



LẬP TRÌNH PYTHON

Giới thiệu

hungdn@ptit.edu.vn



Tại sao Python?

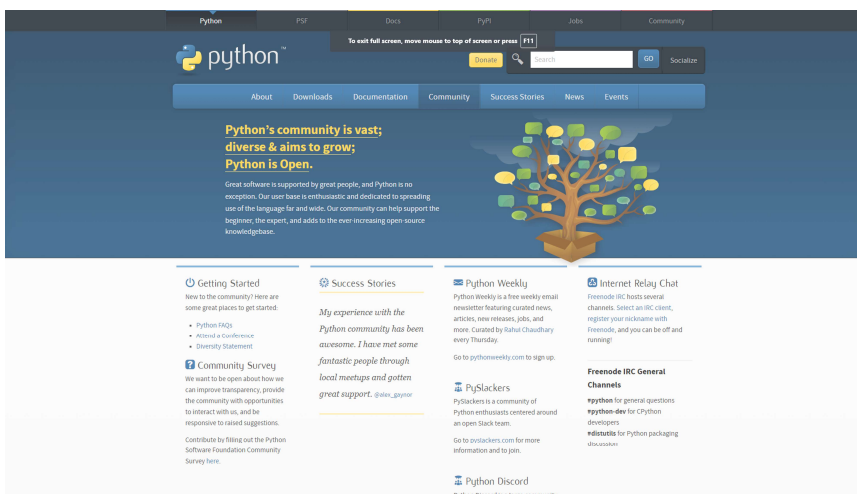
1. Đơn giản để học và dùng



- Yêu cầu kiến thức máy tính tối thiểu
- Dễ đọc, dễ hiểu
- Tập trung vào sự đơn giản và khuyến khích mã đẹp
- Memory vs. Hard Drive
- Operating Systems
- Compilers



2. Cộng đồng tuyệt vời



- Package
- Tool
- Khác

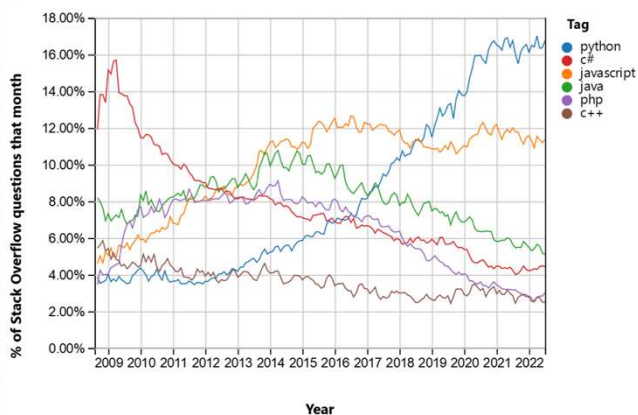
3. Lĩnh vực ứng dụng



- **Web development**
 - API (Flask, Bottle, Pyramid)
 - Website (Django, TurboGears, Web2py)
 - App (Plone, Django-CMS, Mezzanine)
- **Data Science**
 - Big data
 - Machine Learning
- **Education and Learning**
 - STEM
 - Programming
 - Hardware
- **Scripting**
 - Network
 - Machine / Application



4. Nhu cầu



Nguồn: <https://insights.stackoverflow.com/trends/>

Rank	Change	Language	Share	Trend
1		Python	28.11 %	-2.6 %
2		Java	17.35 %	-0.9 %
3		JavaScript	9.48 %	+0.2 %
4		C#	7.08 %	+0.1 %
5		C/C++	6.19 %	-0.3 %
6		PHP	5.47 %	-0.8 %
7		R	4.35 %	+0.6 %
8	↑↑	TypeScript	2.79 %	+1.1 %
9	↑↑	Swift	2.09 %	+0.5 %
10	↓↓	Objective-C	2.03 %	+0.2 %
11	↑	Go	2.03 %	+0.5 %

Nguồn: <https://pypl.github.io/PYPL.html>

4. Nhu cầu



Rank	Language	Type	Score
1	Python		100.0
2	Java		95.4
3	C		94.7
4	C++		92.4
5	JavaScript		88.1
6	C#		82.4
7	R		81.7
8	Go		77.7
9	HTML		75.4
10	Swift		70.4

