





NÊN TẢNG MÁY TÍNH

(Fundamental of computing)

Bài 5: LIST, DICTIOARY, TUPLE



GVGD:

- THS. NGUYỄN THÁI HẢI

 hai.nt@vlu.edu.vn

 nguyenthaihaivlu@gmail.com
 - THS. NGUYỄN THỊ QUYÊN quyen.nt@vlu.edu.vn



HỌC KỲ II – NĂM HỌC 2020-2021



KHÓA 25Q22, 25Q23



01. KIỂU DỮ LIỆU LIST

NỘI DUNG 02. KIỂU DỮ LIỆU DICTIONARY

03. KIỂU DỮ LIỆU TUPLE

04. BÀI TẬP



1. KIỂU DỮ LIỆU LIST (danh sách)

- List dùng để lưu các giá trị (item) vào cùng một biến
- Cho phép dữ liệu trùng nhau
- List được tạo ra dựa vào cặp dấu ngoặc vuông []
 listStudent = ["IT001", "IT002", "IT002"]
- Được truy xuất thông qua vị trí (index)
 print (listStudent[0])
 print (listStudent[1])



Một số hàm thao tác với List

Thêm giá trị vào bất kỳ lúc nào

```
listStudent.append("IT004")
listStudent.append(9.5)
print(listStudent)
Output:['IT001', 'IT002', 'IT003', 'IT004', 9.5]
```

Chiều dài của List

```
chieu_dai = len(listStudent)
print(chieu dai)
```



Truy xuất giá trị tại vị trí âm (vị trí từ cuối lên)
 print (listStudent[-1]) #9.5
 print (listStudent[-2]) #IT004

• Lấy một đoạn dữ liệu (lấy trong đoạn [a:b))
 new_list = listStudent[0: 2]
 print (new_list)
Outut: ['IT001', 'IT002']



• Lấy trong đoạn (tt)

```
print(listStudent[:3])
#Dấu hai chấm ở đầu cho biết lấy từ đầu đến phần tử thứ 3
Output: ['IT001', 'IT002', 'IT008']
```

Lấy trong đoạn (tt)

```
print(listStudent[1:])
```

```
#Dấu hai chấm ở cuối cho biết, lấy từ vị trí thứ 1 đến hết Output: ['IT002', 'IT008', 'IT004', 9.5]
```

• Kiểm tra 1 phần tử có trong List hay không?

```
if "IT002" in listStudent:
   print("Có xuất hiện")
```



• Chèn thêm giá trị tại vị trí mong muốn

```
listStudent.insert(1, "IT009")
    print(listStudent)
#Thêm giá trị tại vị trí thứ 1
Output:['IT001', 'IT009', 'IT002', 'IT008', 'IT004', 9.5]
```

Hàm Extend

```
list1 = ["a", "b", "c"]
list2 = ["d", "e", "f"]
list1.extend(list2)
print(list1)
output: ['a', 'b', 'c', 'd', 'e', 'f']
```



Cộng 2 danh sách với nhau

```
list1 = ["a", "b", "c"]
list2 = ["d", "e", "f"]
new_list = list1 + list2
print(new list)
```

Xóa môt item ra khỏi danh sách

```
list1 = ["a", "b", "c"]
    list1.remove("a")
    print(list1)
output: ['b', 'c']
```

Thảo luận: Điều gì xảy ra nếu phần tử cần xóa không có trong List



Xóa phần tử dựa vào phương thức pop

```
list1 = ["a", "b", "c"]
list1.pop() # xóa phần tử cuối
print(list1)
```

Xóa dựa vào vị trí

```
list1 = ["a", "b", "c"]
list1.pop(1) # xóa phần tử tại vị trí 1
print(list1)
```

- Copy()
- Sort()
- Reverse()



Duyệt phần tử có trong danh sách

```
list1 = ["a", "b", "c"]
for x in list1:
    print(x)
#Cách 2
for i in range(len(list1)):
    print(list1[i])
```

* Thảo luận: hãy duyệt danh sách theo vòng lặp while



List comprehension

Sử dụng list comprehension để cú pháp "gọn" hơn

Ví du 1:

```
list1 = ["a", "b", "c"]
[print(x) for x in list1]
```

Ví dụ 2: Lấy danh sách các số chẵn

```
list1 = [1, 0, 2, 5 ,7 ,8 , 10]
list_cacsochan = [x for x in list1 if x % 2 == 0]
print(list_cacsochan) #[0, 2, 8, 10]
```



2. KIỂU DỮ LIỆU DICTIONARY

- Kiểu dictionary thường dùng để lưu giá trị 1 cặp có dạng key:value
- Không cho phép khóa key trùng giá trị
- Được bao bởi cặp dấu ngoặc nhọn { }
- Ví dụ:

```
adict = {
  "001": "Nguyễn Văn A",
  "002": "Nguyễn Thị B",
  "003": "Trần Anh C"
}
print(adict)
```



Truy xuất phần tử trong Dictionary

```
adict = {
  "001": "Nguyễn Văn A",
  "002": "Nguyễn Thị B",
  "003": "Trần Anh C"
print(adict["002"]) #lấy value dựa vào key "002"
#hoặc dùng phương thức get()
x = adict.get("002")
print(x)
```



