

| PHẦN 1: LẬP TRÌNH PYTHON |                                |  |
|--------------------------|--------------------------------|--|
| NỘI DUNG:                | TRANG                          |  |
| 1                        | BIẾN TRONG PYTHON              |  |
| 2                        | CÁC KIỂU DỮ LIỆU và TOÁN TỬ    |  |
| 3                        | CẤU TRÚC ĐIỀU KHIỂN, VÒNG LẶP  |  |
| 4                        | HÀM TRONG PYTHON               |  |
| 5                        | LÀM VIỆC VỚI FILE TRONG PYTHON |  |
| 6                        | MODULE, XỬ LÝ NGOẠI LỆ         |  |
| 7                        | LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG      |  |

Tạ Quang Chiêu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275

148/112

148

A

D

A

I

# LÀM VIỆC VỚI TẬP TIN TRONG PYTHON

|   | A  | B            | C           | D      | E      |
|---|----|--------------|-------------|--------|--------|
| 1 | id | title        | timing      | genre  | rating |
| 2 |    | 1 Dog with a | 17:30-18:00 | Comedy | 4.7    |
| 3 |    | 2 Liv and Ma | 18:00-18:30 | Comedy | 6.3    |
| 4 |    | 3 Girl Meets | 18:30-19:00 | Comedy | 7.2    |
| 5 |    | 4 KC Underc  | 19:00-19:30 | Comedy | 6.1    |
| 6 |    | 5 Austin and | 19:30-20:00 | Comedy | 6      |
| 7 |    |              |             |        |        |

Tạ Quang Chiêu – [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) – [M]: 0913 522 275

149/112

149

|  |   |
|--|---|
| <div style="display: flex; justify-content: space-between;"> <span>A</span> <span>D</span> </div> <div style="display: flex; justify-content: space-between;"> <span>A</span> <span>I</span> </div>  | <h2 style="margin: 0;">File là gì?</h2> |
| <ul style="list-style-type: none"> <li>▪ File hay còn gọi là tệp, tập tin. File là tập hợp của các thông tin được đặt tên và lưu trữ trên bộ nhớ máy tính như đĩa cứng, đĩa mềm, CD, DVD,...</li> <li>▪ Khi muốn <b>đọc</b> hoặc <b>ghi file</b>, chúng ta cần phải <b>mở file</b> trước. <b>Khi hoàn thành</b>, file cần <b>phải được đóng</b> lại để các tài nguyên được gắn với file được giải phóng.</li> <li>▪ Do đó, trong Python, một thao tác với file diễn ra theo thứ tự sau:             <ul style="list-style-type: none"> <li>✓ Mở tệp tin</li> <li>✓ Đọc hoặc ghi</li> <li>✓ Đóng tệp</li> </ul> </li> </ul> |   |
| <div style="display: flex; justify-content: space-between;"> <span>Tạ Quang Chiêu - [E]: <a href="mailto:quangchieu.ta@gmail.com">quangchieu.ta@gmail.com</a> - [M]: 0913 522 275</span> <span style="background-color: #000080; color: white; padding: 2px 5px;">150/112</span> </div>  |   |

150

|  |                                     |
|--|-------------------------------------|
| <div style="display: flex; justify-content: space-between;"> <span>A</span> <span>D</span> </div> <div style="display: flex; justify-content: space-between;"> <span>A</span> <span>I</span> </div>  | <h2 style="margin: 0;">Mở File</h2> |
| <p>Để mở file trong Python chúng ta sử dụng hàm <b>open</b> với cú pháp như sau:</p> <div style="background-color: #333; color: #ccc; padding: 10px; text-align: center; margin: 10px 0;"> <code>open(filePath, mode, buffer)</code> </div> <p>Trong đó:</p> <ul style="list-style-type: none"> <li>▪ <b>filePath</b> là đường dẫn đến địa chỉ của file.</li> <li>▪ <b>mode</b> là thông số thiết lập chế độ chúng ta mở file được cấp những quyền gì? Mặc định mode sẽ bằng <b>r</b> (xem các mode ở dưới).</li> <li>▪ <b>buffer</b> là thông số đệm cho file mặc định thì nó sẽ là 0.</li> </ul> |                                     |
| <div style="display: flex; justify-content: space-between;"> <span>Tạ Quang Chiêu - [E]: <a href="mailto:quangchieu.ta@gmail.com">quangchieu.ta@gmail.com</a> - [M]: 0913 522 275</span> <span style="background-color: #000080; color: white; padding: 2px 5px;">151/112</span> </div>  |                                     |