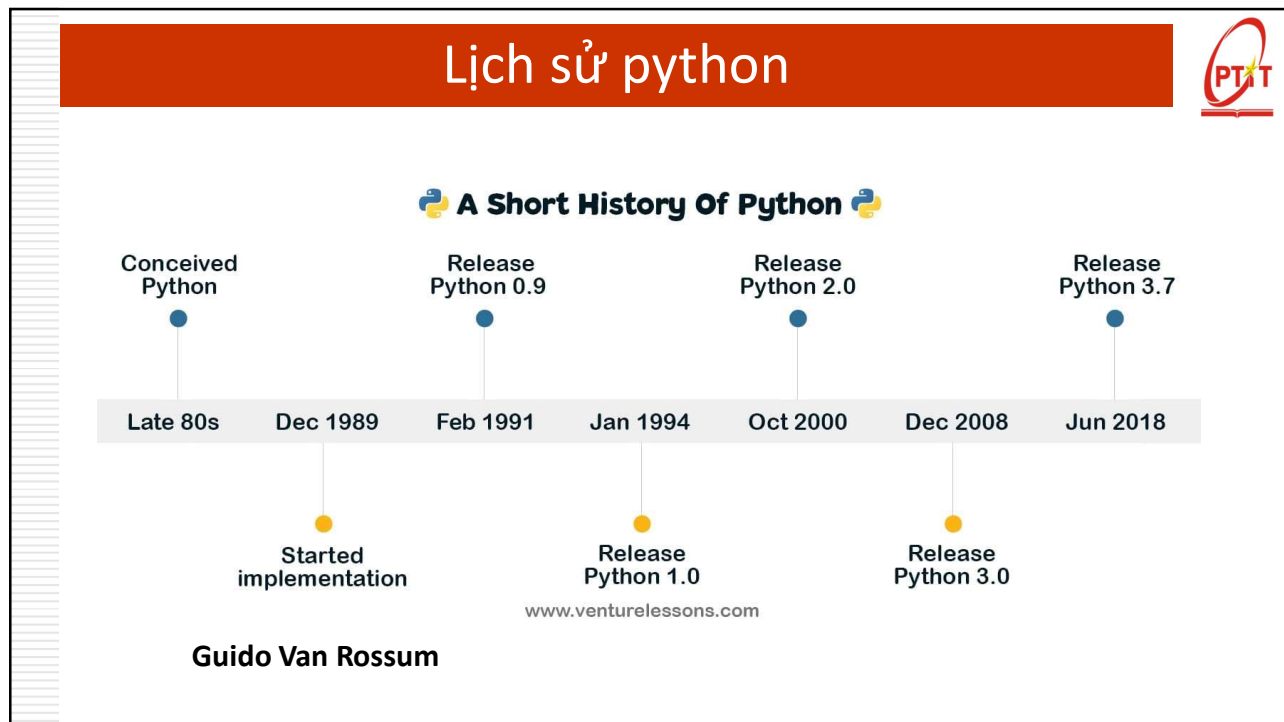
Nguồn: <https://spectrum.ieee.org/top-programming-languages/>

Aug 2022	Aug 2021	Change	Programming Language	Ratings	Change
1	2	▲	Python	15.42%	+3.56%
2	1	▼	C	14.59%	+2.03%
3	3		Java	12.40%	+1.96%
4	4		C++	10.17%	+2.81%
5	5		C#	5.59%	+0.45%
6	6		Visual Basic	4.99%	+0.33%
7	7		JavaScript	2.33%	-0.61%
8	9	▲	Assembly language	2.17%	+0.14%
9	10	▲	SQL	1.70%	+0.23%
10	8	▼	PHP	1.39%	-0.80%

Nguồn: <https://www.tiobe.com/tiobe-index/>



Python?