- Thêm phần tử
 - + Nếu key chưa tồn tại mà được gán một giá trị, dictionary sẽ tự thêm mới phần tử đó vào dictionary

```
adict["004"] = "Nguyễn D"

print(adict)

Output: {'001': 'Nguyễn Văn A', '002': 'Nguyễn Thị B', '003': 'Trần Anh C', '004': 'Nguyễn D'}
```

Lấy danh sách các keys

```
danhSachKeys = adict.keys()
danhSachValues = adict.values()
for k in danhSachKeys:
    print(k)
```



Thay đổi giá trị tại mội key nào đó

```
adict["003"] = "Trần Tuấn Anh"

print(adict)

→ Thay đổi giá trị mới tại key là "003"

# Hoặc dùng hàm update()

adict.update({"002": "Lại Lý Huynh"})

→ Thay đổi giá trị mới tại key là "002"
```

Kiểm tra tồn tại một key có trong dictionary?

```
key_search = '001'
if key_search in adict:
    print("Tìm thấy")
```



Xóa phần tử dựa vào key cho trước

```
adict.pop("001")
# del adic["001"]
Out: {'002': 'Nguyễn Thị B', '003': 'Trần Anh C'}
```

Xóa phần tử cuối cùng

```
dict = {
   "001": "Nguyễn Văn A",
   "002": "Nguyễn Thị B",
   "003": "Trần Anh C"
}
dict.popitem()
print(dict)
Out: {'001': 'Nguyễn Văn A', '002': 'Nguyễn Thị B'}
```



Xóa tất cả các phần tử

```
dict = {
    "001": "Nguyễn Văn A",
    "002": "Nguyễn Thị B",
    }
    dict.clear()
    print(dict)
out: {}
```

Xóa dictionary

```
del adict
print(adict)
```

Out: Traceback (most recent call last): NameError: name

'adict' is not defined

Thảo luận: Phân biệt sự khác nhau giữa xóa tất cả phần tử và xóa dictionary?



Duyệt các key trong dictionary

```
adict = {
        "001": "Nguyễn Văn A",
        "002": "Nguyễn Thị B",
        "003": "Trần Anh C"
      for k in adict:
        print(k)
out: 001 002 003
```



Duyệt các value trong dictionary

```
adict = {
        "001": "Nguyễn Văn A",
        "002": "Nguyễn Thị B",
        "003": "Trần Anh C"
      for k in adict:
        print(addict[k])
out: ??
```



Duyệt từng cặp key, value

```
adict = {
        "001": "Nguyễn Văn A",
        "002": "Nguyễn Thị B",
        "003": "Trần Anh C"
      for k, v in addict.items():
       print(k, " - ", v)
out: 001 - Nguyễn Văn A
      002 - Nguyễn Thị B
      003 - Trần Anh C
```



Tạo một phiên bản từ dictionary cho trước

```
adict = {
        "001": "Nguyễn Văn A",
        "002": "Nguyễn Thị B",
      bdict = adict.copy()
      print(adict)
      print(bdict)
Out:
      {'001': 'Nguyễn Văn A', '002': 'Nguyễn Thị B'}
      {'001': 'Nguyễn Văn A', '002': 'Nguyễn Thị B'}
```



3. KIỂU DỮ LIỆU TUPLE

- Cách dùng hoàn toàn như kiểu dữ liệu List
- •
- •
- •
- Sinh viên tự tìm hiểu thêm kiểu dữ liệu này và trả lời câu hỏi. Giữa List và Tuple khác nhau ở điểm nào? Khi nào sử dụng Tuple để thay thế cho List



4. BÀI TẬP

- 1. Khai báo một danh sách chứa tên các môn học ở học kỳ này
- Hãy dùng vòng lặp for in ra danh sách
- Dùng vòng lặp while in ra danh sách
- Cập nhật danh sách tại vị trí thứ 2 là môn "Python cơ bản"
- Thêm môn học "Python nâng cao" vào cuối danh sách
- Thêm môn học "Hướng đối tượng" vào đầu danh sách
- Kiểm tra xong trong danh sách có môn học "Python cơ bản" không?
- Xóa trong danh sách môn học "Hướng đối tượng"
- Làm rỗng danh sách đã cho



- 2. Khai báo một dictionary chứa 5 sinh viên bao gồm key là MSSV, value là Tên sinh viên
- In ra thông tin của dictionary này bao gồm mssv và tên sv
- Cho người dùng nhập vào mã sinh viên. Xuất ra tên của sinh viên đó, nếu không có in ra thông báo "Không tìm thấy"
- Cho người dùng nhập vào mã số sinh viên và nhập vào tên cần thay đổi thông tin. Hãy cập nhật lại tên của sinh viên có mã số ở trên
- Cho người dùng nhập vào mssv và tên. Nếu chưa có thì thêm vào dictionary, nếu có rồi thì báo "Sinh viên đã tồn tại"
- Cho người dùng nhập vào mssv cần xóa. Nếu tồn tại mssv đó thì xóa trong dictionary, không thì báo "Không tìm thấy mssv cần xóa"
- Đếm xem trong dictionary hiện tại có bao nhiều sinh viên





