

¿POR QUÉ QUEREMOS TANTO A PYTHON?



Milena Armada, 6 de Abril de 2019

CHARLA INSPIRACIONAL VS. CHARLA TÉCNICA

CHARLA INSPIRACIONAL VS. CHARLA TÉCNICA



I ❤️ PYTHON

BÍO E HISTORIA

- Ingeniera en sistemas de información UTN
- 20 años de experiencia
- Fierro
- RadioCut

¿CON QUÉ TECNOLOGÍA EMPEZAMOS UN PROYECTO?



¿POR QUÉ QUEREMOS TANTO A PYTHON?

1. FÁCIL DE LEER

IMPORTANCIA DE LA LEGIBILIDAD

- Al momento de aprender
- Cuando leemos código (¡siempre!)

INDENTACIÓN COMO PARTE DEL LENGUAJE

```
1 class Producto:
2
3     def tieneDisponible(self):
4         if self.stock >= self.reservado:
5             return True
6         else:
7             return False
```

GUÍA DE ESTILO - PEP8

```
1 # Si
2 x = 1
3 y = 2
4 long_variable = 3
5
6 # No:
7 x          = 1
8 y          = 2
9 long_variable = 3
```

IDIOMS

```
1 # No pythónico
2 auth = None
3 if 'auth_token' in payload:
4     auth = payload['auth_token']
5 else:
6     auth = 'Unauthorized'
7
8 # Pythónico
9 auth = payload.get('auth_token', 'Unauthorized')
```

DINÁMICO

```
1 count = 5
2
3 def sumar(a, b):
4     return a + b
```

2. RÁPIDO PARA DESARROLLAR Y PROTOTIPAR

LENGUAJE INTEPRETADO

- No se compila
- Acelera el tiempo de desarrollo
- Intérprete interactivo

INTÉRPRETE INTERACTIVO

```
milena@mexico:~ $ python
Python 2.7.12 (default, Nov 12 2018, 14:36:49)
[GCC 5.4.0 20160609] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> edades = [34, 13, 45, 67, 9]
>>> mayores = [e for e in edades if e >= 18]
>>> mayores
[34, 45, 67]
>>> █
```

Permite programar, probar, explorar y leer documentación

JUPYTER

Aplicación web que permite crear documentos que ejecutan código, muestra código narrativo y visualizaciones.


```
In [ ]: from colofon.client.cli.fierroCLILauncher import login
        from colofon.client.model.managers import BookManager
        from colofon.client.model.bookFilters import TitleFilter
```

```
In [ ]: login('PYROL0C://localhost:8002', 'fierro', 'a', 'SUC1')
```

```
In [ ]: books = BookManager.queryItems(TitleFilter("matem est"))
```

```
In [ ]: print len(books)
```

```
In [ ]: for book in books:
        print book.title, book.getIsbn()
```

3. ORIENTADO A OBJETOS Y MULTIPARADIGMA

- Programación orientada a objetos
 - Asignación de responsabilidades
 - Encapsulamiento
 - Herencia
- Programación procedural
 - En la práctica poder escribir funciones es útil

4. MADURO Y OPEN SOURCE

- 25 años
- Diseño elegante y robusto
- Evoluciona - Python 3
- Todos podemos mirar y aportar

5. "BATERÍAS INCLUÍDAS"

LIBRERÍA STANDARD

Servicios del sistema, fecha y hora, subprocesos, sockets, internacionalización y localización, base de datos, threads, formatos zip, bzip2, gzip, tar, expresiones regulares, XML (DOM y SAX), Unicode, SGML, HTML, XHTML, XML-RPC (cliente y servidor), email, manejo asincrónico de sockets, clientes HTTP, FTP, SMTP, NNTP, POP3, IMAP4, servidores HTTP, SMTP, herramientas MIME, interfaz con el garbage collector, serializador y deserializador de objetos, debugger, profiler, random, curses, logging, compilador, decompilador, CSV, análisis lexicográfico, interfaz gráfica incorporada, matemática real y compleja, criptografía (MD5 y SHA), introspección, unit testing, doc testing, etc., etc...

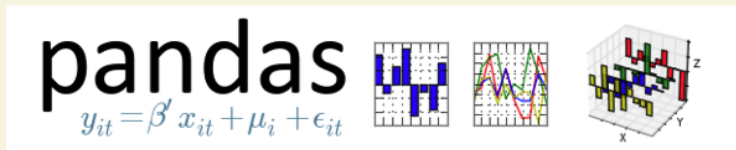
MÁS BATERÍAS • Web
• Data Science
• Bases de datos
• Etc, etc.

CLAVE EN LA PRODUCTIVIDAD DE PYTHON

DATA SCIENCE



IP[y]: IPython
Interactive Computing



6. LA COMUNIDAD

MILES DE PROGRAMADORES EN TODO EL MUNDO INTERACTUANDO EN FORMA FÍSICA Y VIRTUAL

- Python Software Foundation
- PyCon

