


Open · Opened 4 days ago by  **Rubén Montero**

Un print muuuuuuuuuuuuuu largo

Resumen

- Veremos cómo definir cadenas de texto multilinea usando las triples comillas (`"""`)
- Implementaremos una función que hace un `print` muy largo en `functions.py`

Descripción

Los `string` pueden declararse encerrados en dobles comillas (`"`):

```
una_cadena = "Esto es un texto"
```

...o comillas simples (`'`):

```
otra_cadena = 'Esto también es un texto'
```

Esto es útil si queremos que nuestros `string` contengan comillas simples o dobles, porque los encerramos en las comillas *del otro tipo*.

Por ejemplo:

```
frase_mafioso1 = "Cuando te dije que te 'encargaras' de él, no me refería a que lo mataras"
```

Y:

```
frase_mafioso2 = 'Entonces, ¿a qué te referías con "encargarme" de él?'
```

¿No hace falta *escapar* caracteres con un *slash* invertido (`\`)?

Puedes hacerlo; `\n` ó `\"` son válidos en un `string` Python.

Pero es raro que se usen.

Incluso los saltos de línea pueden generarse de forma más clara y legible empleando un...

`string` multilinea

Va encerrado en triples comillas, simples (`'''`) o dobles (`"""`).

Por ejemplo:

```
print("""
Con diez cañones por banda
Viento en popa a toda vela
No corta el mar sino vuela
Un velero bergantín...

¿Qué poeta escribió estos versos?
""")
```

La tarea

Se pide que añadas una nueva función en `functions.py` :

- Se llamará `long_print`
- No recibirá parámetros
- Únicamente, hará un `print` de un `string` multilinea con varias líneas. Se pide que en dicho `print` escribas un **chiste** que contenga la palabra **programador**
 - El contenido del chiste es *libre*. Basta con que *tenga* esa palabra.

Por último

Verifica que tu código pasa el *test* asociado a la tarea correctamente.

Haz **commit** y **push** para subir los cambios al repositorio.



[Rubén Montero @ruben.montero](#) changed milestone to [%Sprint 3](#) 4 days ago