ASSIGNMENT 03

TOPIC: NEWS TRACKER APPLICATION

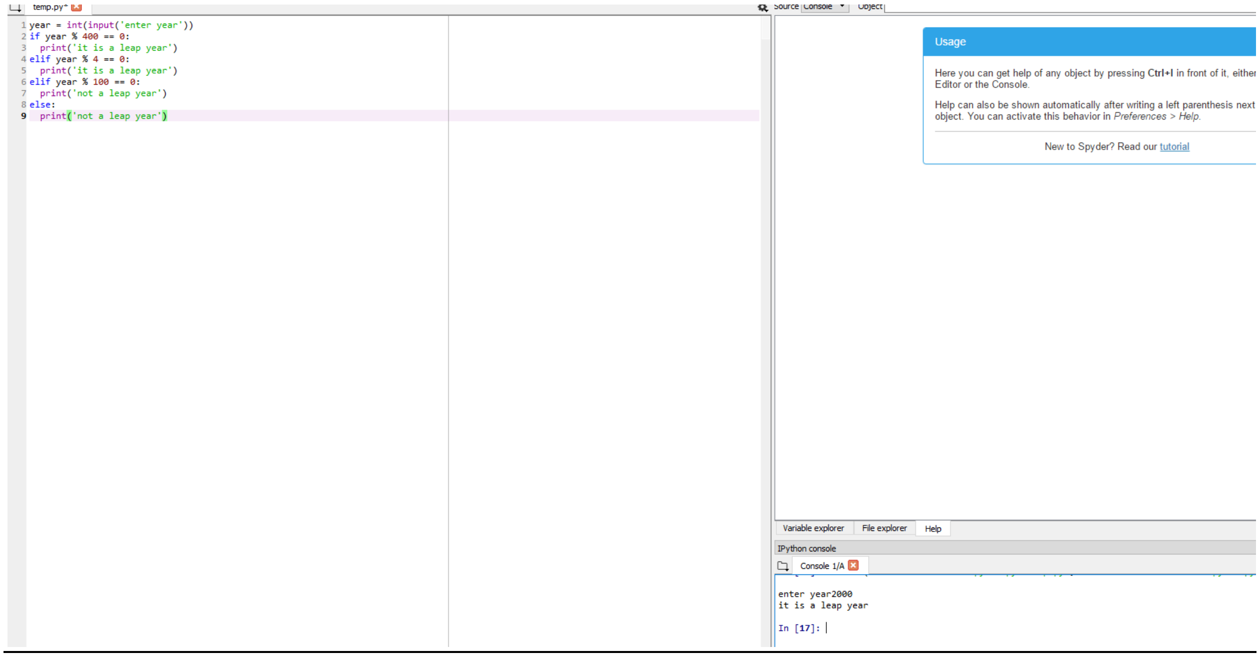
TEAM MEMBERS: S.RABBANI

S.MOHAMMED KHUBABE

M.JAI PRAVEEN

S.SURYA KUMAR

1.PYTHON PROGRAMS IN SPYDER



```
year = int(input('enter year'))
if year % 400 == 0:
  print('it is a leap year')
elif year % 4 == 0:
  print('it is a leap year')
elif year % 100 == 0:
  print('not a leap year')
else:
  print('not a leap year')
```

