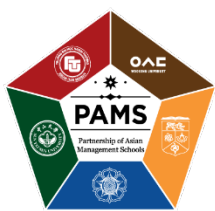


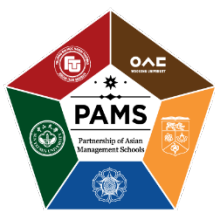
Introduction to Python

Mai Xuan Trang, Ph.D.



Brief History of Python

- Invented in the Netherlands, early 90s by Guido van Rossum
- Named after Monty Python
- Open sourced from the beginning
- Considered a scripting language, but is much more
- Scalable, object oriented and functional from the beginning
- Used by Google from the beginning
- Increasingly popular



Why Python?

- The most popular language programming
- Using in many fields, especially in data and AI

Projections of future traffic for major programming languages

Future traffic is predicted with an STL model, along with an 80% prediction interval.

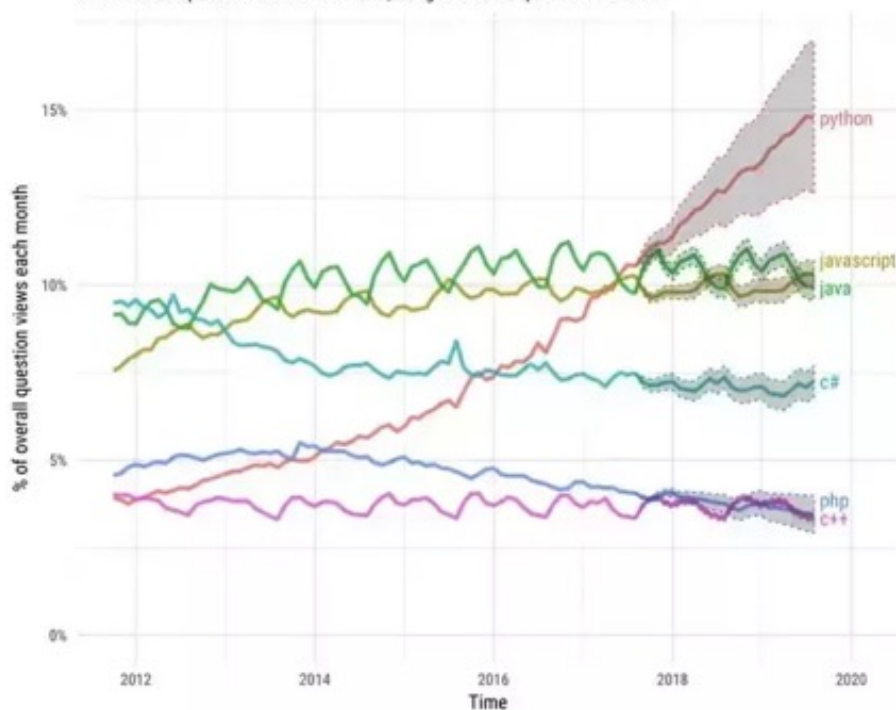
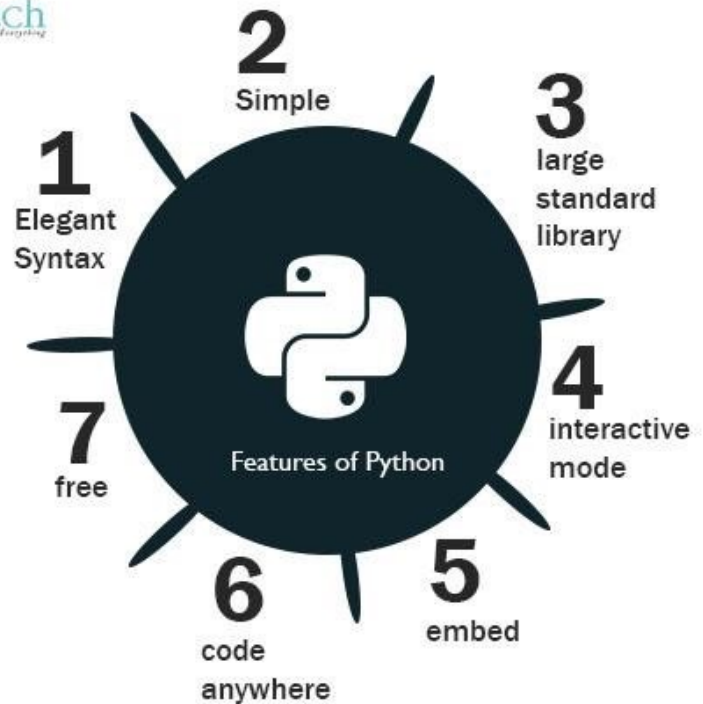
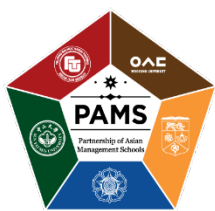


Image: Stack Overflow





Why Python?

```
#include <iostream>
int main() {
    std::cout << "Hello, world! ";
    return 0;
}
```

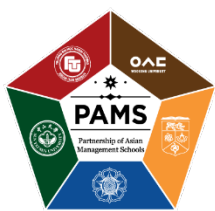
C++

```
public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Hello, World");
    }
}
```

