

# Comentarios

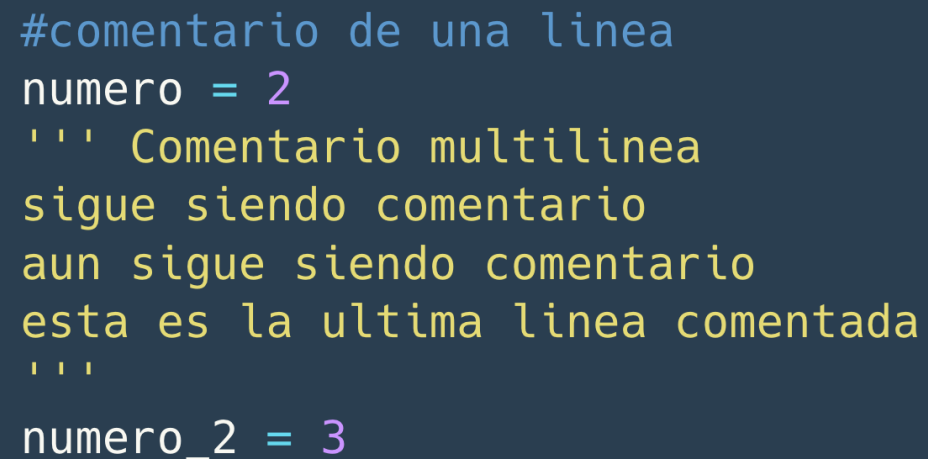
The background is a dark navy blue. It features a pattern of overlapping, organic, pill-shaped and circular elements in various colors: teal, green, brown, and a muted purple. A light blue grid is overlaid on the entire image, consisting of vertical and horizontal lines. There are also faint, light blue curved lines at the top and bottom edges, suggesting a circular or hemispherical shape.

Poner  
comentarios en el  
código: ¿por qué,  
cuándo y dónde?



Quizá en algún momento será necesario poner algunas palabras en el código dirigidas no a Python, sino a las personas quienes estén leyendo el código con el fin de explicarles como es que funciona, o tal vez especificar el significado de las variables, también para documentar quien es el autor del programa y en que fecha fue escrito.

Un texto insertado en el programa el cual es, omitido en la ejecución, es denominado un comentario.



```
#comentario de una linea
numero = 2
''' Comentario multilinea
sigue siendo comentario
aun sigue siendo comentario
esta es la ultima linea comentada
'''
numero_2 = 3
```

¿Cómo se colocan este tipo de comentarios en el código fuente? Tiene que ser hecho de cierta manera para que Python no intente interpretarlo como parte del código.

Cuando Python se encuentra con un comentario en el programa, el comentario es completamente transparente, desde el punto de vista de Python, el comentario es solo un espacio vacío, sin importar que tan largo sea.



EN PYTHON, UN COMENTARIO ES UN TEXTO QUE COMIENZA CON EL SÍMBOLO # Y SE EXTIENDE HASTA EL FINAL DE LA LÍNEA.



SI SE DESEA COLOCAR UN COMENTARIO QUE ABARCA VARIAS LÍNEAS, SE DEBE COLOCAR ESTE SÍMBOLO EN CADA LÍNEA.