151

A

D

A

I

## Mở File

### Các chế độ

| Mode | Chú thích   |
|------|---|
| r    | Chế độ chỉ được phép đọc.   |
| rb   | Chế độ chỉ được phép đọc nhưng cho định dạng nhị phân.  |
| r+   | Chế độ này cho phép đọc và ghi file, con trỏ nó sẽ nằm ở đầu file.  |
| rb+  | Chế độ này cho phép đọc và ghi file ở dạng nhị phân, con trỏ sẽ nằm ở đầu file.   |
| w    | Chế độ ghi file, nếu như file không tồn tại thì nó sẽ tạo mới file và ghi nội dung, còn nếu như file đã tồn tại nó sẽ ghi đè nội dung lên file cũ.                  |
| wb   | Tương tự chế độ w nhưng đối với nhị phân.   |
| w+   | Mở file trong chế độ đọc và ghi. còn lại như w.   |
| wb+  | Giống chế độ w+ nhưng đối với nhị phân  |
| a    | Mở file trong chế độ ghi tiếp. Nếu file đã tồn tại rồi thì nó sẽ ghi tiếp nội dung, và nếu như file chưa tồn tại thì nó sẽ tạo một file mới và ghi nội dung vào đó. |
| ab   | Tương tự a nhưng đối với nhị phân.  |
| a+   | Mở file trong chế độ đọc và ghi tiếp nội dung, còn lại cơ chế giống chế độ a.   |
| ab+  | Tương tự chế độ a+ nhưng đối với nhị phân.  |

Tạ Quang Chiêu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275
152/112

152

A

D

A

I

## Mở File

➤ Ví dụ:

```
f=open("test.txt") #mở file mode 'r' hoặc 'rt' để đọc

f=open("test.txt",'w') #mở file mode 'w' để ghi

import csv
f=open('data.csv','rt') #mở file mode 'r' hoặc 'rt' để đọc file csv
```

Tạ Quang Chiêu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275
153/112

153

A
D

A
I

## Đóng File

- Việc đóng file được xây dựng trong Python bằng hàm `close()` với cú pháp như sau:

`fileObject.close()`
- Trong đó, `fileObject` là đối tượng mà chúng ta thu được khi sử dụng hàm `open()`.

Tạ Quang Chiếu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275

154/112

154

A
D

A
I

## Đọc file

- Sau khi đã mở được file ra rồi, để đọc được file thì chúng ta sử dụng phương thức `read` với cú pháp:

`fileObject.read(length);`

Trong đó:

- `fileObject` là đối tượng mà chúng ta thu được khi sử dụng hàm `open()`.
- `length` là dung lượng của dữ liệu mà chúng ta muốn đọc, nếu để trống tham số này thì nó sẽ đọc hết file hoặc nếu file lớn quá thì nó sẽ đọc đến khi giới hạn của bộ nhớ cho phép.

Tạ Quang Chiếu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275

155/112

155

A

D

A

I

## Đọc file

➤ Ví dụ: đọc và xuất ra dữ liệu đọc được trong file readme ở trên

```
# Mở file để đọc dữ liệu
fo = open("E:\Python\File\data.txt", "r+")
# Đọc một chuỗi trong file
str = fo.read(20)
print("Chuỗi được đọc là: ", str)
str = fo.read()
# In ra chuỗi được đọc
print("Chuỗi được đọc là: ", str)
# Đóng file lại
fo.close()
```

Tạ Quang Chiêu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275

156/112

156

A

D

A

I

## Ghi file

- Để ghi được file thì bạn phải chắc chắn là đang mở file ở các chế độ cho phép ghi. Và sử dụng phương thức **write** với cú pháp sau:

`fileObject.write(data)`

Trong đó:

- **fileObject** là đối tượng mà chúng ta thu được khi sử dụng hàm **open()**.
- **data** là dữ liệu mà chúng ta muốn ghi vào trong file.

