ntic master
revolucionamos la comunicación

UNIVERSIDAD COMPLUTENSE DE MADRID

Programación. Python

Claridad

Cristóbal Pareja Flores

Claridad

- Programa bien diseñado: correcto, eficiente y **mantenible**
- **Legible**, por uno mismo y por los demás
- **Claridad** y **normas estándar**

PEP 8 -- Style Guide for Python Code

Identificadores

- Nemotécnicos, ASCII
- Referidos al concepto representado, no a su tipo de datos

	edad	entero
	nombre	cadena
Índices:	i, j, k, _	l
Varias palabras:	paises_de_europa PaisesDeEuropa	lista
Constantes:	COUNTRY_CODES PI	diccionario

```
import math

PI = math.pi
radio = 4.5
area_del_circulo = PI * radio**2
print(area_del_circulo)
```

```
63.61725123519331
```

Escritura del código

sangrado

```
# Ejemplos:

for i in range(10):
    for _ in range(i):
        print(i, end="")
    print()

print()

for pais, abreviatura in [("España", "SP"), ("Francia", "FR")]:
    print(pais, abreviatura)

print()

for pais, abreviatura in [("España", "SP"), ("Francia", "FR"), \
                          ("Portugal", "PT"), ("Gran Bretaña", "GB")]:
    print(pais, abreviatura)

print()

for pais, abreviatura in [
    ("España", "SP"), ("Francia", "FR"), \
    ("Portugal", "PT"), ("Gran Bretaña", "GB")
]:
    print(pais, abreviatura)
```

```
1
22
333
4444
55555
666666
7777777
88888888
999999999

España SP
Francia FR

España SP
Francia FR
Portugal PT
Gran Bretaña GB

España SP
Francia FR
Portugal PT
Gran Bretaña GB
```

Escritura del código

sangrado

```
# Ejemplos:

for i in range(10):
    for _ in range(i):
        print(i, end="")
    print()

print()

for pais, abreviatura in [("España", "SP"), ("Francia", "FR")]:
    print(pais, abreviatura)

print()

for pais, abreviatura in [("España", "SP"), ("Francia", "FR"), \
                          ("Portugal", "PT"), ("Gran Bretaña", "GB")]:
    print(pais, abreviatura)

print()

for pais, abreviatura in [
    ("España", "SP"), ("Francia", "FR"), \
    ("Portugal", "PT"), ("Gran Bretaña", "GB")
]:
    print(pais, abreviatura)
```

```
1
22
333
4444
55555
666666
7777777
88888888
999999999

España SP
Francia FR

España SP
Francia FR
Portugal PT
Gran Bretaña GB

España SP
Francia FR
Portugal PT
Gran Bretaña GB
```

Escritura del código

```
# Ejemplos:

for i in range(10):
    for _ in range(i):
        print(i, end="")
    print()
print()

for pais, abreviatura in [("España", "SP"), ("Francia", "FR")]:
    print(pais, abreviatura)

print()

for pais, abreviatura in [("España", "SP"), ("Francia", "FR"), \
                          ("Portugal", "PT"), ("Gran Bretaña", "GB")]:
    print(pais, abreviatura)

print()

for pais, abreviatura in [
    ("España", "SP"), ("Francia", "FR"), \
    ("Portugal", "PT"), ("Gran Bretaña", "GB")
]:
    print(pais, abreviatura)
```

sangrado

<= 79 cars

líneas

```
1
22
333
4444
55555
666666
7777777
88888888
999999999

España SP
Francia FR

España SP
Francia FR
Portugal PT
Gran Bretaña GB

España SP
Francia FR
Portugal PT
Gran Bretaña GB
```

Comentarios

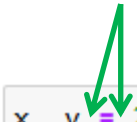
- ASCII
- Breves
- Docstring:
  - Descripción
  - Parámetros
  - Requisitos
  - Devuelve
  - Ejemplos

```
def radius_of_circle(area):
    """
    Given the area of a circle, returns its radius

    Parameters:
    -----------
    area: float
        the area of the circle

    Precondition:
    -------------
    area >= 0

    Returns:
    --------
    float
        The radios of the circle

    Example:
    --------
    >>> radius_of_circle(9.0)
    1.692568750643269
    """

    PI = math.pi
    return math.sqrt(area / PI)
```

Constantes globales

```
# Constantes globales:

PI = 3.14
DATA = "./data/clients.csv"
COUNTRY_CODES = {"Spain": "ES", "Ireland": "IE", "France": "FR", \
                 "United Kingdom": "GB", "United States": "US"}
```

Espacios



```
x, y = 2.5, 5.0
z = math.sin(math.pi*5)
print(z**4 + 4*x**3)
```

```
62.5
```

