



# PYTHON 101

Julián Perelli  
@jperelli

Cualbondi  
Mainstorconcept  
Docente UTN FRLP

A close-up photograph of a snake's head, showing its blue and black scales and a single eye. The snake is positioned on the left side of the frame, with its head angled towards the right. The background is a dark, textured surface, possibly a wall or a piece of fabric, which is out of focus.

# Historia



# Guido van Rossum



## Python's Benevolent Dictator for Life

“In December 1989, I was looking for a “hobby” programming project that would keep me occupied during the week around Christmas”



# Guido van Rossum



“I decided to write an interpreter for the new scripting language I had been thinking about lately: a descendant of ABC that would appeal to Unix/C hackers.”



# Guido van Rossum



- CWI, Centrum Wiskunde & Informatica
- Google
- Dropbox



# Monty python







# Filosofía



# The zen of python

Beautiful is better than ugly.  
Explicit is better than implicit.  
Simple is better than complex.  
Complex is better than complicated.  
Flat is better than nested.  
Sparse is better than dense.  
Readability counts.  
Special cases aren't special enough to break the rules.  
Although practicality beats purity.  
Errors should never pass silently.  
Unless explicitly silenced.  
In the face of ambiguity, refuse the temptation to guess.  
There should be one-- and preferably only one --obvious way to do it.  
Although that way may not be obvious at first unless you're Dutch.  
Now is better than never.  
Although never is often better than \*right\* now.  
If the implementation is hard to explain, it's a bad idea.  
If the implementation is easy to explain, it may be a good idea.  
Namespaces are one honking great idea -- let's do more of those!



# The zen of python

Beautiful is better than ugly.

Simple is better than complex.

Flat is better than nested.

**Readability counts.**



# Fanatismo

**Premature optimization vs clarity**

**Pythonic**

**Pythonista**





# Features



# Multiparadigma

(Propósito general)

Imperativo + procedural

Orientado a Objetos

Funcional

- map / reduce / filter /comprehensions / generators

Orientado a aspectos



# Features

## Tipado dinámico

- No se declaran variables
- Las variables pueden cambiar de tipo en tiempo de ejecución

## Garbage collector

## Full unicode (utf8) support



A close-up photograph of a snake's head, showing its blue and black scales and a single eye. The snake is positioned on the left side of the frame, with its head angled towards the right. The background is a dark, textured surface.

# Sintaxis



# Sintaxis: Expresiones

```
a = 1  
b = a + 2  
a = "Hola"  
b = " Mundo"  
print a + b
```



# Sintaxis: Convenciones

There are two types of people.

```
if (Condition)
{
    Statements
    /*
     ...
    */
}
```

```
if (Condition) {
    Statements
    /*
     ...
    */
}
```

Programmers will know.



# Sintaxis: Convenciones

```
if (condition) {  
    /* statements */  
}
```

```
if (condition)  
{  
    /* statements */  
}
```

```
if (condition)  
{  
    /* statements */  
}
```

```
if (condition)  
{  
    /* statements */  
}
```

```
if (condition){  
    /* statements */  
}
```

```
if (condition) {  
    /* statements */  
}
```

```
if (condition)  
{  
    /* statements */  
}
```

```
if (condition) { /* statements */ }
```

```
if (condition){  
    /* statements */  
}
```



# Sintaxis: Convenciones

## Tabs vs Espacios



# Sintaxis: Convenciones

## Tabs vs Espacios





# Sintaxis: Control

```
if a < b:  
    print "a es"  
    print "menor"  
else:  
    print "a es mayor o igual"
```



# Sintaxis: Loop

```
for e in [1, 5, "pepe"]:  
    print e
```



# Sintaxis

```
void bubble_sort (int *a, int n) {  
    int i, t, s = 1;  
    while (s) {  
        s = 0;  
        for (i = 1; i < n; i++) {  
            if (a[i] < a[i - 1]) {  
                t = a[i];  
                a[i] = a[i - 1];  
                a[i - 1] = t;  
                s = 1;  
            }  
        }  
    }  
}
```

```
def bubble_sort(a):  
    s = True  
    while s:  
        s = False  
        for i in xrange(len(a) - 1):  
            if a[i] > a[i+1]:  
                a[i], a[i+1] = a[i+1], a[i]  
                s = True  
    return None
```