➤ Ví dụ:

```
# Mở file để đọc dữ liệu
fo = open('E:/Python/File/readme.txt', 'w')
#Ghi file
fo.write('Ta Quang Chiêu - khoa CNTT - HUMG')
# Đóng file lại
fo.close()
```

Tạ Quang Chiêu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275

157/112

157

A D
A I

## Ghi file

➤ Các thuộc tính trong file.

| Thuộc tính               | Chú thích   |
|--------------------------|---|
| <code>file.name</code>   | Trả về tên của file đang được mở.                               |
| <code>file.mode</code>   | Trả về chế độ mode của file đang được mở.                       |
| <code>file.closed</code> | Trả về true nếu file đã được đóng, và false nếu file chưa đóng. |

➤ Ví dụ: In ra thông số của file readme.md ở trên

```

1 # Mở file để đọc dữ liệu
2 fo = open('E:/Python/File/readme.txt', 'w')
3 #Trả về tên của file đang mở
4 print(fo.name)
5 #Trả về chế độ mode của file đang được mở.
6 print(fo.mode)
7 #Trả về true nếu file đã được đóng, và false nếu file chưa đóng.
8 print(fo.closed)
9 fo.close()

E:/Python/File/readme.txt
w
False
```

Tạ Quang Chiêu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275
158/112

158

A D
A I

## Ví dụ

Ví dụ 1: Ghi dữ liệu vào File "data.txt"

```

# Mở file để ghi
fo = open("data.txt", "w")
# Ghi dữ liệu lên file
fo.write("Tobe or not tobe. \n Nghi lon de thanh cong ! \n");
# Close opened file
fo.close()
print("Ghi file thanh cong !")
```

Ví dụ 2: Đọc và ghi dữ liệu từ một File

```

obj=open("test.txt","w")
obj.write("Chao mung cac ban den voi khoa CNTT")
obj.close()
obj1=open("test.txt","r")
s=obj1.read()
print (s)
obj1.close()
obj2=open("test.txt","r")
s1=obj2.read(20)
print (s1)
obj2.close()
```

Tạ Quang Chiêu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275
159/112

159

|   |   |                                   |
|---|---|-----------------------------------|
| A   | D | <h2>Một số hàm khác của file</h2> |
| A   | I |                                   |
| <ul style="list-style-type: none"> <li>▪ flush(): ép đẩy các dữ liệu trên bộ nhớ tạm ra file</li> <li>▪ tell(): trả về vị trí hiện tại của con trỏ file</li> <li>▪ seek(n): dịch con trỏ file đến vị trí byte thứ n             <ul style="list-style-type: none"> <li>▪ Hàm có thêm tham số thứ 2, cho phép diễn giải cách hiểu của tham số n</li> <li>▪ Nếu không viết, hoặc =0: vị trí n tính từ đầu file</li> <li>▪ =1: vị trí n tính từ vị trí hiện tại</li> <li>▪ =2: vị trí n tính từ cuối file</li> </ul> </li> <li>▪ truncate(n): cắt file ở vị trí byte thứ n, hoặc vị trí hiện tại (nếu không viết giá trị n)</li> </ul> |   |                                   |
| Tạ Quang Chiếu - [E]: <a href="mailto:quangchieu.ta@gmail.com">quangchieu.ta@gmail.com</a> - [M]: 0913 522 275  |   | 160/112                           |

160

|  |   |         |
|--|---|---------|
| A  | D |         |
| A  | I |         |
| <h1>THỰC HÀNH</h1>   |   |         |
| Tạ Quang Chiếu - [E]: <a href="mailto:quangchieu.ta@gmail.com">quangchieu.ta@gmail.com</a> - [M]: 0913 522 275 |   | 161/112 |

161

A

D

A

I

## Bài tập

### Bài 21: Đọc/Ghi file

**Đọc dữ liệu trong file dayso1\_bai17.txt chứa dữ liệu như hình dưới:**

- Tìm phần tử lớn nhất và nhỏ nhất trong dãy, sau đó thực hiện đổi chỗ phần tử lớn nhất xuất hiện đầu tiên trong dãy cho phần tử nhỏ nhất xuất hiện đầu tiên trong dãy. Lưu dãy mới đã đổi chỗ sang file dayso2\_bai17.txt

Tạ Quang Chiêu – [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) – [M]: 0913 522 275

