

# Điều khiển luồng



TH.S TRẦN ĐỨC LỢI  
PYTHONVIETNAM.INFO

# Ôn tập bài cũ



# Mục đích bài học



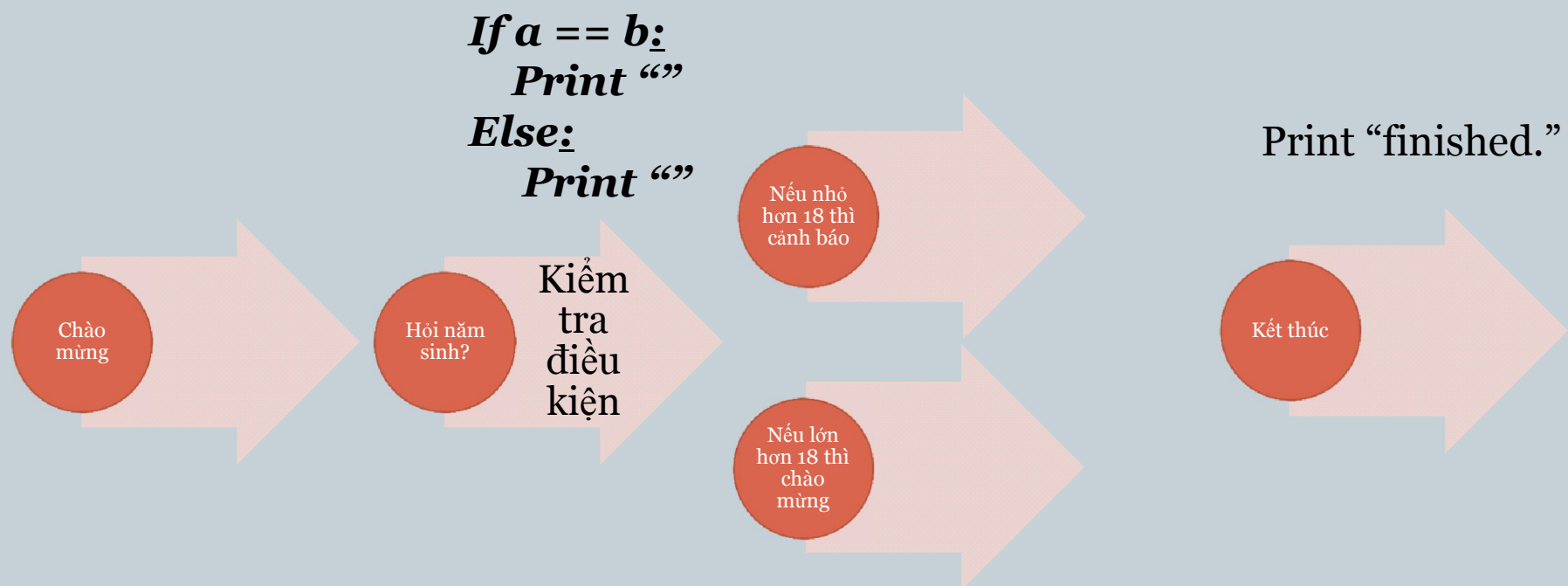
- Tìm hiểu về điều khiển luồng trong python với các câu lệnh if-else, while, for, ...
- Cấu trúc try-catch

# Điều khiển rẽ nhánh



- Đặt vấn đề: Viết thuật toán và code chương trình *python18+*:
  - Chào mừng người dùng
  - Hỏi năm sinh
  - Nếu người dùng nhỏ hơn 18 tuổi thì thông báo không đủ tuổi
  - Nếu đủ hoặc hơn 18 tuổi thì thông báo chào mừng

# Điều khiển rẽ nhánh



# Điều khiển rẽ nhánh if



## Đặc điểm

- Không thay đổi giá trị của biến tại biểu thức so sánh
- Biểu thức này yêu cầu trả về True/False
- Sử dụng căn lề để bắt đầu – kết thúc một block
- If-elif-else

# Điều khiển rẽ nhánh đa quyết định



- Đặt vấn đề: Viết thuật toán và code chương trình *python18++*:
  - Chào mừng người dùng
  - Hỏi năm sinh
  - Nếu người dùng nhỏ hơn 16 tuổi thì thông báo không đủ tuổi
  - Nếu đủ hoặc hơn 18 tuổi thì thông báo chào mừng
  - Nếu từ 16-18 tuổi thì thông báo nên cân nhắc

# Điều khiển rẽ nhánh if



- *If*  $a == b$  :
- *Print* «»
- *Elif*  $b == c$  :
- *Print* «»
- *Else:*
- *Print* «»