1. Python Syntax

```

long Ex_function();
/* int */ other_function();
/* int */ calling_function()
{
    long test1;
    register /* int */ test2;
    test1 = Ex_function();
    if (test1 > 0)
        test2 = 0;
    else
        test2 = other_function();
    return test2;
}

```

```

n = int(input('input number: '))
if n < 0:
    raise ValueError('Non-negative integer')
fact = 1
for i in range(2, n + 1):
    fact *= i
print(fact)

```

C, C++, C#, Java, JavaScript, Go, PHP ...

PYTHON

2. General-purpose and High-Level



• Ứng dụng



• Lập trình bậc cao



3. Multi-paradigm



- **Structured Programming (Lập trình cấu trúc)**
- **Object-oriented Programming (Lập trình hướng đối tượng)**
- **Functional Programming (Lập trình hàm)**

```
class Animal:
    def pet(self):
        print("...")
class Dog(Animal):
    def pet(self):
        print("bark!")
class Cat(Animal):
    def pet(self):
        print("meow!")
def pet_animal(animal):
    animal.pet()

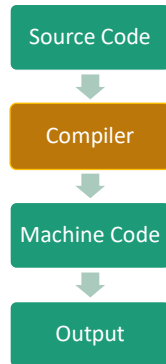
pet_animal(Dog())
pet_animal(Cat())
```

Object-oriented Programming

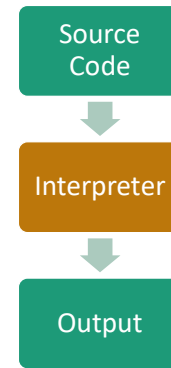
4. Python Is Interpreted



• Compiler



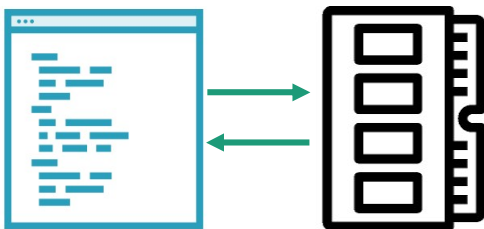
• Interpreter



5. Python Is Garbage-collected



• Chương trình



- Trách nhiệm dọn dẹp bộ nhớ
- Tránh rò rỉ/lãng phí bộ nhớ
- Ngăn chặn hầu hết các vấn đề bảo mật
- Triển khai hiệu quả một số cấu trúc dữ liệu

6. Python Is Dynamically-typed



• Static Type

```
Int num = 100;
...
String str = "Hello World";

num = "50"; // Error
num = "one hundred"; Error
```

• Dynamic Type

```
answer = 100
...
...
answer = "one hundred"
...
...
answer = 100.0

...
answer + "two"

# TypeError: unsupported operand
type(s) for +: 'float' and 'str'
```

Python



1. Syntax
2. General-purpose
3. Multi-paradigm
4. Interpreted
5. Garbage-collectedSyntax
6. Dynamically-typed

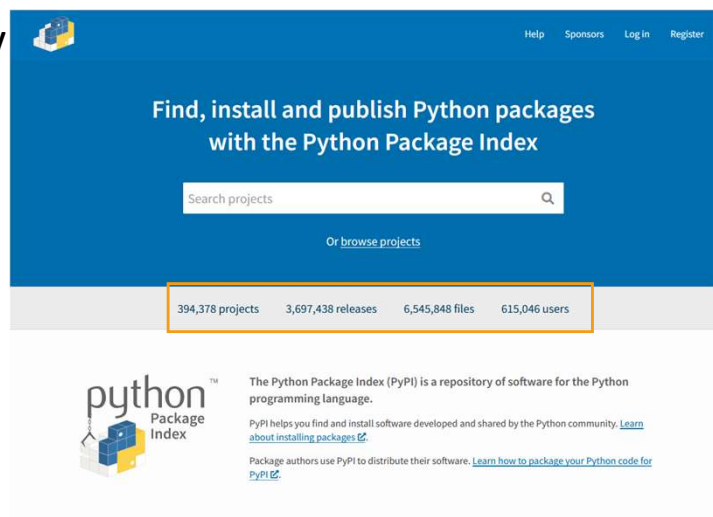


Python Pros and Cons

Python Pros



- **Comprehensive Standard Library**
- **Community-driven**
- **3rd Party Libraries**
- **3rd Party Tools**