162/112

162

A

D

A

I

## Bài tập

22. Đọc 1 file và in ra màn hình 5 dòng cuối cùng
23. Đọc 1 file, tìm và in ra nội dung của dòng dài nhất trong file đó
24. Đọc 1 file, tìm và in ra từ dài nhất trong file
25. Đọc 1 file, thống kê và in ra tất cả các chữ cái có trong file và số lần xuất hiện của các chữ đó
26. Đọc 1 file, thống kê và in ra tần suất xuất hiện của tất cả các từ trong file, in theo thứ tự giảm dần của số lần xuất hiện

Tạ Quang Chiêu – [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) – [M]: 0913 522 275

163/112

163



| PHẦN 1: LẬP TRÌNH PYTHON   |                                |         |
|--|--------------------------------|---------|
|  | NỘI DUNG:                      | TRANG   |
| 1  | BIẾN TRONG PYTHON              |         |
| 2  | CÁC KIỂU DỮ LIỆU và TOÁN TỬ    |         |
| 3  | CẤU TRÚC ĐIỀU KHIỂN, VÒNG LẶP  |         |
| 4  | HÀM TRONG PYTHON               |         |
| 5  | LÀM VIỆC VỚI FILE TRONG PYTHON |         |
| 6  | <b>MODULE, XỬ LÝ NGOẠI LỆ</b>  |         |
| 7  | LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG      |         |
| Tạ Quang Chiếu - [E]: <a href="mailto:quangchieu.ta@gmail.com">quangchieu.ta@gmail.com</a> - [M]: 0913 522 275 |                                | 164/112 |

164

| MODULE VÀ PACKAGE  |  |         |
|--|--|---------|
| Tạ Quang Chiếu - [E]: <a href="mailto:quangchieu.ta@gmail.com">quangchieu.ta@gmail.com</a> - [M]: 0913 522 275 |  | 165/112 |

165

A

D

Module

- Một file mã nguồn trong python được xem là một module
  - Có phần mở rộng .py
  - Mọi hàm, biến, kiểu trong file là các thành phần của module
- Sử dụng module:
  - Khai báo import module đó: `import <tên-module>`
  - Có thể khai báo import cùng lúc nhiều module cách nhau bởi dấu phẩy
  - Nếu muốn sử dụng các hàm, biến trong module thì cần viết trước mình tên module đó
  - Hoặc có thể import riêng một hàm hoặc nhiều hàm, cú pháp: `from <tên-module> import fuc1, fuc2,..., fucN`

Tạ Quang Chiêu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275

166/112

166

A

D

Module

- Ví dụ:
 

```
#Ví dụ về module Python

def sum(p, q):
    """Module này thêm 2 số
    và trả về kết quả """
    ket_qua = p + q
    return ket_qua
```
- **Nhập module và sửa tên khi nhập**

```
import math as m
print("Giá trị của pi là: ", m.pi)
```

Tạ Quang Chiêu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275

167/112

167

|  |                                    |
|--|------------------------------------|
| <div style="display: flex; justify-content: space-between;"> <div style="background-color: #000080; color: white; padding: 2px;">A</div> <div style="background-color: #000080; color: white; padding: 2px;">D</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="background-color: #000080; color: white; padding: 2px;">A</div> <div style="background-color: #000080; color: white; padding: 2px;">I</div> </div>  | <h2 style="margin: 0;">Module</h2> |
| <ul style="list-style-type: none"> <li> <div style="display: flex; align-items: flex-start;"> <div style="width: 20px; height: 10px; background-color: #000080; margin-bottom: 5px;"></div> <div> <p><b>Lệnh from...import trong Python</b></p> <p>Có thể nhập một tên cụ thể từ module mà không cần nhập toàn bộ module như ví dụ</p> <pre style="background-color: #e0f0e0; padding: 5px;">from math import pi print("Giá trị của pi là: ",pi)</pre> </div> </div> </li> <li> <div style="display: flex; align-items: flex-start;"> <div style="width: 20px; height: 10px; background-color: #000080; margin-bottom: 5px;"></div> <div> <p>Chúng ta có thể nhập nhiều thuộc tính của module như ví dụ dưới đây:</p> <pre style="background-color: #e0f0e0; padding: 5px;">from math import pi, e</pre> </div> </div> </li> <li> <div style="display: flex; align-items: flex-start;"> <div style="width: 20px; height: 10px; background-color: #000080; margin-bottom: 5px;"></div> <div> <p><b>Nhập tất cả tên</b></p> <pre style="background-color: #e0f0e0; padding: 5px;">from math import * print("Giá trị của pi là: ",pi)</pre> </div> </div> </li> </ul> |                                    |
| <div style="display: flex; justify-content: space-between;"> <span>Tạ Quang Chiêu - [E]: <a href="mailto:quangchieu.ta@gmail.com">quangchieu.ta@gmail.com</a> - [M]: 0913 522 275</span> <span style="background-color: #000080; color: white; padding: 2px 5px;">168/112</span> </div>  |                                    |