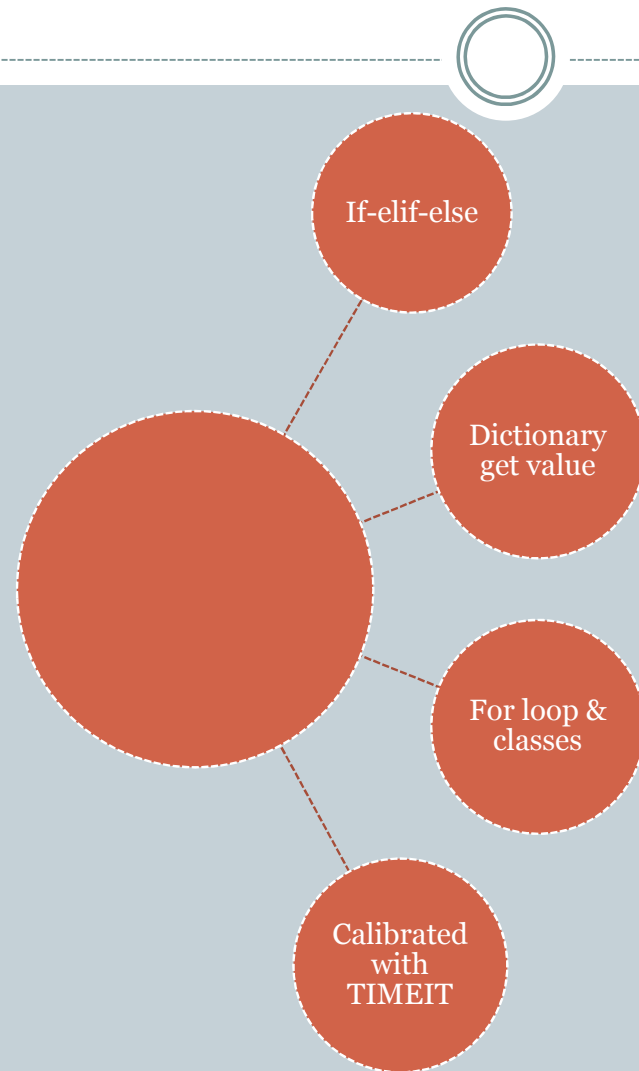
# Switch?



Không có lệnh  
switch-case  
trong python

Sử dụng các cấu  
trúc khác để  
thay thế

# Switch?



# Vòng lặp While



- *While a == b:*
- *Print «»*
- *Else:*
- *Print «the loop is ended»*
- *Print «finished.»*

# Vòng lặp for



- *For i in range( 1,5):*
- *Print «»*
- *Else:*
- *Print «»*

# Câu lệnh break



- Thoát khỏi vòng lặp bằng câu lệnh break

# Lệnh continue



- Bỏ qua một số câu lệnh trong vòng loop bằng lệnh continue
- Các dòng lệnh trong vòng loop và sau câu continue sẽ bị bỏ qua, không thực hiện mà chuyển sang vòng loop tiếp theo luôn

# Exception



- Exception hay ngoại lệ xảy ra theo một tình huống nào đó trong một chương trình.
- Ví dụ:
  - - đọc file nhưng file không tồn tại
  - - delete file khi đang sử dụng
  - ...
- và từ đây sinh ra khái niệm bắt ngoại lệ.

# Exception



- Nên bao quanh các đoạn code “nguy hiểm” bằng đoạn `try...except`
- Trong đoạn này, nếu `try` block được thực thi thì `except` block sẽ được bỏ qua và **ngược lại**
- Bài `pythoncalculator`



# Exception



- `try:`
- `f = open('x')`
- `except:`
- `print 'Exception occurred'`
- `print 'prog will now exit'`

# Exception



- `try:`
- `f = open('x')`
- `except IOError:`
- `print 'caught'`
- `except:`
- `print 'x'`
- `print 'prog will now exit'`

# Cấu trúc try-catch



- `d = int(raw_input("Nhap gioi han cua day fibonacci:"))`
- Try:
- Print «»
- Except Exception, e:
- Print «»

# Raise Exception



- Exception có sẵn của python
- try:
- raise IOError()
- except IOError, e:
- print e
- except:
- print 'x'

# Try ... Finally



- `try:`
- `raise LoiEx('I hate')`
- `#f = open('x')`
- `except IOError, e:`
- `print e`
- `finally:`
- `print 'I am the final'`

# Tổng kết bài học



- If-elif-else
- While loop
- For loop
- Exceptions

Cảm ơn đã lắng nghe

