

Lenguaje de Programación Python

Introducción

Dr. Mario Marcelo Berón

Universidad Nacional de San Luis
Departamento de Informática
Área de Programación y Metodologías de Desarrollo de Software



- 1 Historia
- 2 Características y Filosofía
- 3 El Intérprete de Python





- Fue pensado en 1980 por Guido van Rossum en Centrum Wiskunde & Informática en Los Países Bajos.
- Fue el sucesor del lenguaje ABC capaz de manejar excepciones e interactuar con Amoeba.
- La implementación del lenguaje comenzó en 1989.
- Guido van Rossum fue el encargado de desarrollar el proyecto hasta el 12 de julio de 2018 donde anunció que entra en Vacaciones Permanentes.





- Python 2.0 - 16 de octubre de 2000 - Recolector de basura que detecta ciclos y soporte para Unicode.
- Python 3.0 - 3 de diciembre de 2008. Es una revisión del lenguaje que no es completamente compatible con las anteriores.





- Lenguaje Multiparadigma - Imperativo, Funcional, Orientado a Objetos.
- Tipado Dinámico.
- Usa recolector de basura.
- El soporte funcional es similar al de Lisp.





El zen de python es una colección de 19 principios para escribir programas. Estos principios fueron escritos por Tim Peters. 1999.

- Bello es mejor que feo.
- Explícito es mejor que implícito.
- Simple es mejor que complejo.
- Complejo es mejor que complicado.
- Plano es mejor que anidado.
- Espaciado es mejor que denso.





- La legibilidad es importante.
- Los casos especiales no son lo suficientemente especiales como para romper las reglas.
- Sin embargo la practicidad le gana a la pureza.
- Los errores nunca deberían pasar silenciosamente.
- A menos que se silencien explícitamente.
- Frente a la ambigüedad, evitar la tentación de adivinar.

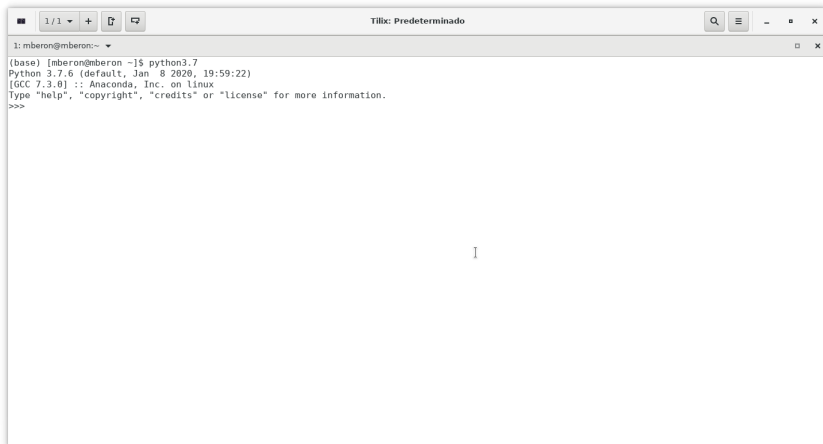




- Debería haber una, y preferiblemente solo una, manera obvia de hacerlo.
- A pesar de que eso no sea obvio al principio a menos que seas Holandés.
- Ahora es mejor que nunca.
- A pesar de que nunca es muchas veces mejor que *ahora* mismo.
- Si la implementación es difícil de explicar, es una mala idea.
- Si la implementación es fácil de explicar, puede que sea una buena idea.
- Los espacios de nombres son una gran idea, ¡tenemos más de esos!



Python en la Terminal



```
1/1 + [ ] [ ]  
Tilix: Predeterminado [ ] [ ] [ ] [ ] [ ]  
1: mberon@mberson:~  
(base) [mberon@mberson ~]$ python3.7  
Python 3.7.6 (default, Jan 8 2020, 19:59:22)  
[GCC 7.3.0] :: Anaconda, Inc. on linux  
Type "help", "copyright", "credits" or "license" for more information.  
>>>
```



Python: Calculadora

```
>>>2 + 2
```

```
4
```

```
>>>50 - 5 * 6
```

```
20
```

```
>>>50 - 5 * 8 / 4
```

```
10
```

```
>>>8 / 5
```

```
1.6
```



Python: Calculadora

```
>>> "doesn't" #o de lo contrario usa comillas dobles
"doesn't"
>>> '"Si," _le_dijo.'
'"Si," _le_dijo.'
>>> "\"Si,\" _le_dijo."
'"Si," _le_dijo.'
>>> '"Isn\'t," _she_said.'
'"Isn\'t," _she_said.'
```