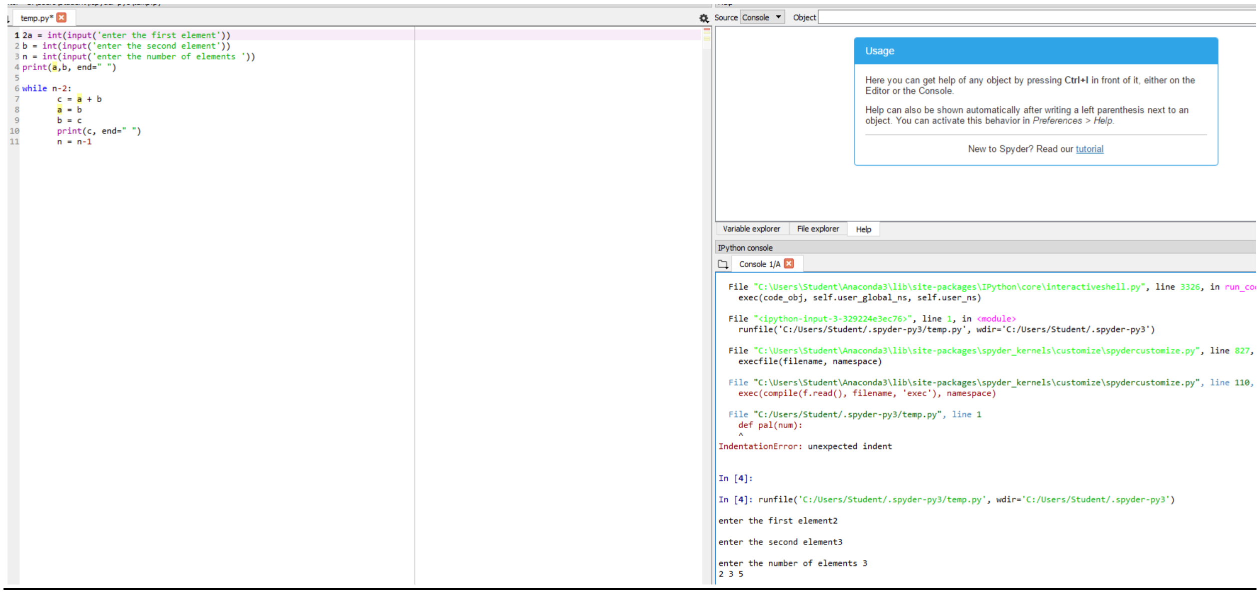
Usage

Here you can get help of any object by pressing **Ctrl+I** in front of it, either Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next object. You can activate this behavior in *Preferences > Help*.

New to Spyder? Read our tutorial

```
enter year2000
it is a leap year

In [17]: 
```



```
2a = int(input('enter the first element'))
b = int(input('enter the second element'))
n = int(input('enter the number of elements '))
print(a,b, end=" ")

while n-2:
        c = a + b
        a = b
        b = c
        print(c, end=" ")
        n = n-1
```