168

|  |                                    |
|--|------------------------------------|
| <div style="display: flex; justify-content: space-between;"> <div style="background-color: #000080; color: white; padding: 2px;">A</div> <div style="background-color: #000080; color: white; padding: 2px;">D</div> </div> <div style="display: flex; justify-content: space-between;"> <div style="background-color: #000080; color: white; padding: 2px;">A</div> <div style="background-color: #000080; color: white; padding: 2px;">I</div> </div>  | <h2 style="margin: 0;">Module</h2> |
| <ul style="list-style-type: none"> <li> <div style="display: flex; align-items: flex-start;"> <div style="width: 20px; height: 10px; background-color: #000080; margin-bottom: 5px;"></div> <div> <p><b>Đường dẫn tìm kiếm module Python</b></p> <p>Nhập lần lượt các lệnh sau để xem đường dẫn:</p> <pre style="background-color: #e0f0e0; padding: 5px;">&gt;&gt;&gt; import sys &gt;&gt;&gt; sys.path</pre> </div> </div> </li> <li> <div style="display: flex; align-items: flex-start;"> <div style="width: 20px; height: 10px; background-color: #000080; margin-bottom: 5px;"></div> <div> <p><b>Hàm dir() trong Python</b></p> <p>Hàm dir() trong Python được sử dụng để tìm ra các tên được định nghĩa trong một module</p> </div> </div> </li> </ul> |                                    |
| <div style="display: flex; justify-content: space-between;"> <span>Tạ Quang Chiêu - [E]: <a href="mailto:quangchieu.ta@gmail.com">quangchieu.ta@gmail.com</a> - [M]: 0913 522 275</span> <span style="background-color: #000080; color: white; padding: 2px 5px;">169/112</span> </div>  |                                    |

169

A

D

A

I

## Package

- Package = Thư mục các module (lưu trữ vật lý): **Package trong Python là một thư mục chứa một hoặc nhiều modules hay các package khác nhau**, nó được tạo ra nhằm mục đích phân bố các modules có cùng chức năng hay một cái gì đó, để dễ quản lý source code
- Module và Package giúp quản lý tốt hơn mã nguồn
- Gom, nhóm các hàm, biến, lớp xử lý cùng một chủ đề, giúp phân cấp và sử dụng dễ dàng hơn

Tạ Quang Chiêu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275

170/112

170

A

D

A

I

## Package

```
|--demopackage/
|               |--modules.py
|               |__ __init__.py
|--main.py
```

```
# demopackage/modules.py
def say_hello():
    print("Hello!")

def say_goodbye():
    print("Goodbye!")
```

Tạ Quang Chiêu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275

171/112

171

A

D

A

I

## Package

- Package chứa package.

```
|--demopackage/
|
|   |--modules.py
|   |--packagechild/
|       |-- childmodule.py
|       |   |-- __init__.py
|       |   |-- __init__.py
|   |-- __init__.py
|--main.py
```

Tạ Quang Chiêu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275

172/112

172

A

D

A

I

## XỬ LÝ NGOẠI LỆ

Tạ Quang Chiêu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275

173/112

173

A

D

A

I

## Ngoại lệ là gì?

- Ngoại lệ = lỗi, đúng, nhưng không hẳn
- Thường người ta chia lỗi thành 3 nhóm
  1. **Lỗi khi viết chương trình**: hệ quả là chương trình không chạy được nếu là thông dịch (hoặc không dịch được, nếu là biên dịch)
  2. **Lỗi khi chương trình chạy**: hệ quả là phải thực hiện lại
    - Chẳng hạn như nhập liệu không đúng, thì phải nhập lại
  3. **Ngoại lệ**: vẫn là lỗi, xảy ra khi có một bất thường và khiến một chức năng không thể thực hiện được
    - Chẳng hạn như đang ghi dữ liệu ra một file, nhưng file đó lại bị một tiến trình khác xóa mất

Tạ Quang Chiêu – [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) – [M]: 0913 522 275

