

# GUÍAS DE ESTILO PARA PYTHON



## Integrantes:

Melissa Vélez Martínez #A00827220  
Juan Daniel Pérez Berlanga #A00830152  
Ian Javier Duran Román #A00829799  
Joaquín Amaya Elizondo #A01721028  
Diego Alejandro Torres Cruz #A01283874  
Alejandro Daniel Dennis Hernandez #A00831138

## 1 DISEÑO DEL CÓDIGO

Recomendaciones:

- Utilizar 4 espacios para la indentación.
- Las líneas deben de tener una longitud máxima de 79 caracteres.
- Los imports deben de realizarse uno por línea.
- Debe de existir un line-break antes de un operador binario, no después.



## 2 COMILLAS PARA STRINGS

- En Python existen comillas simples y dobles
- Se recomienda escoger un tipo de comilla para crear un string.
- Permite tener consistencia



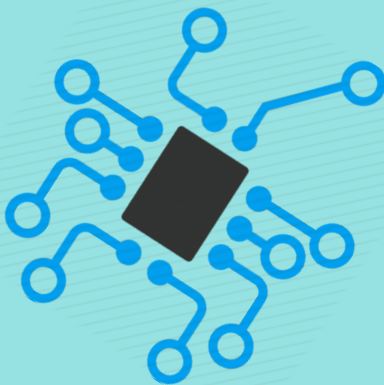
## 3 ESPACIOS EN BLANCO

No se recomienda utilizar en los siguientes casos:

- Dentro de paréntesis, corchetes o llaves
- Entre una coma y un paréntesis al final
- Antes de una coma, punto y coma o dos puntos
- Antes del paréntesis para llamar una función

## 4 COMAS FINALES

- Las comas finales suelen ser opcionales, excepto que son obligatorias al hacer una tupla de un elemento.
- Cuando las comas finales son redundantes, a menudo son útiles cuando se utiliza un sistema de control de versiones
- No tiene sentido tener una coma al final en la misma línea que el delimitador de cierre.



## 5 COMENTARIOS

- Utilice los comentarios en línea con moderación.
- Actualiza los comentarios cuando se cambie el código.
- Los comentarios generalmente se aplican al código que le sigue y están al mismo nivel que el código.
- Asegúrese de que sus comentarios sean claros y fácilmente comprensibles.



## 6 CONVENCIONES DE NOMBRES

- Los nombres que puede ver el usuario en el "API" deben reflejar el uso en lugar de la implementación.
- Existen muchos estilos de nomenclatura diferentes por lo que es útil poder reconocer qué estilo de nomenclatura se está utilizando.
- Los identificadores utilizados en la biblioteca estándar deben ser compatibles con ASCII
- Los módulos deben tener nombres cortos en minúsculas

## 7 RECOMENDACIONES DE PROGRAMACIÓN

- El código debe escribirse sin perjudicar a otras funciones de Python.
- Las comparaciones nunca se hacen con operadores de igualdad.
- Las anotaciones para variables de nivel de módulo, variables de clase locales deben tener un solo espacio después de ":"
- Utilice siempre una declaración "def" en lugar de una declaración de asignación que vincule una expresión lambda directamente a un identificador.

