Comments can be used to explain Python code.

Comments can be used to make the code more readable.

Comments can be used to prevent execution when testing code.

## Creating a Comment

Comments starts with a #, and Python will ignore them:

### **Example**[**Get your own Python Server**](https://www.w3schools.com/spaces/)

#This is a comment  
print("Hello, World!")

[Try it Yourself »](https://www.w3schools.com/python/trypython.asp?filename=demo_comment1)

Comments can be placed at the end of a line, and Python will ignore the rest of the line:

### **Example**

print("Hello, World!") #This is a comment

[Try it Yourself »](https://www.w3schools.com/python/trypython.asp?filename=demo_comment2)

A comment does not have to be text that explains the code, it can also be used to prevent Python from executing code:

### **Example**

#print("Hello, World!")  
print("Cheers, Mate!")

[Try it Yourself »](https://www.w3schools.com/python/trypython.asp?filename=demo_comment3)

## Multiline Comments

Python does not really have a syntax for multiline comments.

To add a multiline comment you could insert a # for each line:

### **Example**

#This is a comment  
#written in  
#more than just one line  
print("Hello, World!")

[Try it Yourself »](https://www.w3schools.com/python/trypython.asp?filename=demo_comment4)

Or, not quite as intended, you can use a multiline string.

Since Python will ignore string literals that are not assigned to a variable, you can add a multiline string (triple quotes) in your code, and place your comment inside it:

### **Example**

"""  
This is a comment  
written in  
more than just one line  
"""  
print("Hello, World!")

[Try it Yourself »](https://www.w3schools.com/python/trypython.asp?filename=demo_comment5)

As long as the string is not assigned to a variable, Python will read the code, but then ignore it, and you have made a multiline comment.