

EJERCICIO GUIADO 2

PROPIEDAD COLECTIVA DE CÓDIGO Y NORMATIVAS DE CÓDIGO

Sonsoles Molina Abad – 100432073

Lorenzo Largacha Sanz – 100432129

Doble Grado de Ingeniería Informática y ADE – Grupo 50

Desarrollo de Software

Universidad Carlos III de Madrid, Campus de Colmenarejo

25 febrero, 2022

1. Normativa de código – reglas de estilo [BASIC]

REGLA 1: módulos → camelCase

En los nombres de los módulos, usando camelCase, la primera letra de la primera palabra debe ser minúscula y la primera letra de las siguientes palabras deben ser mayúsculas (línea 422).

REGLA 2: variables → UPPER_CASE

En los nombres de las variables definimos UPPER_CASE de modo que todas las letras sean mayúsculas (línea 442).

REGLA 3: constantes → snake_case

En los nombres de las constantes utilizamos snake_case para que todas las palabras deban estar escritas en minúscula (línea 375).

REGLA 4: constantes de clase → regex

Para que todas las constantes de clase empiecen por CONST_ y los demás caracteres sean letras mayúsculas utilizamos una expresión regex (línea 365):

```
class-const-rgx= CONST_[A-Z]{1,}
```

REGLA 5: argumentos → any

Para los nombres de los argumentos hemos decidido que no exista ninguna limitación, por lo que utilizamos any como regla de estilo (línea 328).

REGLA 6: clases → camelCase

En los nombres de las clases utilizamos camelCase para que la primera letra de la primera palabra sea minúscula y la primera letra de las siguientes palabras sean mayúsculas (línea 368).

REGLA 7: métodos → PascalCase

En los nombres de los métodos definimos PascalCase de modo que la primera letra de la primera palabra debe ser mayúscula y la primera letra de las siguientes palabras también deben ser mayúsculas, mientras que las demás son minúsculas (línea 415).

REGLA 8: funciones → PascalCase

En los nombres de las funciones definimos PascalCase de modo que la primera letra de la primera palabra debe ser mayúscula y la primera letra de las siguientes

palabras también deben ser mayúsculas, mientras que las demás son minúsculas (línea 386).

REGLA 9: atributos → camelCase

En los nombres de los atributos utilizamos camelCase para que la primera letra de la primera palabra sea minúscula y la primera letra de las siguientes palabras sean mayúsculas (línea 335).

REGLA 10: inlinevar → regex

Para que todas las variables de los bucles empiecen por un número, seguido de letras minúsculas, utilizamos una expresión regex. Su longitud mínima debe ser de dos caracteres, y la máxima de 20 (línea 412):

```
inlinevar-rgx=[0-9][a-z]{2,20}$
```

2. Normativa de código – otras configuraciones

REGLA 11: comentarios → sin restricciones [MASTER]

Para evitar que se muestren warnings debido a la cantidad de comentarios en el código utilizamos la función disable C0114 (elimina missing-module-docstring), C0115 (elimina missing-class-docstring), C0116 (elimina missing-function-docstring) (línea 66).

REGLA 12: número máximo de argumentos [DESIGN]

Modificamos el número máximo de argumentos a 15 (línea 538).

REGLA 13: número máximo de caracteres por línea [FORMAT]

Modificamos el número máximo de caracteres que puede contener una línea a 150 (línea 292).

REGLA 14: if statements de una sola línea [FORMAT]

Permitimos que los if puedan estar escritos en una sola línea si no cuentan con else (línea 303).