

with open

How Does the With Statement Work in Python?

The with statement works with the open() function to open a file.

So, you can re-write the code we used in the open() function example like this:

```
with open("hello.txt") as my_file:  
    print(my_file.read())
```

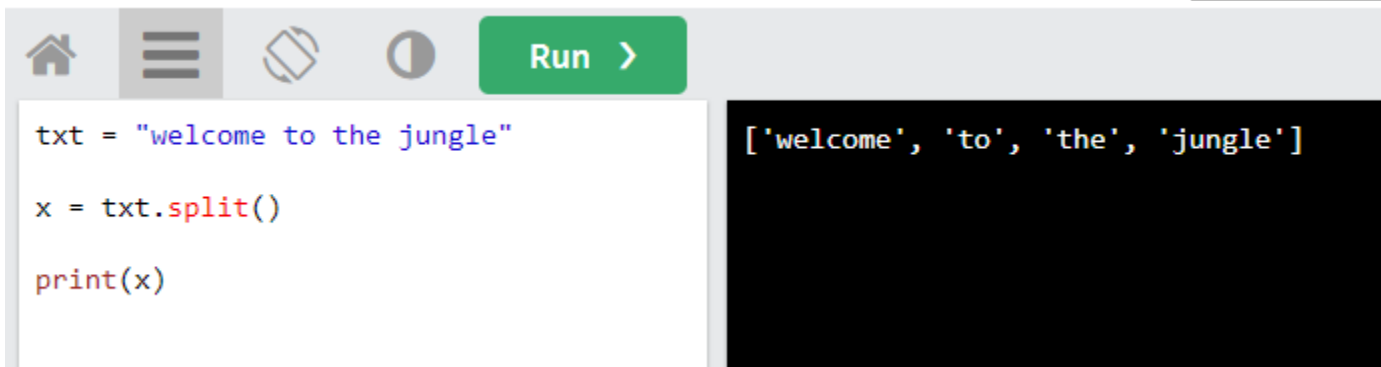
```
# Output :  
# Hello world  
# I hope you're doing well today  
# This is a text file
```

split() method

Definition and Usage

The split() method splits a string into a list.

You can specify the separator, default separator is any whitespace.



The screenshot shows a Python IDE interface. At the top, there is a toolbar with icons for home, menu, undo, and redo, followed by a green 'Run >' button. Below the toolbar, the code editor contains the following Python code:

```
txt = "welcome to the jungle"  
x = txt.split()  
print(x)
```

To the right of the code editor, the output is displayed on a black background with yellow text:

```
['welcome', 'to', 'the', 'jungle']
```

Dictionary Methods

Python has a set of built-in methods that you can use on dictionaries.

Method	Description
<code>clear()</code>	Removes all the elements from the dictionary
<code>copy()</code>	Returns a copy of the dictionary
<code>fromkeys()</code>	Returns a dictionary with the specified keys and value
<code>get()</code>	Returns the value of the specified key
<code>items()</code>	Returns a list containing a tuple for each key value pair
<code>keys()</code>	Returns a list containing the dictionary's keys
<code>pop()</code>	Removes the element with the specified key
<code>popitem()</code>	Removes the last inserted key-value pair
<code>setdefault()</code>	Returns the value of the specified key. If the key does not exist: insert the key, with the specified value
<code>update()</code>	Updates the dictionary with the specified key-value pairs
<code>values()</code>	Returns a list of all the values in the dictionary

Horizontal Bars

Creating Bars

With Pyplot, you can use the `bar()` function to draw bar graphs:

Example

[Get your own Python Server](#)

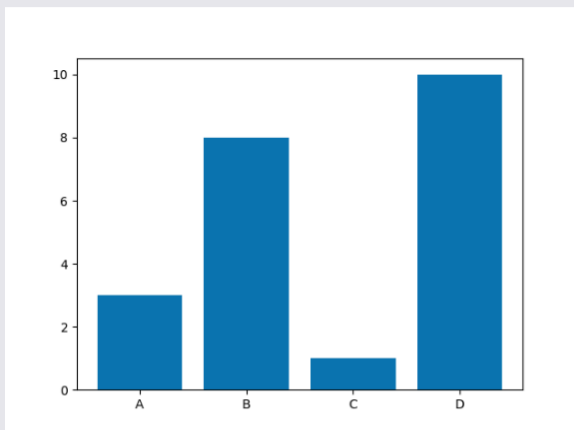
Draw 4 bars:

```
import matplotlib.pyplot as plt
import numpy as np

x = np.array(["A", "B", "C", "D"])
y = np.array([3, 8, 1, 10])

plt.bar(x,y)
plt.show()
```

Result:

[Try it Yourself »](#)

The `bar()` function takes arguments that describes the layout of the bars.

The categories and their values represented by the *first* and *second* argument as arrays.