# Sintaxis

```
void bubble_sort (int *a, int n) {  
    int i, t, s = 1;  
    while (s) {  
        s = 0;  
        for (i = 1; i < n; i++) {  
            if (a[i] < a[i - 1]) {  
                t = a[i];  
                a[i] = a[i - 1];  
                a[i - 1] = t;  
                s = 1;  
            }  
        }  
    }  
}
```

```
def bubble_sort(a):  
    s = True  
    while s:  
        s = False  
        for i in xrange(len(a) - 1):  
            if a[i] > a[i+1]:  
                a[i], a[i+1] = a[i+1], a[i]  
                s = True  
    return None
```



# Sintaxis ofuscada

```
void bubble_sort (int *a, int n) {int i, t, s = 1;while (s) {s = 0;
for (i = 1; i < n; i++) {if (a[i] < a[i - 1]) {t = a[i];a[i] = a[i - 1];a[i - 1] = t;s =
1;}}}}
```

```
def bubble_sort(a):
    s = True
    while s:
        s = False
        for i in xrange(len(a) - 1):
            if a[i] > a[i+1]:
                a[i], a[i+1] = a[i+1], a[i]
                s = True
    return None
```

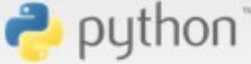


A close-up photograph of a snake's head, showing its blue and black scales and a single eye. The snake is positioned on the left side of the frame, with its head angled towards the right. The background is a dark, textured surface.

# Comunidad



# Comunidad: Paquetes



» Package Index

PACKAGE INDEX »

- Browse packages
- Package submission
- List trove classifiers
- List packages
- RSS (latest 40 updates)
- RSS (newest 40 packages)
- Python 3 Packages
- PyPI Tutorial
- PyPI Security
- PyPI Support
- PyPI Bug Reports
- PyPI Discussion
- PyPI Developer Info

ABOUT »

NEWS »

DOCUMENTATION »

DOWNLOAD »

COMMUNITY »

FOUNDATION »



CORE DEVELOPMENT »

## PyPI - the Python Package Index

The Python Package Index is a repository of software for the Python programming language. There are currently **60504** packages here.

To contact the PyPI admins, please use the [Support](#) or [Bug reports](#) links.

**Not Logged In**

[Login](#)  
[Register](#)  
[Lost Login?](#)  
Use [OpenID](#)  

**Status**

[Nothing to report](#)

**Get Packages**

To use a package from this index either "pip install package" ([get pip](#)) or download, unpack and "python setup.py install" it.

**Package Authors**

Submit packages with "[python setup.py upload](#)". The index [hosts package docs](#). You may also use the [web form](#). You must [register](#). Testing? Use [testpypi](#).

