆ 體重為負數或0
  - ◆ ....

```
bmi5.py - D:\Python\Solutions\Ch7\bmi5.py (3.8.1)
File Edit Format Run Options Window Help

def calcBMI(h,w):
    return w/((h/100)**2)

def diagnose(bmi):
    if(bmi>30):
        result = '肥胖'
    elif(bmi>25):
        result = '過重'
    elif(bmi>18.5):
        result = '正常'
    elif(bmi>0):
        result = '過輕'
    else:
        raise OverflowError('BMI數值錯誤'+str(bmi))
    return result

try :
    height = float(input('請輸入身高，單位為公分: '))
    weight = int(input('請輸入體重，單位為公斤: '))
    if height<0:
        raise ZeroDivisionError('身高數值錯誤:'+str(height))
    bmi = calcBMI(height, weight)
    print("bmi: %.2f, 判定結果: %s" %(bmi, diagnose(bmi)))
except ValueError as e:
    print('身高體重請輸入數值,', e)
except ZeroDivisionError as e:
    print('身高請輸入正值,', e)
except OverflowError as e:
    print('體重請輸入正值,', e)

Ln: 30 Col: 0
```

```
Python 3.8.1 Shell
File Edit Shell Debug Options Window Help

===== RESTART: D:\Python\Solutions\Ch7\bmi5.py =====
請輸入身高，單位為公分: 178
請輸入體重，單位為公斤: 80
bmi: 25.25, 判定結果: 過重
>>>

===== RESTART: D:\Python\Solutions\Ch7\bmi5.py =====
請輸入身高，單位為公分: ABC
身高體重請輸入數值, could not convert string to float: 'ABC'
>>>

===== RESTART: D:\Python\Solutions\Ch7\bmi5.py =====
請輸入身高，單位為公分: 178
請輸入體重，單位為公斤: XYZ
身高體重請輸入數值, invalid literal for int() with base 10: 'XYZ'
>>>

===== RESTART: D:\Python\Solutions\Ch7\bmi5.py =====
請輸入身高，單位為公分: 0
請輸入體重，單位為公斤: 80
身高請輸入正值, float division by zero
>>>

===== RESTART: D:\Python\Solutions\Ch7\bmi5.py =====
請輸入身高，單位為公分: -178
請輸入體重，單位為公斤: 80
身高請輸入正值, 身高數值錯誤:-178.0
>>>

===== RESTART: D:\Python\Solutions\Ch7\bmi5.py =====
請輸入身高，單位為公分: 178
請輸入體重，單位為公斤: 0
體重請輸入正值, BMI數值錯誤0.0
>>>

===== RESTART: D:\Python\Solutions\Ch7\bmi5.py =====
請輸入身高，單位為公分: 178
請輸入體重，單位為公斤: -80
體重請輸入正值, BMI數值錯誤-25.24933720489837
>>>

Ln: 37 Col: 4
```