



Investigación

362



5 DE NOVIEMBRE DE 2023
CUEVAS ALVARADO AXEL XAVIER

¿QUE ES PEP8?

La PEP8 es una guía que indica las convenciones estilísticas a seguir para escribir código Python. Se trata de un conjunto de recomendaciones cuyo objetivo es ayudar a escribir código más legible y abarca desde cómo nombrar variables, al número máximo de caracteres que una línea debe tener.

De acuerdo con Guido van Rossum, el código es leído más veces que escrito, por lo que resulta importante escribir código que no sólo funcione correctamente, sino que además pueda ser leído con facilidad. Esto es precisamente lo que veremos en este artículo.

Code is read much more often than it is written, Guido van Rossum

Dos mismos códigos pueden realizar lo mismo funcionalmente, pero si no se siguen unas directrices estilísticas, se puede acabar teniendo un código muy difícil de leer. Los problemas más frecuentes suelen ser:

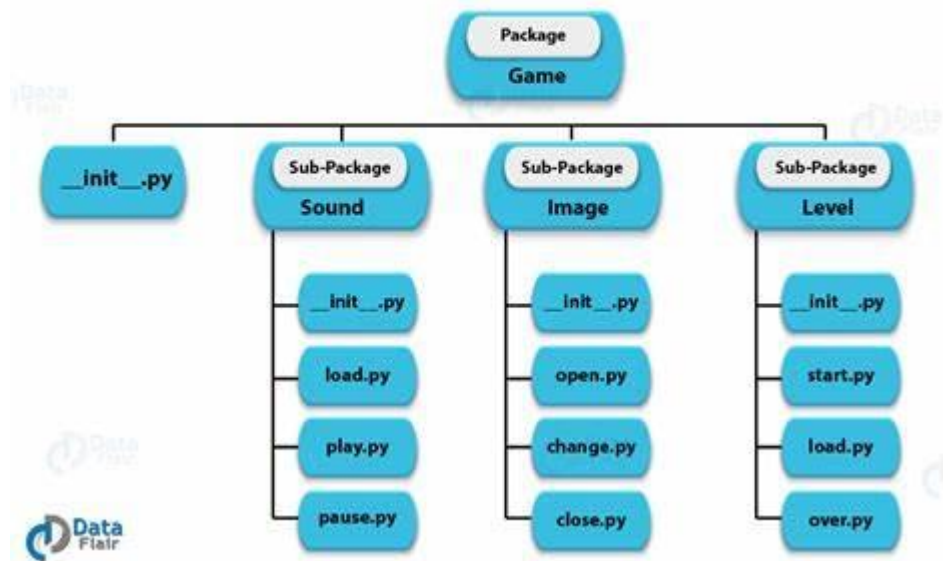
- Líneas demasiado largas.
- Nombres de variables poco explicativos.
- Código mal comentado.
- Uso incorrecto de espacios y líneas en blanco.
- Código mal indentado.

MODULOS Y PAQUETES

QUE SON

Módulos y paquetes en Python Los módulos son archivos que permiten introducir funciones y clases dentro de ellos para tener nuestro código organizado, posteriormente podemos llamar a estas funciones y clases desde cualquier parte de nuestro código, los paquetes son agrupaciones de módulos.

Package Module Structure



Para se existieron

En primer lugar, nos permite unificar distintos módulos bajo un mismo nombre de paquete, pudiendo crear jerarquías de módulos y submódulos, o también subpaquetes.

Por otra parte nos permiten distribuir y manejar fácilmente nuestro código como si fueran librerías instalables de Python. De esta forma se pueden utilizar como módulos estándar desde el intérprete o scripts sin cargarlos previamente.

entornos virtuales

Si nosotros instalamos una librería con una versión en particular a nivel sistema operativo, por ejemplo instalamos Django 1.11, podremos usar Django, en esa versión en particular, para todos los proyectos en nuestra computadora.

Sin embargo, qué pasa si algún proyecto en particular de nuestro sistema requiere Django, pero no en su versión 1.11, si no ahora en la versión 4.2. Bueno allí tendríamos un problema, ya que ciertos proyectos requieren una versión en particular y otros proyectos requieren otras versiones.

Quizás te estés preguntando, Si ciertos proyectos requieren una visión y otros, otras versiones, ¿por qué no instalamos todas las versiones que necesitemos y listo?

Bueno, si hacemos eso tendremos un problema mucho mayor, ya que los proyectos pueden entrar en conflictos con sus versiones, y existe la posibilidad que, al tener tantas versiones instaladas todos los proyectos dejen de funcionar

Por ejemplo.

```
python -m venv env
```

```
pip install -r requirements.txt
```

- pip install pandas
- pip install django
- pip install polars

pypi py

Python Package Index, conocido como PyPI, es un repositorio de software para la plataforma de lenguaje de programación Python. Fue creado por el equipo de desarrollo de Python en el año 2003 para proporcionar una forma sencilla de compartir paquetes de software para la plataforma Python. El repositorio PyPI es el lugar ideal para buscar y descargar herramientas, módulos y librerías de Python.