**Infrastructure**

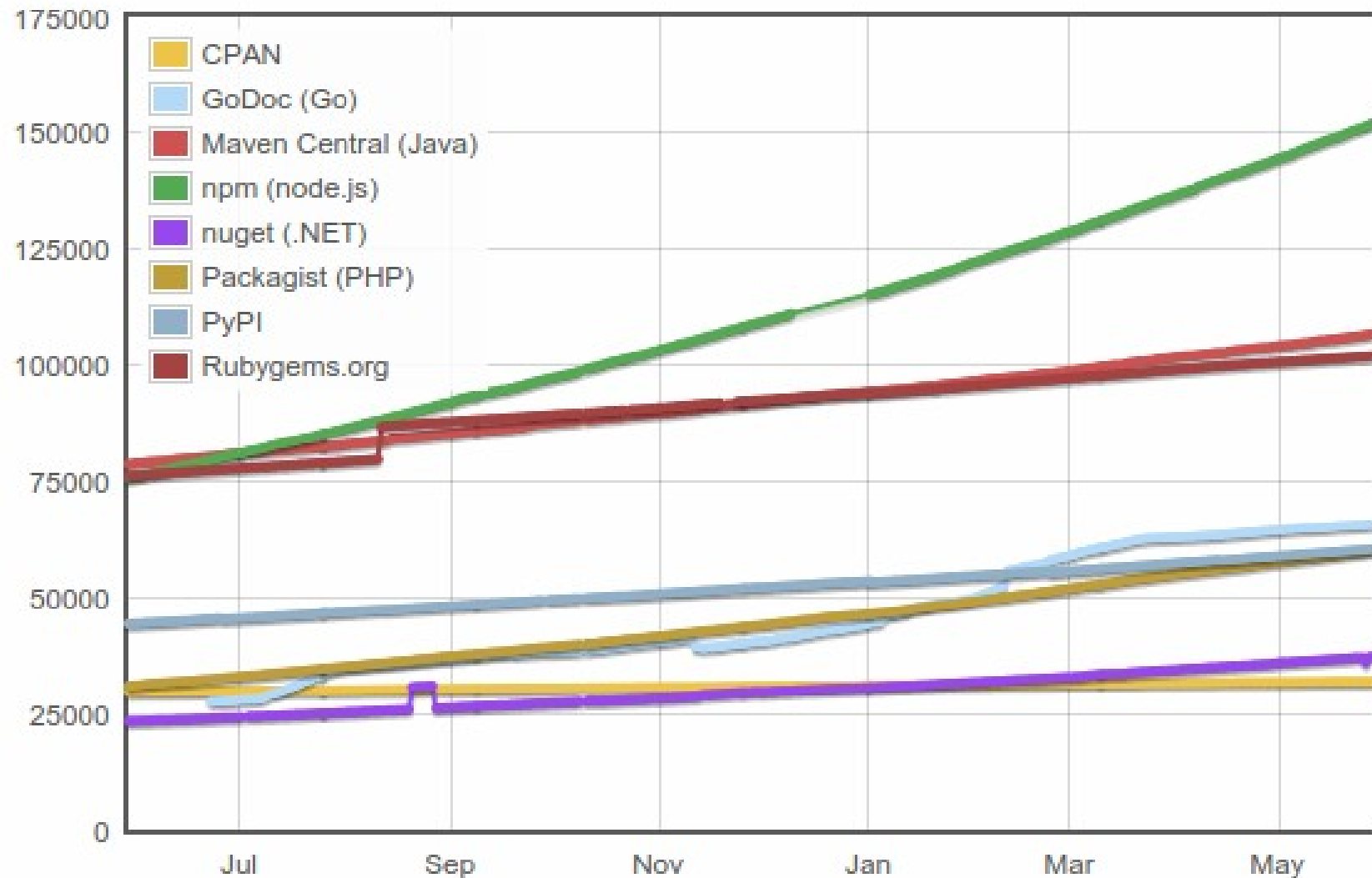
To interoperate with the index use the [JSON](#), [OAuth](#), [XML-RPC](#) or [HTTP](#) interfaces. Use [local mirroring](#) or [caching](#) to make installation more robust.

Updated	Package	Description
2015-05-29	<a href="#">downhill 0.1.0</a>	Stochastic optimization routines for Theano
2015-05-29	<a href="#">oletools 0.11</a>	Python tools to analyze security characteristics of MS Office and OLE files (also called Structured Storage, Compound File Binary Format or Compound Document File Format), for Malware Analysis and Incident Response #DFIR
2015-05-29	<a href="#">synccloud-owncloud 8.0.3.67</a>	ownCloud app
2015-05-29	<a href="#">charlatan 0.4.5</a>	Efficiently manage and install data fixtures
2015-05-29	<a href="#">xonsh 0.1.6</a>	an exotic, usable shell
2015-05-29	<a href="#">prismh.core 0.1.1</a>	Parsing and Validation library for PRISMH Files
2015-05-29	<a href="#">generateDS 2.16a0</a>	Generate Python data structures and XML parser from Xschema
2015-05-29	<a href="#">greatjob 1.1.1</a>	Congratulations
2015-05-29	<a href="#">python-jsonrpc 0.7.7</a>	Python JSON-RPC Client Server Library - Simple To Use Python JSONRPC-Library
2015-05-29	<a href="#">python-zpar 0.9.3</a>	A Wrapper around the ZPar statistical tagger/parser for English
2015-05-29	<a href="#">PyScreeze 0.1.7</a>	A simple, cross-platform screenshot module for Python 2 and 3.
2015-05-29	<a href="#">django-cas-client 1.2.0</a>	Django Cas Client
2015-05-29	<a href="#">rebound 0.4.2</a>	An open-source multi-purpose N-body code
2015-05-29	<a href="#">django-uuslug 1.1.5</a>	A Django slugify application that guarantees uniqueness and handles unicode.
2015-05-29	<a href="#">django-activity 1.2.0</a>	Activity app for django



