



Universidad Nacional Autónoma de México Facultad de Ingeniería

Actividad:

#5 Lunes

Alumno: Emiliano Martínez Angel Fecha: 28/07/2021

Características

Las **características** del lenguaje de programación Python se resumen a continuación:

- Es un **lenguaje interpretado**, **no compilado**, usa **tipado dinámico**, fuertemente tipado.
- Es **multiplataforma**, lo cual es ventajoso para hacer ejecutable su código fuente entre varios sistemas operativos.
- Es un lenguaje de programación **multiparadigma**, el cual soporta varios paradigmas de programación como **orientación a objetos**, **estructurada**, **programación imperativa** y, en menor medida, **programación funcional**.
- En Python, el formato del código (p. ej., la indentación) es estructural.

Fuertemente tipado

El **fuertemente tipado** significa que el tipo de valor no cambia repentinamente. Un **string** que contiene solo dígitos no se convierte mágicamente en un número. Cada cambio de tipo requiere una conversión explícita. A continuación un ejemplo de este concepto:

```
# se usa el metodo int() para convertir a entero
total = valor1 + int(valor2)

# se usa el metodo str() para convertir a cadena
print ("El total es: " + str(total))
```

Tipado dinámico

El **tipado dinámico** significa que los objetos en tiempo de ejecución (valores) tienen un tipo, a diferencia del tipado estático donde las variables tienen un tipo. A continuación un ejemplo de este concepto:

```
print (variable, type(variable))

# "variable" guarda un valor string

variable = "activo"

print (variable, type(variable))
```

Multiplataforma

Python es [multiplataforma](#), lo cual es ventajoso para hacer ejecutable su código fuente entre varios sistema operativos, eso quiere decir, soporta las siguientes plataformas para su ejecución:

Filosofía «Incluye baterías»

- Python ha mantenido durante mucho tiempo esta filosofía de «baterías incluidas»:

«Tener una biblioteca estándar rica y versátil que está disponible de inmediato. Sin que el usuario descargue paquetes separados.»

- Esto le da al lenguaje una ventaja en muchos proyectos.
- Las «baterías incluidas» están en la [librería estándar Python](#).

Zen de Python

Es una colección de 20 principios de software que influyen en el diseño del Lenguaje de Programación Python, de los cuales 19 fueron escritos por *Tim Peters* en junio de 1999. El texto es distribuido como dominio público.

El *Zen de Python* está escrito como la entrada informativa número 20 de las propuestas de mejoras de Python (*Python Enhancement Proposals - PEP*), y se puede encontrar en el sitio oficial de Python.

Los principios están listados a continuación:

- 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.
- Disperso es mejor que denso.
- La legibilidad cuenta.
- Los casos especiales no son tan especiales como para quebrantar las reglas.
- Lo práctico gana a lo puro.
- Los errores nunca deberían dejarse pasar silenciosamente.
- A menos que hayan sido silenciados explícitamente.
- Frente a la ambigüedad, rechaza la tentación de adivinar.
- Debería haber una -y preferiblemente sólo una- manera obvia de hacerlo.
- Aunque esa manera puede no ser obvia al principio a menos que usted sea holandés.
- Ahora es mejor que nunca.
- Aunque nunca es a menudo mejor que ya 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 (namespaces) son una gran idea ¡Hagamos más de esas cosas!