Java

```
print("Hello, world!")
```

Python



http://docs.python.org/

3.9.1 Documentation

docs.python.org/3/

Apps Startup Machine Learning Research Invest DataSart Kindle Books Teaching Other Bookmarks

Python » English 3.9.1 Documentation » Quick search Go | modules | index

Download

Download these documents

Docs by version

Python 3.10 (in development)
Python 3.9 (stable)
Python 3.8 (stable)
Python 3.7 (security-fixes)
Python 3.6 (security-fixes)
Python 3.5 (EOL)
Python 2.7 (EOL)
All versions

Other resources

PEP Index
Beginner's Guide
Book List
Audio/Visual Talks
Python Developer's Guide

Python 3.9.1 documentation

Welcome! This is the documentation for Python 3.9.1.

Parts of the documentation:

What's new in Python 3.9?
or all "What's new" documents since 2.0

Tutorial
start here

Library Reference
keep this under your pillow

Language Reference
describes syntax and language elements

Python Setup and Usage
how to use Python on different platforms

Python HOWTOs
in-depth documents on specific topics

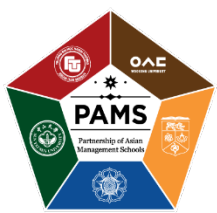
Installing Python Modules
installing from the Python Package Index & other sources

Distributing Python Modules
publishing modules for installation by others

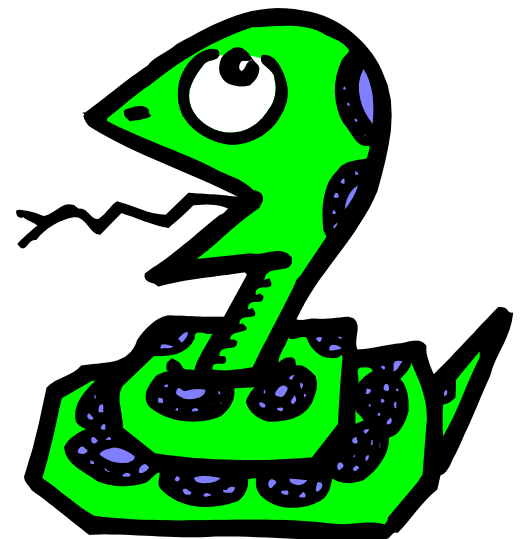
Extending and Embedding
tutorial for C/C++ programmers

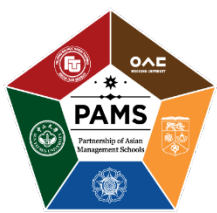
Python/C API
reference for C/C++ programmers

FAQs
frequently asked questions (with answers!)



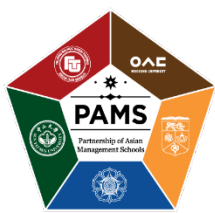
Running Python





Where should you run Python

- It's best to install it on your own computer
 - You'll have more control, can run Jupyter notebooks and learn more about it
- You can also run it on Jupyterhub unix system
- You can also use remote notebook servers: Google Colab, ...



Installing Python 3

- Using Anaconda (Miniconda): A distribution of the Python (good for AI & data science).
- [How to install](#)
- Install Anaconda (Miniconda) on your computer.
- Running it on your own computer makes it easier to install packages, IDEs, and use notebooks
- And will give you more experience

Running Interactively on Python Shell

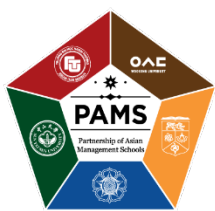
On command line tool (Terminal, Prompt,...)

```
% python
```

```
>>> 3+3
```

```
6
```

- Python prompts with '>>>'.
- To exit Python:
 - In Unix, press CONTROL-D
 - In Windows, press CONTROL-Z + <Enter>
 - run `exit()` or `quit()`



Example 'script': fact.py

```
#!/usr/bin/python
```

```
def fact(x):
```

```
    """Returns the factorial of its argument, assumed to be a posint"""
```

```
    if x == 0:
```

```
        return 1
```

```
    return x * fact(x - 1)
```

```
print ('N fact(N)')
```

```
print("-----")
```

```
for n in range(10):
```

```
    print(n, fact(n))
```

- Run the script with python interpreter

```
% python fact.py
```

Write your first Python Program



Library

Data
structure

Expression
& Control
flow

Variable

```
import sys
import random

def say_hello(user):
    # some greeting in different languages
    prefix_dict = {
        1: "Hello ",
        2: "Xin Chao ",
        3: "ni hao "
    }
    key = random.randint(1,3)
    prefix = prefix_dict[key]
    print(prefix + user)

if __name__ == "__main__":
    user = sys.argv[1]
    say_hello(user)
```

Jupyter Notebooks

Using Google Colab

Resources

- Official Website: <https://www.python.org/>
- Python tutorials for beginners:
<https://thepythonguru.com/>
- Many guides for beginners:
<https://wiki.python.org/moin/BeginnersGuide>
- Good book: Python for Data Analysis