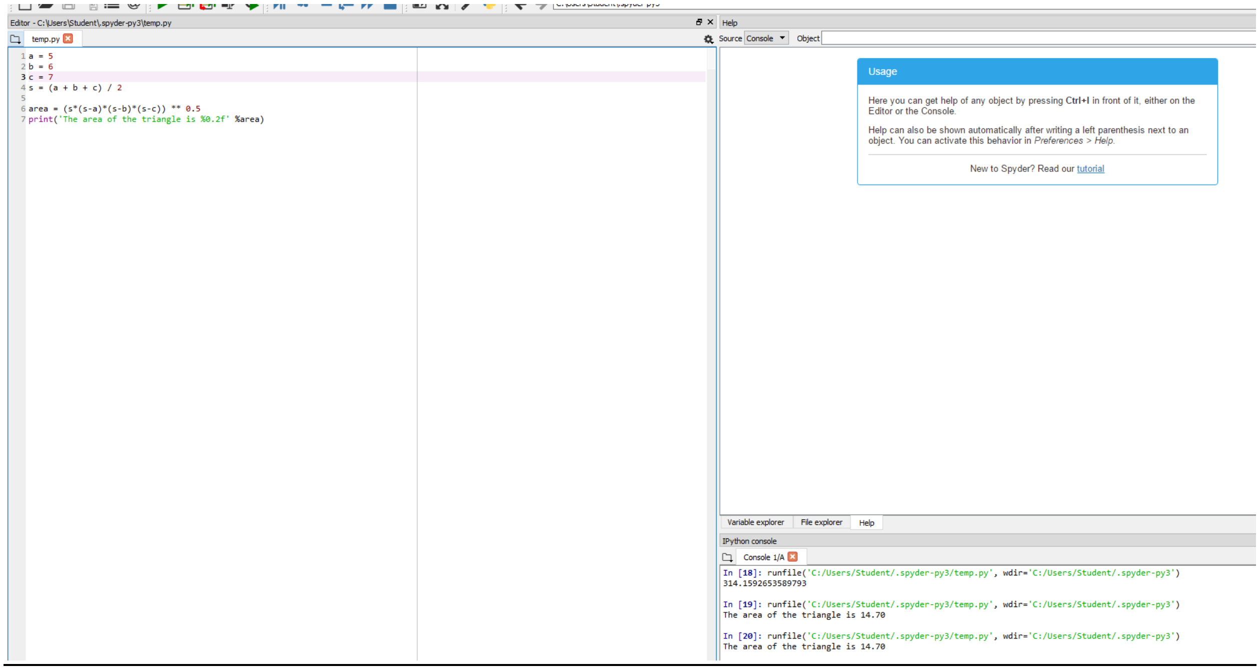
Usage

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

New to Spyder? Read our tutorial

```
  File "C:\Users\Student\Anaconda3\lib\site-packages\IPython\core\interactiveshell.py", line 3326, in run_co
    exec(code_obj, self.user_global_ns, self.user_ns)

  File "<ipython-input-3-329224e3ec76>", line 1, in <module>
    runfile('C:/Users/Student/.spyder-py3/temp.py', wdir='C:/Users/Student/.spyder-py3')

  File "C:\Users\Student\Anaconda3\lib\site-packages\spyder_kernels\customize\spydercustomize.py", line 827,
    execfile(filename, namespace)

  File "C:\Users\Student\Anaconda3\lib\site-packages\spyder_kernels\customize\spydercustomize.py", line 110,
    exec(compile(f.read(), filename, 'exec'), namespace)

  File "C:/Users/Student/.spyder-py3/temp.py", line 1
    def pal(num):
    ^
IndentationError: unexpected indent


In [4]:

In [4]: runfile('C:/Users/Student/.spyder-py3/temp.py', wdir='C:/Users/Student/.spyder-py3')

enter the first element2

enter the second element3

enter the number of elements 3
2 3 5
```



```python
a = 5
b = 6
c = 7
s = (a + b + c) / 2

area = (s*(s-a)*(s-b)*(s-c)) ** 0.5
print('The area of the triangle is %0.2f' %area)
```