# Comunidad: Paquetes

## Module Counts





**Me voy a node.js?**



# Me voy a node.js?

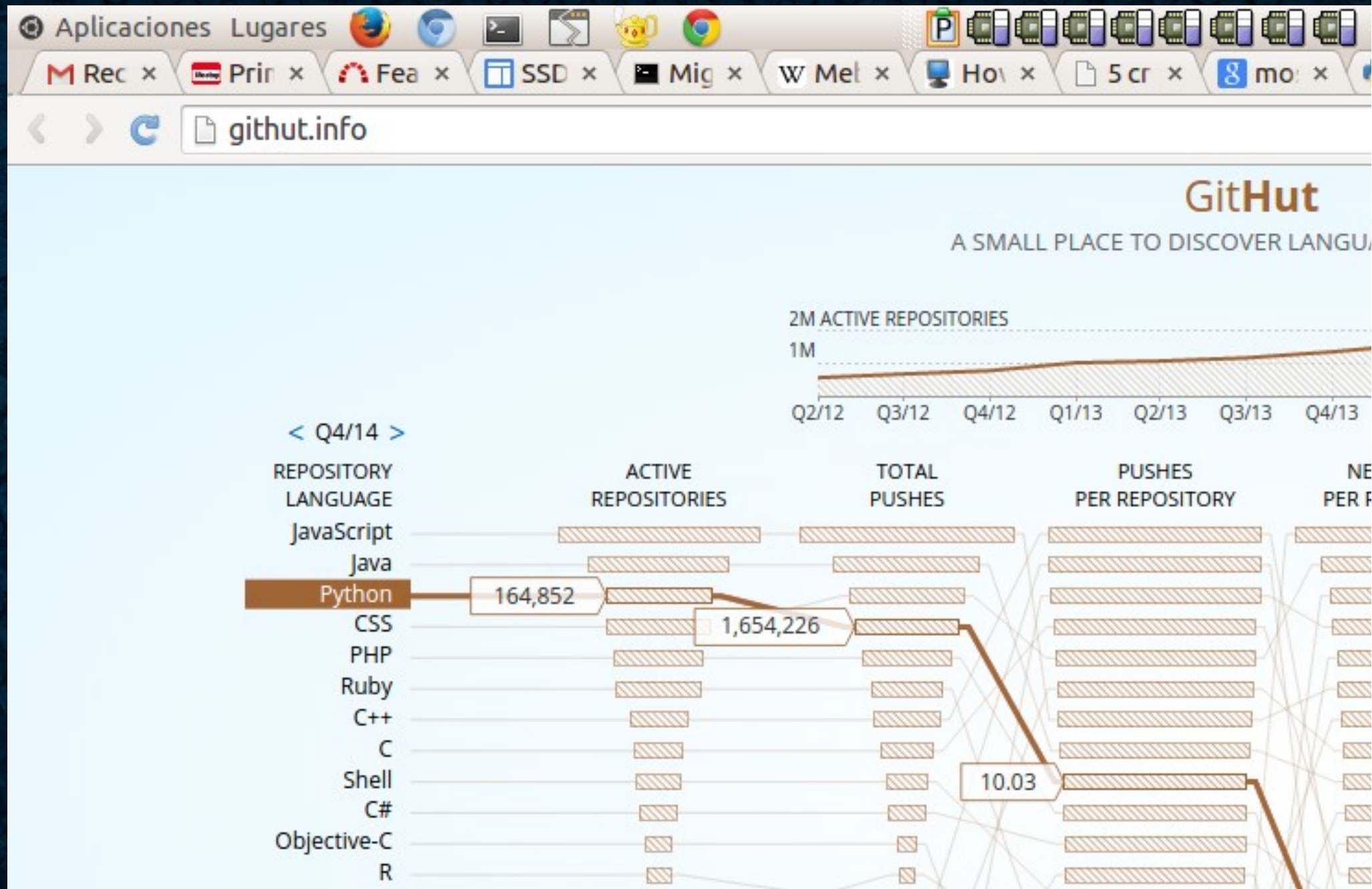
NO

Diferentes propósitos

Node's early packaging



# Comunidad: Github





# Quién lo usa

Infraestructura (scripting linux)

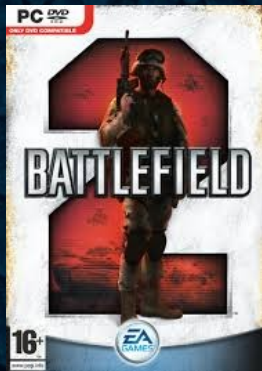
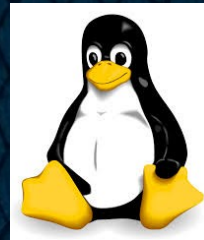
Web sites

Math experts

Standalone PC apps



# Quien lo usa





Quien no lo usa?



Quien no lo usa?

¿Qué esperarás?





**Gracias!**