Python Event Class | set() Method with Example

FAST, RELIABLE FIBRE.
FLEXIBLE TV PACKAGE.
Shop now →
£4.95 P&P. Terms and price changes apply.

Python Event.set() Method: Here, we are going to learn about the **set() method of Event Class in Python** with its definition, syntax, and examples.

Submitted by Hritika Rajput, on May 22, 2020

Python Event.set() Method

set() is an inbuilt method of the *Event* class of the *threading* module in Python.

When the `set()` method is called, the internal flag of that event class object is set to true. As the `set()` method gets called for an object, all the threads waiting for that event object get awakened.

Module:

```
from threading import Event
```

Syntax:

```
set()
```

Parameter(s):

- None

Return value:

The return type of this method is `<class 'NoneType'>`. The method does not return anything. It sets the internal flag of an event object to true.

Example 1:

Copy

```
# Python program to explain the
# use of set() method in Event() class
import threading
import time

def helper_function(event_obj, timeout,i):
  # Thread has started, but it will wait 10 seconds
  # for the event
  print("Thread started, for the event to set")

  flag = event_obj.wait(timeout)
  if flag:
    print("Event was set to true() earlier, moving ahead with the thread")
  else:
    print("Time out occured, event internal flag still false. Executing thread without wai
    print("Value to be printed=", i)

if __name__ == '__main__':
  # Initialising an event object
  event_obj = threading.Event()

  # starting the thread who will wait for the event
  thread1 = threading.Thread(target=helper_function, args=(event_obj,10,27))
  thread1.start()
  # sleeping the current thread for 5 seconds
  time.sleep(5)

  # generating the event
  event_obj.set()
  print("Event is set to true. Now threads can be released.")
  print()
```

Output:

```
Thread started, for the event to set
Event is set to true. Now threads can be released.
Event was set to true() earlier, moving ahead with the thread
```

Example 2:

Copy

```
# Python program to explain the
# use of set() method in Event() class
import threading
import time

def helper_function(event_obj, timeout,i):
  # Thread has started, but it will wait 3 seconds
  # for the event
  print("Thread started, for the event to set")

  flag = event_obj.wait(timeout)
  if flag:
    print("Event was set to true() earlier, moving ahead with the thread")
  else:
    print("Time out occured, event internal flag still false. Executing thread without wai
    print("Value to be printed=", i)

if __name__ == '__main__':
  # Initialising an event object
  event_obj = threading.Event()

  # starting the thread who will wait for the event
  thread1 = threading.Thread(target=helper_function, args=(event_obj,3,27))
  thread1.start()
  # sleeping the current thread for 5 seconds
  time.sleep(5)

  # generating the event
  event_obj.set()
  print("Event is set to true. Now threads can be released.")
  print()
```

Output:

```
Thread started, for the event to set
Time out occured, event internal flag still false. Executing thread without waiting for ev
Value to be printed= 27
Event is set to true. Now threads can be released.
```


TOP Interview Coding Problems/Challenges

Run-length encoding (find/print frequency of letters in a string)
Sort an array of 0's, 1's and 2's in linear time complexity

Checking Anagrams (check whether two string is anagrams or not)
Relative sorting algorithm
Finding subarray with given sum
Find the level in a binary tree with given sum K
Check whether a Binary Tree is BST (Binary Search Tree) or not
1[0]1 Pattern Count
Capitalize first and last letter of each word in a line
Print vertical sum of a binary tree
Print Boundary Sum of a Binary Tree
Reverse a single linked list
Greedy Strategy to solve major algorithm problems
Job sequencing problem
Root to leaf Path Sum
Exit Point in a Matrix
Find length of loop in a linked list
Toppers of Class
Print All Nodes that don't have Sibling
Transform to Sum Tree
Shortest Source to Destination Path

Comments and Discussions

Load comments

© https://www.includehelp.com some rights reserved.