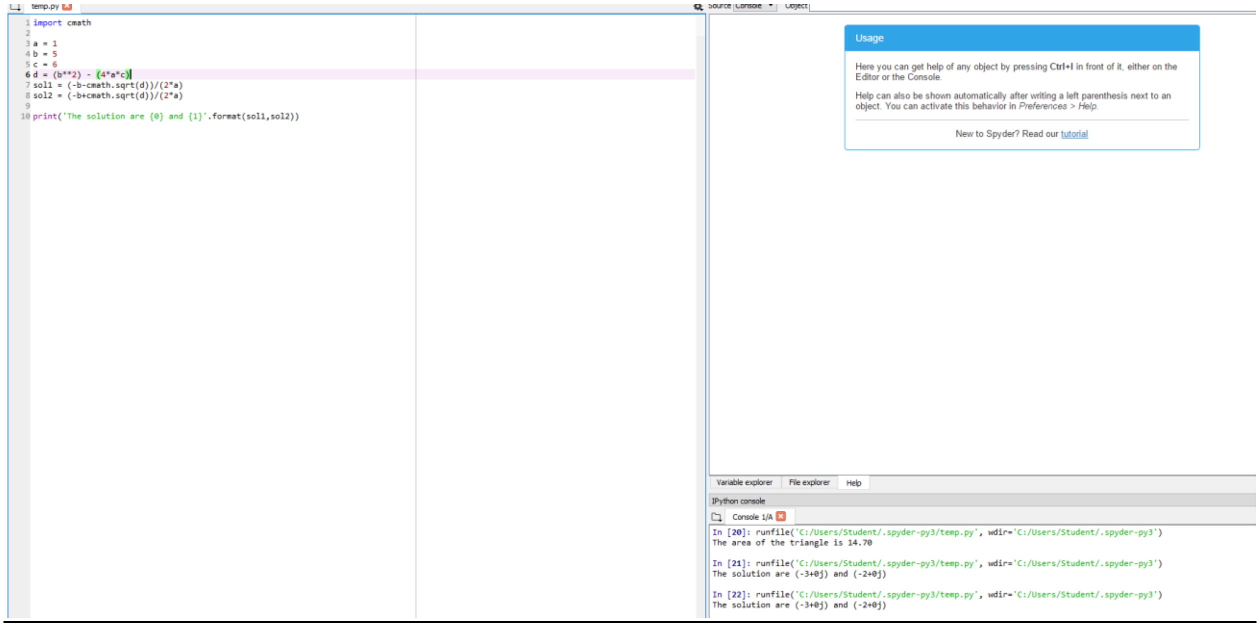
Usage

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

New to Spyder? Read our tutorial

```
In [18]: runfile('C:/Users/Student/.spyder-py3/temp.py', wdir='C:/Users/Student/.spyder-py3')
314.1592653589793

In [19]: runfile('C:/Users/Student/.spyder-py3/temp.py', wdir='C:/Users/Student/.spyder-py3')
The area of the triangle is 14.70

In [20]: runfile('C:/Users/Student/.spyder-py3/temp.py', wdir='C:/Users/Student/.spyder-py3')
The area of the triangle is 14.70
```



```python
import cmath

a = 1
b = 5
c = 6
d = (b**2) - (4*a*c)
sol1 = (-b-cmath.sqrt(d))/(2*a)
sol2 = (-b+cmath.sqrt(d))/(2*a)

print('The solution are {0} and {1}'.format(sol1,sol2))
```