174/112

174

A

D

A

I

## Ngoại lệ là gì?

- Ranh giới giữa ngoại lệ và lỗi khá mong manh, thậm chí khó phân biệt trong nhiều tình huống
- Cách chia lỗi thành 3 nhóm có khuynh hướng cho rằng môi trường thực thi của chương trình là thân thiện và hoàn hảo
- Python có xu hướng chia lỗi thành 2 loại
  - **Syntax error**: viết sai cú pháp, khiến chương trình thông dịch không dịch được
  - **Exception**: xảy ra bất thường không như thiết kế
    - Như vậy xử lý exception sẽ khiến chương trình ổn định và hoạt động tốt trong mọi tình huống

Tạ Quang Chiêu – [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) – [M]: 0913 522 275

175/112

175

A D
A I
Ngoại lệ là gì?

- Ví dụ về syntax error:
 

```
>>> while True print('Hello world')
      File "<stdin>", line 1
        while True print('Hello world')
                        ^
      SyntaxError: invalid syntax
```
- Ví dụ về exception:
 

```
>>> 10 * (1/0)
      Traceback (most recent call last):
        File "<stdin>", line 1, in <module>
      ZeroDivisionError: division by zero
```
- Có vẻ như syntax error cũng chỉ là một exception!!!

Tạ Quang Chiêu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275
176/112

176

A D
A I
Ngoại lệ là gì?

```
while True:
    try:
        x = int(input("Nhập số X: "))
        break
    except ValueError:
        print("Lỗi, hãy nhập lại.")
print("X =", x)
```

Vòng lặp nhập X  
cho đến khi người dùng  
nhập vào đúng giá trị số

Khối nhập X  
(có thể nhập lỗi)

Xử lý khi lỗi xảy ra

Tạ Quang Chiêu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275
177/112

177

|   |   |                                 |
|---|---|---------------------------------|
| A | D | Cú pháp try-except-else-finally |
| A | I |                                 |

■ Có thể gồm tới 4 khối:

- Khối **“try”**: đoạn mã có khả năng gây lỗi, khi lỗi xảy ra, khối này sẽ bị dừng ở dòng gây lỗi
- Khối **“except”**: đoạn mã xử lý lỗi, chỉ thực hiện nếu có lỗi xảy ra, nếu không sẽ bị bỏ qua
- Khối **“else”**: có thể xuất hiện ngay sau khối **except** cuối cùng, đoạn mã sẽ được thực hiện nếu không có **except** nào được thực hiện (đoạn try không có lỗi)
- Khối **“finally”**: còn được gọi là khối clean-up, luôn được thực hiện dù có xảy ra lỗi hay không

Tạ Quang Chiêu – [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) – [M]: 0913 522 275

178/112

178

|   |   |                                 |
|---|---|---------------------------------|
| A | D | Cú pháp try-except-else-finally |
| A | I |                                 |

■ Chú ý:

- Khối **try** chỉ có 1 khối duy nhất, phải viết đầu tiên
- Khối **finally** có thể có hay không, nếu có thì khối này phải viết cuối cùng
- Khối **except** có thể không viết, có một khối, hoặc nhiều khối except (để xử lý nhiều tình huống lỗi khác nhau)
- Một khối **except** có thể xử lý một loại lỗi, nhiều loại lỗi hoặc tất cả các loại lỗi

Tạ Quang Chiêu – [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) – [M]: 0913 522 275

179/112

179



A

D

A

I

## Cú pháp try-except-else-finally

```

except (NameError, TypeError):
    print("Name or Type error")

except IOError as e:
    print(e)
    raise

except ValueError:
    print("Value error")

except:
    print("An error occurred")
    raise NameError("Ko bit")

else:
    print("OK")

```

- # xử lý 2 loại lỗi
- # lấy đối tượng lỗi, đặt tên e
- # trả lại lỗi này
- # xử lý lỗi Value
- # xử lý tất cả các lỗi còn lại
- # tạo ra một lỗi "Ko bit"
- # thực hiện nếu không có lỗi nào

