

Python: Introdução

Álvaro Justen

a.k.a. Turicas

alvaro.justen@peta5.com.br



peta 5



pythonRio



<http://creativecommons.org/licenses/by-nc-sa/2.5/br/>

Sobre mim

- Engenharia de Telecomunicações
- UFF: CdD, PET-Tele, Meta, IF, MídiaCom, Monitoria
- Peta5:
 - Telefonia IP
 - Cidades digitais
 - TV Digital
- Software Livre
- Desenvolvedor Web



Python:
Introdução

Roteiro

- Afinal, o que é Python?
- O interpretador
- Sintaxe básica
- Controle de fluxo
- Estruturas de dados
- Funções
- Classes
- Módulos e Standard Library



Python:
Introdução

O que é Python?

- Interpretada, VHLL
- Criada por Guido van Rossum em 1989
- Multiplataforma
- OO, Procedural e funcional
- Software livre
- Baterias incluídas!
- Sintaxe elegante, simples e clara
- Tipagem **dinâmica** e **forte**



Python:
Introdução

Onde é usada?

- Empresas :
 - Google
 - NASA
 - Peta5 (. . .)
- Software :
 - Trac
 - GIMP
 - Blender
 - GNOME (. . .)



Python:
Introdução

Onde roda?

- UNIX likes:
 - GNU/Linux (maioria vem por padrão)
 - Mac OS X (sempre por padrão)
- Windows (tem que instalar)
- Nokia Series 60
- ...



Python:
Introdução

Aplicações

- Scripts
- Computação gráfica
- Web :
 - web2py :-)
 - Django
 - Zope
 - ...
- Jogos
- Programas em geral (GUI)



Python:
Introdução

História

About Google!



Links

- [Research Papers about Google and the WebBase](#)
- [Pictures and stats for the Stanford Google Hardware](#)

Credits

- Current Development: [Sergey Brin](#), [Larry Page](#), and [Craig Silverstein](#)
- Design and Implementation Assistance: [Scott Hassan](#) and [Alan Steremberg](#)
- Faculty Guidance: [Hector Garcia-Molina](#), [Rajeev Motwani](#), [Jeffrey D. Ullman](#), and [Terry Winograd](#)
- Research Funding: [NSF](#), [NASA](#), [DARPA](#) and [Interval Research](#)
- Equipment Donations: [IBM](#), [Intel](#), and [Sun](#)
- Equipment Consulting: [Penguin Computing](#)
- Software: [GNU](#), [Linux](#), [Python](#), [Parasoft](#) (debugging), and [Gimp](#) (logo design)
- Collaborating Groups in the [Computer Science Department](#) at [Stanford University](#): [The Digital Libraries Project](#), [The Project on People Computers and Design](#), [The Database Group](#), [The Stanford InfoLab](#), [The MIDAS Data Mining Group](#), and [The Theory Division](#)
- Outside Collaborators: [Interval Research Corporation](#) and the [IBM Almaden Research Center](#)
- Technical Assistance: [The Computer Science Department's Computer Facilities Group](#), [Stanford's Distributed Computing](#) and [Intra-Networking Systems Group](#)

Last modified: Sun Sep 27 20:26:46 PDT 1998



Python:
Introdução

Características e Vantagens

- Fácil de aprender
- Poderosa
- Alta produtividade
- Estruturas de dados de alto nível: tuplas, listas, dicionários, ...
- Linguagem dinâmica -> runtime!
- Blocos de código são delimitados por endentação (!?)



Python:
Introdução

Interpretador e Shell

- Interpretador interativo:
 - "python" (shell)
 - "python arquivo.py" (interpretador)
- IPython



Python:
Introdução

Tipos de dados

- Tipos básicos:
 - Inteiro: 42
 - Ponto flutuante: 3.14
 - Complexo (!): 1.5 + 5.3j
 - String: `"""minha string"""`
- Estruturas de alto nível:
 - Tuplas: (1, "aaa")
 - Listas: [1, 2.3, "abc", [1, "b"]]
 - Dicionários: {'a': 1, 2: 'b', [1, 2]: [3, 4]}



Controle de Fluxo

Nada de parênteses! Nada de chaves!

if expressão:

Comando1

...

elif expressão2:

Comando3

...

else:

...



Python:
Introdução

Controle de Fluxo (2)

```
for i in iterable:
```

```
    Comando1
```

```
    Comando2
```

```
    ...
```

```
while expressão2:
```

```
    Comando3
```

```
    ...
```

```
iterable → list, range(), dict, ...
```

```
break, continue, pass, for ... else
```



Python:
Introdução

Funções

```
def nome(args):
```

```
    Comando1
```

```
    ...
```

```
    [return X]
```

- *args
- **args
- arg1=valor, arg2=valor, ...
- lambda
- docstrings



Python:
Introdução

Módulos

- `arq.py`
- `import arq`
- `arq.X`
- `arq.f()`
- `f2 = arq.f`
- `arq.__name__`
- `from arq import X, f`
- `from arq import *`



Python:
Introdução

Arquivos

- `open('nome', 'formato')`
 - `formato = rw, r, ...`
- `fp.`
 - `read()`
 - `readlines()`
 - `write()`
 - `writelines()`
 - `close()`
- `for l in fp: print l`



Classes

```
class Nome:
```

```
    """Essa é a minha classe...  
    que não faz muita coisa"""
```

```
    self.atributo = valor
```

```
    def método(self, args...):
```

```
        cmd...
```



Python:
Introdução

Classes (2)

- `__init__()`
- Overload of operators
- Métodos fora das classes:

```
def f1(self, a, ...):  
    ...  
class teste:  
    f = f1  
    def f2(self, b, ...): ...
```
- Herança [múltipla]

Baterias incluídas

- `sys`, `os`, `re`, `string`, `zlib`
- `socket`, `urllib`, `httpplib`, `imaplib`
- `time`, `datetime`, `math`, `random`
- `doctest`, `unittest`, `xml`, `xmlrpclib`
- `struct`, `pickle`, `cPickle`, `threading`
- `logging`, `decimal`
- ...



Python:
Introdução

Outras Bibliotecas

- NumPy
- SciPy
- Matplotlib
- PyODE
- Python-OpenGL
- Twisted Matrix
- . . .



Python:
Introdução

Python na prática



Python:
Introdução

The Zen of Python

```
>>> import this
```



Python:
Introdução

www.python.org

- PEPs
- Python Library Reference
- Python Manual Reference
- PyPI
- Implementações:
 - CPython
 - Jython
 - IronPython
 - PyPy



Python:
Introdução

Sites interessantes

Pyth0nRio:

<http://br.groups.yahoo.com/group/pythonrio/>

Python Brasil:

<http://www.python.org.br/>

web2py:

<http://www.web2py.com/>

web2py-users-brazil:

[http://groups.google.com/group/
web2py-users-brazil/](http://groups.google.com/group/web2py-users-brazil/)



Python:
Introdução