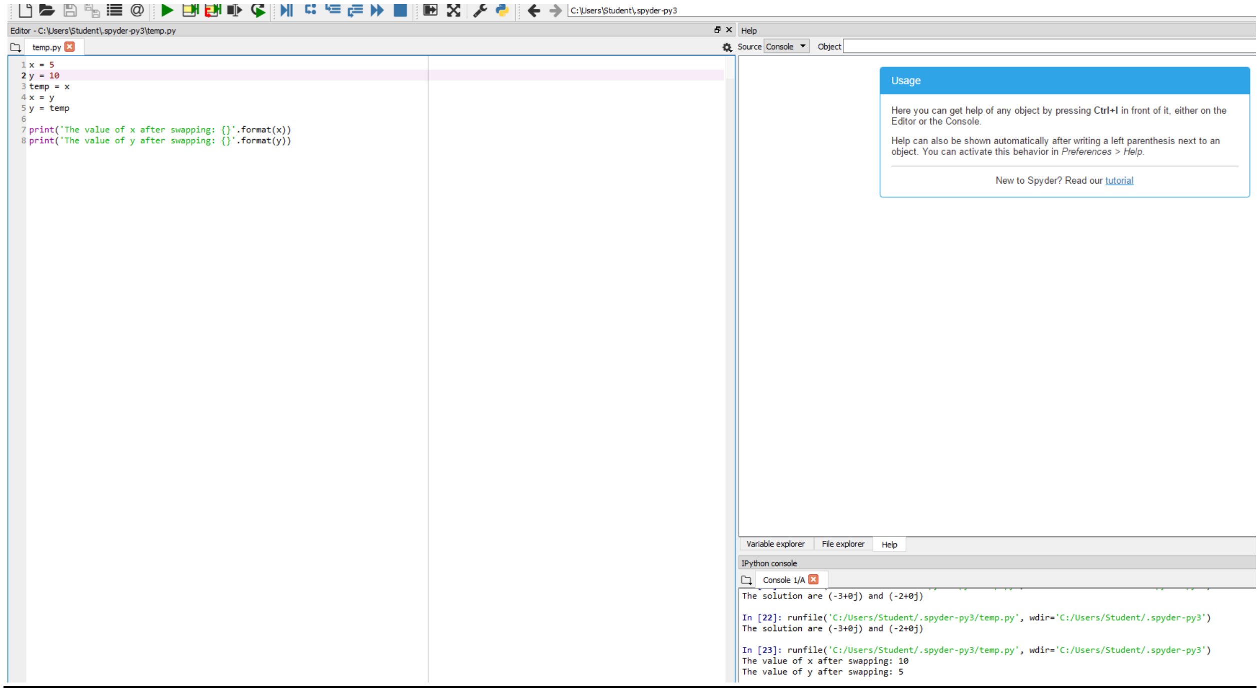
Usage

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

New to Spyder? Read our tutorial

```
In [20]: runfile('C:/Users/Student/.spyder-py3/temp.py', wdir='C:/Users/Student/.spyder-py3')
The area of the triangle is 14.70

In [21]: runfile('C:/Users/Student/.spyder-py3/temp.py', wdir='C:/Users/Student/.spyder-py3')
The solution are (-3+0j) and (-2+0j)

In [22]: runfile('C:/Users/Student/.spyder-py3/temp.py', wdir='C:/Users/Student/.spyder-py3')
The solution are (-3+0j) and (-2+0j)
```



```python
x = 5
y = 10
temp = x
x = y
y = temp

print('The value of x after swapping: {}'.format(x))
print('The value of y after swapping: {}'.format(y))
```