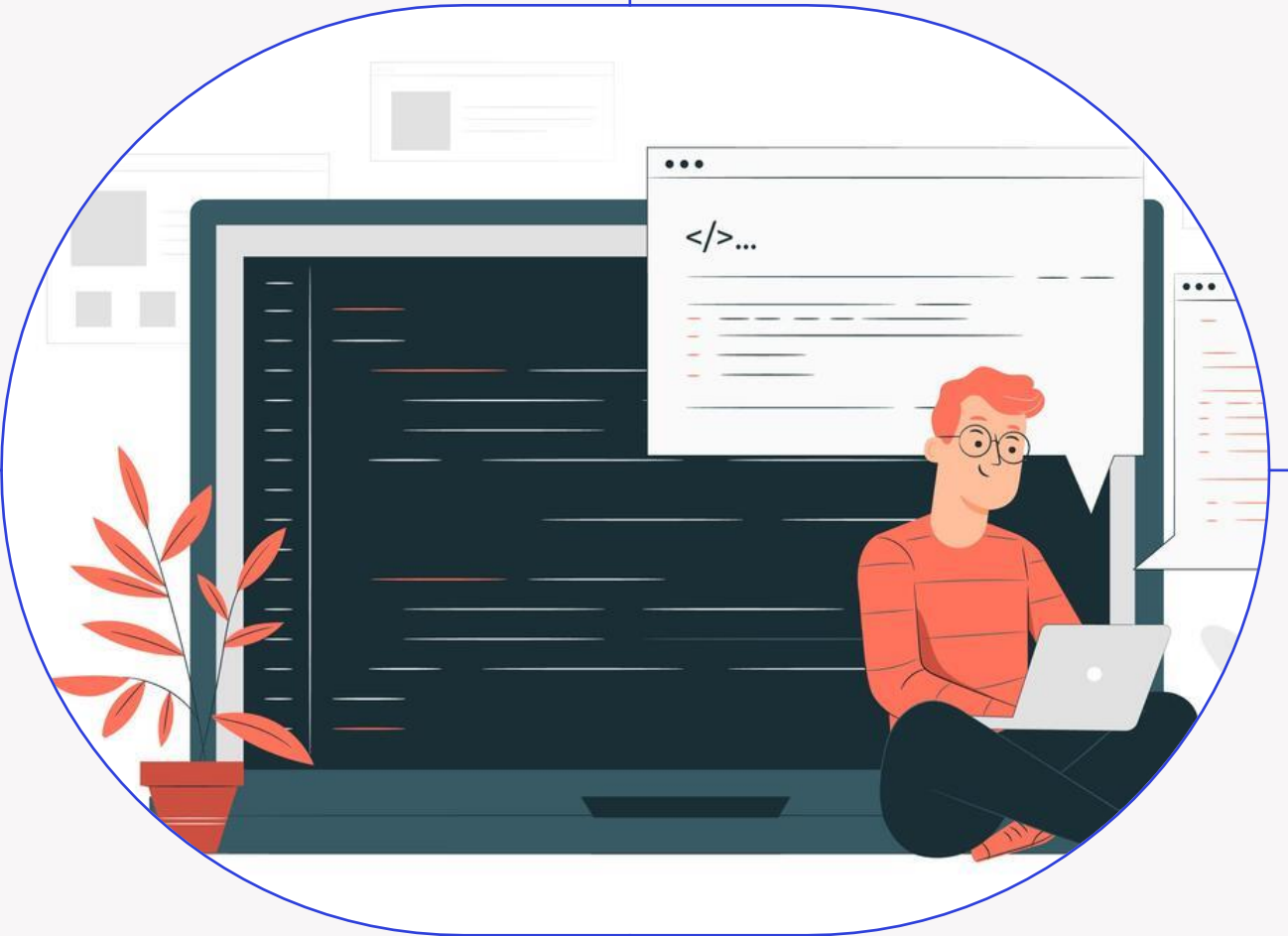
JUSTO COMO EL SIGUIENTE CÓDIGO:

SECTION (33%)

```
# Esta programa calcula la hipotenusa (c)
# a y b son las longitudes de los catetos
a = 3.0
b = 4.0
c = (a ** 2 + b ** 2) ** 0.5 # se utiliza ** en lugar de la raíz cuadrada.
print("c =", c)
```

# ¿Importancia?

Los desarrolladores buenos y responsables describen cada pieza importante de código, por ejemplo, el explicar el rol de una variable; aunque la mejor manera de comentar una variable es dándole un nombre que no sea ambiguo.



Por ejemplo, si una variable determinada esta diseñada para almacenar el área de un cuadrado, el nombre `area_cuadrado` será muchísimo mejor que `tia_juana`.

El nombre dado a la variable se puede definir como auto-comentable.

Los comentarios pueden ser útiles en otro aspecto, se pueden utilizar para marcar un fragmento de código que actualmente no se necesita, cual sea la razón. Observa el siguiente ejemplo, si se descomenta la línea resaltada, esto afectara la salida o resultado del código:

```
# Este es un programa de prueba.  
x = 1  
y = 2  
# y = y + x  
print(x + y)
```

Esto es frecuentemente realizado cuando se esta probando un programa, con el fin de aislar un fragmento de código donde posiblemente se encuentra un error.