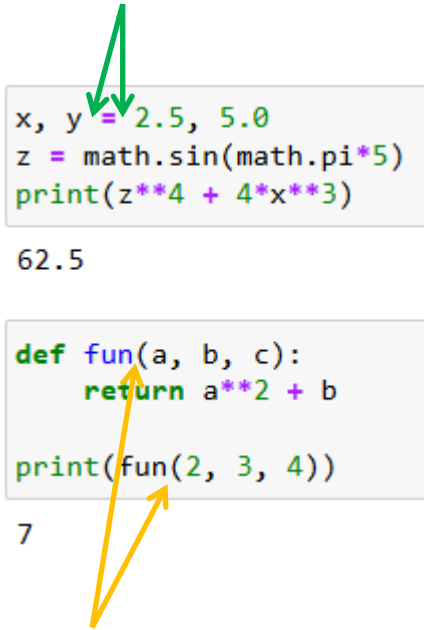
```
def fun(a, b, c):
    return a**2 + b

print(fun(2, 3, 4))
```

```
7
```

Constantes globales

```
# Constantes globales:

PI = 3.14
DATA = "./data/clients.csv"
COUNTRY_CODES = {"Spain": "ES", "Ireland": "IE", "France": "FR", \
                 "United Kingdom": "GB", "United States": "US"}
```

Espacios



```
x, y = 2.5, 5.0
z = math.sin(math.pi*5)
print(z**4 + 4*x**3)
```

```
62.5
```

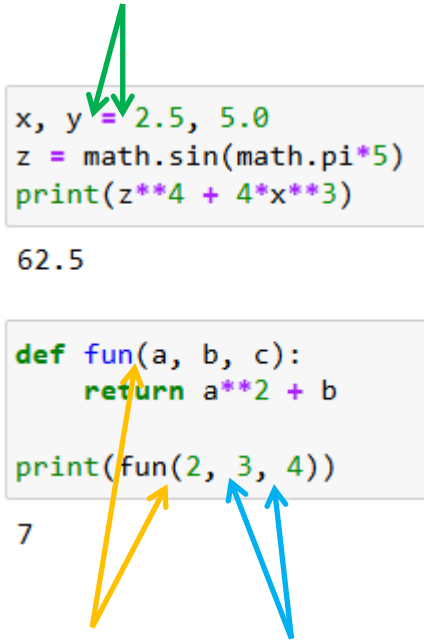
```
def fun(a, b, c):
    return a**2 + b

print(fun(2, 3, 4))
```

```
7
```

Constantes globales

```
# Constantes globales:

PI = 3.14
DATA = "./data/clients.csv"
COUNTRY_CODES = {"Spain": "ES", "Ireland": "IE", "France": "FR", \
                 "United Kingdom": "GB", "United States": "US"}
```

Espacios



```
x, y = 2.5, 5.0
z = math.sin(math.pi*5)
print(z**4 + 4*x**3)
```

```
62.5
```

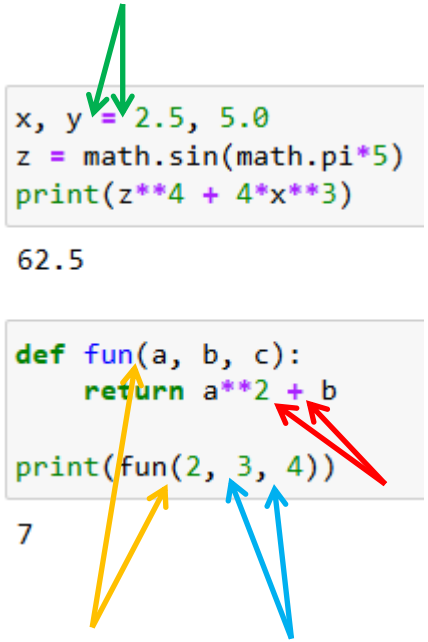
```
def fun(a, b, c):
    return a**2 + b

print(fun(2, 3, 4))
```

```
7
```

Constantes globales

```
# Constantes globales:

PI = 3.14
DATA = "./data/clients.csv"
COUNTRY_CODES = {"Spain": "ES", "Ireland": "IE", "France": "FR", \
                 "United Kingdom": "GB", "United States": "US"}
```

Espacios



```
x, y = 2.5, 5.0
z = math.sin(math.pi*5)
print(z**4 + 4*x**3)
```

```
62.5
```

```
def fun(a, b, c):
    return a**2 + b

print(fun(2, 3, 4))
```

```
7
```

ntic master
revolucionamos la comunicación

UNIVERSIDAD COMPLUTENSE DE MADRID

Programación. Python

Claridad

Cristóbal Pareja Flores