3rd Party Libraries

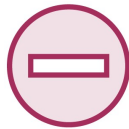


- **pypi.python.org**

- Dependencies
- Package Groups
- Versions



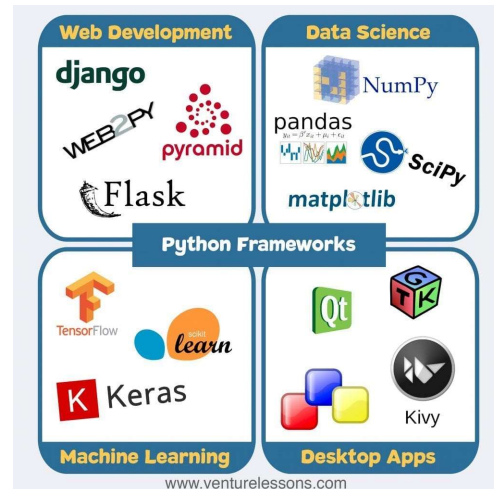
install



uninstall

```
$> pip install <PackageName>
```

```
$> pip uninstall <PackageName>
```



3rd Party Tools



IDEs:

- Pydev
- Pycharm
- Visual studio Code
- Spyder



Editors:

- Sublime
- Vim
- GNU/Emacs

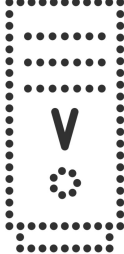
Code Tools

- Flake 8 (Style Guide Enforcement)
- PyLint (Code Analysis)
- Black (Code Formatter)
- VTune - Intel (Performance Analyzers)


Python Cons



Interpreted



Not Native



Dynamic



Lập trình Python

Mục tiêu môn học



- Kiến thức: Trang bị cho người học các kỹ năng cơ bản lập trình với ngôn ngữ Python
- Kỹ năng:
 1. Phát triển các kỹ năng cơ bản lập trình ngôn ngữ Python
 2. Khả năng thiết kế và thực thi chương trình Python để giải quyết các bài toán lập trình cơ bản
 3. Lên ý tưởng, thiết kế và giải quyết bài toán ứng dụng cụ thể (dự án)

Nội dung môn học



- Chương 1. Giới thiệu
- Chương 2. Biến và những kiểu dữ liệu đơn giản
- Chương 3. Danh sách (Lists)
- Chương 4. Câu lệnh Rẽ nhánh
- Chương 5. Từ điển (Dictionaries)
- Chương 6. Đầu vào của người dùng và vòng lặp while
- Chương 7. Hàm
- Chương 8. Lớp
- Chương 9. Tập và ngoại lệ
- Chương 10. Dự án

Tài liệu tham khảo



• Học liệu bắt buộc

- [1] Eric Matthes. Python crash course, 2nd Edition: A hands-on, project-based introduction to programming, No Starch Press, 2019.

• Học liệu tham khảo

- [2] Allen B. Downey, Think Python: How to Think Like a Computer Scientist, O'Reilly, 2015
- [3] Zed A. Shaw, Learn Python 3 the Hard Way, Addison-Wesley, 2016

Đánh giá môn học



Đánh giá	Tỷ lệ	Hình thức đánh giá
- Chuyên cần	10 %	Cá nhân
- Trung bình các điểm bài tập lớn	20% ↑	Nhóm, không quá 03 SV
- Trung bình các bài kiểm tra	10%	Trắc nghiệm và thực hành
- Kiểm tra cuối kỳ	60%	Thực hành

Phân nhóm bài tập lớn



- Bài tập lớn
 - Tìm hiểu thư viện/công cụ/framework
 - Ứng dụng
- Thời gian
 - Nhóm: ...
 - Đề tài: ...
- Nhóm fb:
 - ...
 - ...
- Lớp trưởng

Thực hành



- Thực hành online: code.ptit.edu.vn
- Lịch theo qldt.ptit.edu.vn
- >>>

DEMO



- **Install & REPL**
- **Visual Studio Code**
- **Dir**
- **Help**
- ...
- ...
- **Code**
 - Hello World
 - FizzBuzz
 - code.ptit.edu.vn
- **Đa mô hình (SP | OOP | FP)**

Q & A

