

# Python en Educación

Diego Cañizares



# Indice

- Introducción
- ¿Qué es Python?
- ¿Por qué Python?
- ¿Cómo implementar Python?
- Casos de éxito
- Lecturas recomendadas
- ¿Eso es todo?



# Introducción

- ¿Quién te conoce?
- ¿De qué se trata todo esto?



# ¿Quién te conoce?

- Nombre: Diego Cañizares
- E-Mail: dcanizares <at> bynarius.com
- Blog: <http://www.diegocanizares.com.ar>
- Participo en:
  - CaFeLUG
  - PyAr
  - Gleducar
- ~~Robando~~ Trabajando en:
  - Owner en Bynarius
  - Arquitecto de Software en EMMSA Consulting
  - Estudiante de Análisis de Sistemas
- Estudiante de Análisis de Sistemas



# ¿De qué se trata todo esto?



+



Jornadas del Sur

Diego Cañizares

# ¿Qué es Python?

- Algunas características
- Propiedades del lenguaje



# Algunas características

- Gratis **Y** Libre
- Maduro (+14 años)
- **Fácil** de aprender
- Extremadamente **portable**
- Multipropósito



# Propiedades del lenguaje

- Compila a bytecode interpretado
- **Multi-paradigma**
- Se pone las pilas! (o mejor, **baterias**)
- Manejo moderno de **errores**
- Tipos de datos de **alto nivel**
- Intérprete interactivo



# ¿Por qué Python?

- Creado para problemas reales
- Características de otros lenguajes
- Menos masoquista, ouch!
- Divertido, pero también productivo
- Equilibrio práctico-conceptual
- Permite enfocarse en el problema
- Fácil de aprender y de enseñar
- Simplifica ejemplos de programación



# Ejemplo de Programación - C++

```
#include <iostream.h>
void main()
{
    cout << "Hola, mundo" << endl;
}
```



# Ejemplo de Programación - Python

```
print "Hola, Mundo"
```



# ¿Cómo implementar Python?

- Correcto equilibrio entre:
  - Conceptos importantes
  - Material interesante
  - Abstracción matemática
- Evitar frustración por problemas:
  - Sintaxis
  - Compilación



# ¿Cómo implementar Python?

- Estudiantes inspirados:
  - Resolviendo problemas interesantes
  - Explorando ideas
  - Probando enfoques no convencionales
  - Rompiendo reglas (pero no PCs :))
  - Aprendiendo de sus errores
- Pero, evitando perder tiempo por:
  - Sintaxis compleja
  - Mensajes de error incomprensibles
  - 1<sup>^</sup>100 maneras de generar General Protection Fault



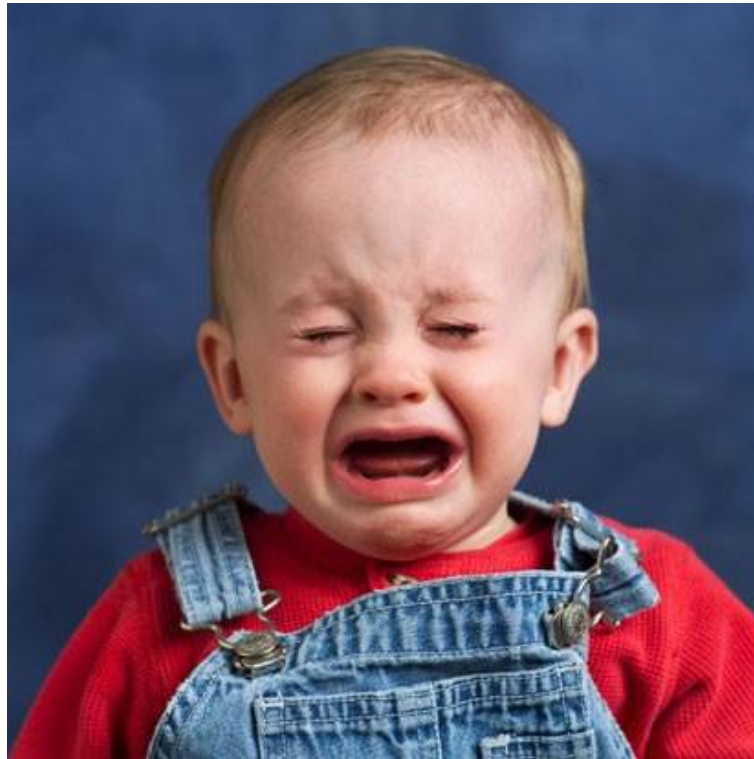
# Casos de éxito

- Escuela de Oficios San Cayetano
  - Curso de Programador de Computadoras
- Universidad de Chicago
  - Introduction to Computer Science 2
- Escuela Secundaria Yorktown, Virginia
  - Computer Science 1
- Universitat Jaume, España
  - Introducción a la Programación
- MIT
  - A Gentle Introduction to Programming Using Python

# Lecturas recomendadas

- Python
  - <http://www.python.org>
- EDU-SIG (Education Special Interest Group)
  - <http://www.python.org/community/sigs/current/edu-sig>
- Aprenda a pensar como un Programador con Python
  - <http://manuales.gfc.edu.co/python/tlc>
- PyBiblio
  - <http://openbookproject.net/pybiblio/>
- Snake Wrangling for Kids
  - <http://www.briggs.net.nz/log/writing/snake-wrangling-for-kids/>
- Promoting computer literacy through programming Python
  - <http://www.python.org/files/miller-dissertation.pdf>

# ¿Eso es todo?



## Lamentablemente, SI



Jornadas del Sur

Diego Cañizares



# ¿Eso es todo?



**Pero pueden hacer preguntas!**



# Muchas Gracias

Diego Cañizares





Este documento se distribuye  
bajo la siguiente licencia

[Creative Commons Argentina](#)  
[Atribución - NoComercial 2.5](#)

Algunos derechos reservados.