Tạ Quang Chiêu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275

180/112

180

A

D

A

I

## Một số loại exception thường gặp

| Exception          | Miêu tả   |
|--------------------|---|
| Exception          | Lớp cơ sở (base class) của tất cả các ngoại lệ                                      |
| StopIteration      | Được tạo khi phương thức next() của một iterator không trở tới bất kỳ đối tượng nào |
| StandardError      | Lớp cơ sở của tất cả exception có sẵn ngoại trừ StopIteration và SystemExit         |
| ArithmeticError    | Lớp cơ sở của tất cả các lỗi xảy ra cho phép tính số học                            |
| OverflowError      | Được tạo khi một phép tính vượt quá giới hạn tối đa cho một kiểu số                 |
| FloatingPointError | Được tạo khi một phép tính số thực thất bại   |
| ZeroDivisionError  | Được tạo khi thực hiện phép chia cho số 0 với tất cả kiểu số                        |
| AssertionError     | Được tạo trong trường hợp lệnh assert thất bại                                      |

Tạ Quang Chiêu - [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) - [M]: 0913 522 275

181/112

181

A

D

A

I

## Một số loại exception thường gặp

| Exception         | Miêu tả   |
|-------------------|---|
| AttributeError    | Được tạo trong trường hợp tham chiếu hoặc gán thuộc tính thất bại   |
| EOFError          | Được tạo khi không có input nào từ hàm <code>raw_input()</code> hoặc hàm <code>input()</code> và tới EOF (viết tắt của end of file) |
| ImportError       | Được tạo khi một lệnh import thất bại   |
| KeyboardInterrupt | Được tạo khi người dùng ngắt việc thực thi chương trình, thường là bởi nhấn Ctrl+c  |
| LookupError       | Lớp cơ sở cho tất cả các lỗi truy cứu   |
| IndexError        | Được tạo khi một chỉ mục không được tìm thấy trong một dãy (sequence)   |
| KeyError          | Được tạo khi key đã cho không được tìm thấy trong Dictionary  |
| NameError         | Được tạo khi một định danh không được tìm thấy trong local hoặc global namespace  |

Tạ Quang Chiêu – [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) – [M]: 0913 522 275

182/112

182

A

D

A

I

## Một số loại exception thường gặp

| Exception         | Miêu tả  |
|-------------------|--|
| UnboundLocalError | Được tạo khi cố gắng truy cập một biến cục bộ từ một hàm hoặc phương thức nhưng mà không có giá trị nào đã được gán cho nó       |
| EnvironmentError  | Lớp cơ sở cho tất cả ngoại lệ mà xuất hiện ở ngoài môi trường Python   |
| IOError           | Được tạo khi hoạt động i/o thất bại, chẳng hạn như lệnh print hoặc hàm <code>open()</code> khi cố gắng mở một file không tồn tại |
| OSError           | Được do các lỗi liên quan tới hệ điều hành   |
| SyntaxError       | Được tạo khi có một lỗi liên quan tới cú pháp  |
| IndentationError  | Được tạo khi độ thụt dòng code không được xác định hợp lý  |
| SystemError       | Được tạo khi trình thông dịch tìm thấy một vấn đề nội tại, nhưng khi lỗi này được bắt gặp thì trình thông dịch không thoát ra    |

Tạ Quang Chiêu – [E]: [quangchieu.ta@gmail.com](mailto:quangchieu.ta@gmail.com) – [M]: 0913 522 275

183/112

183

| <div> <div>A</div> <div>D</div> <div>A</div> <div>I</div> </div> <b>Một số loại exception thường gặp</b>                          |   |
|---|---|
| Exception   | Miêu tả   |
| SystemExit  | Được tạo khi trình thông dịch thoát ra bởi sử dụng hàm <code>sys.exit()</code> . Nếu không được xử lý trong code, sẽ làm cho trình thông dịch thoát |
| TypeError   | Được tạo khi một hoạt động hoặc hàm sử dụng một kiểu dữ liệu không hợp lệ   |
| ValueError  | Được tạo khi hàm đã được xây dựng sẵn có các kiểu tham số hợp lệ nhưng các giá trị được xác định cho tham số đó là không hợp lệ                     |
| RuntimeError  | Được tạo khi một lỗi đã được tạo ra là không trong loại nào   |
| NotImplementedError   | Được tạo khi một phương thức abstract, mà cần được triển khai trong một lớp được kế thừa, đã không được triển khai thực sự                          |
| Tạ Quang Chiếu – [E]: <a href="mailto:quangchieu.ta@gmail.com">quangchieu.ta@gmail.com</a> – [M]: 0913 522 275 <div>184/112</div> |   |