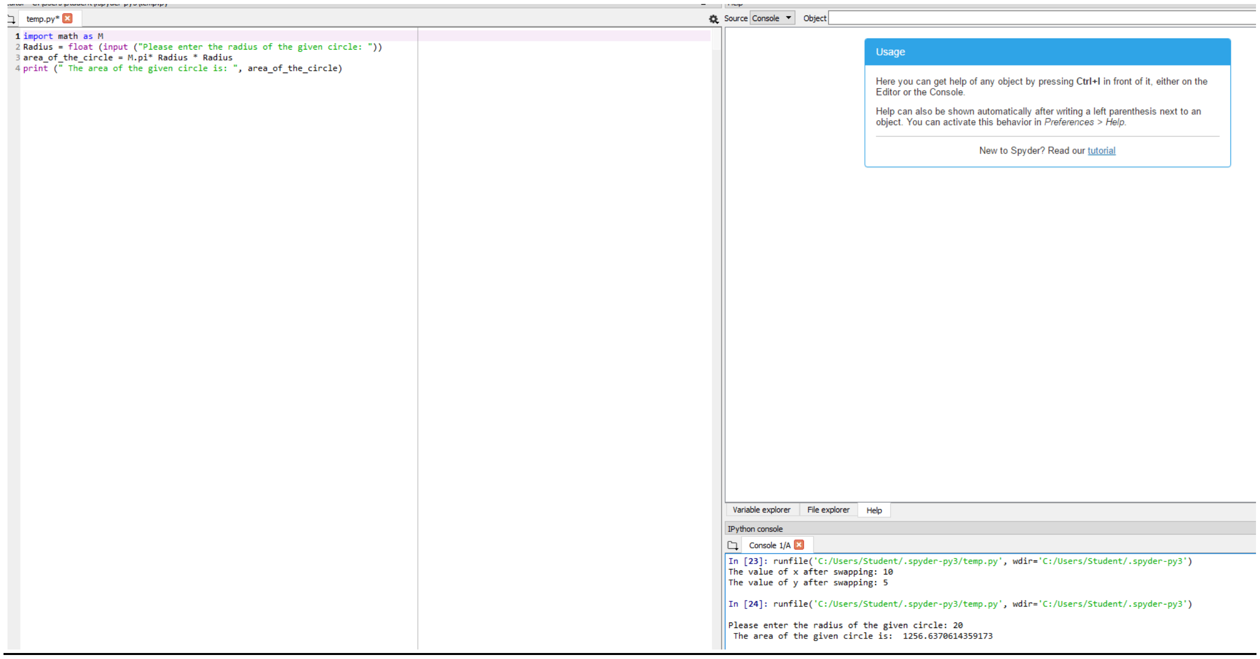
Usage

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help.*

New to Spyder? Read our tutorial

```
The solution are (-3+0j) and (-2+0j)

In [22]: runfile('C:/Users/Student/.spyder-py3/temp.py', wdir='C:/Users/Student/.spyder-py3')
The solution are (-3+0j) and (-2+0j)

In [23]: runfile('C:/Users/Student/.spyder-py3/temp.py', wdir='C:/Users/Student/.spyder-py3')
The value of x after swapping: 10
The value of y after swapping: 5
```



```
import math as M
Radius = float (input ("Please enter the radius of the given circle: "))
area_of_the_circle = M.pi* Radius * Radius
print (" The area of the given circle is: ", area_of_the_circle)
```

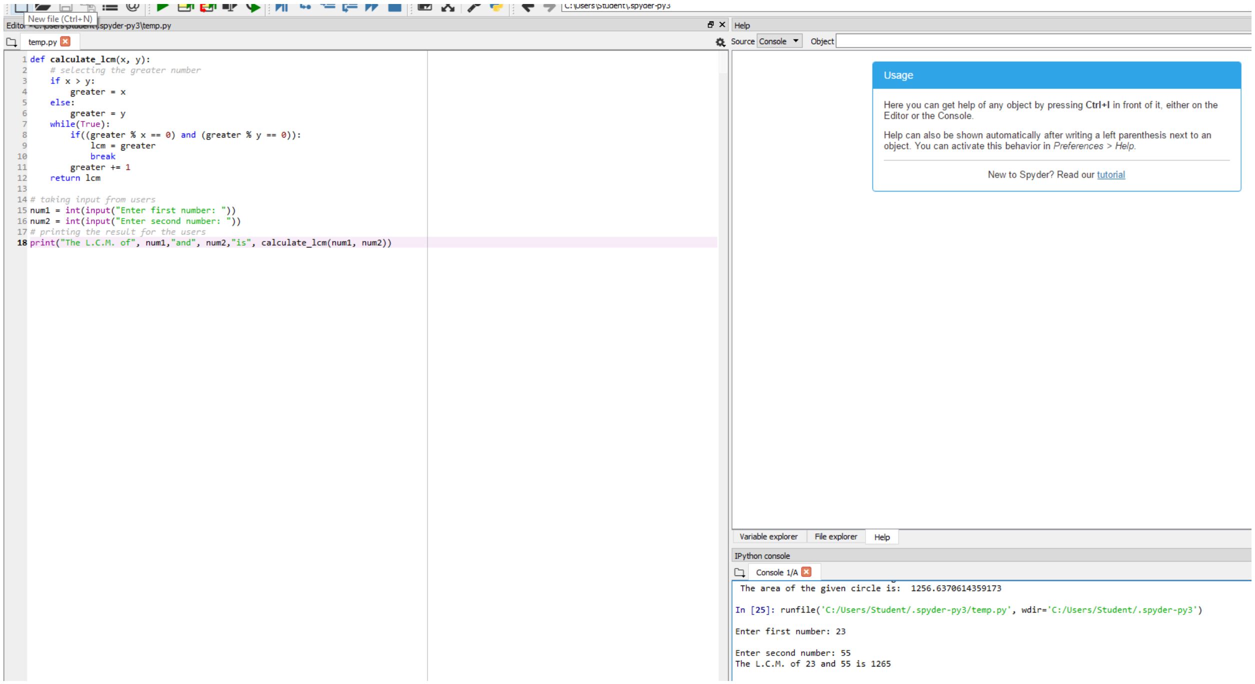
Usage

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

New to Spyder? Read our tutorial

```
In [23]: runfile('C:/Users/Student/.spyder-py3/temp.py', wdir='C:/Users/Student/.spyder-py3')
The value of x after swapping: 10
The value of y after swapping: 5

In [24]: runfile('C:/Users/Student/.spyder-py3/temp.py', wdir='C:/Users/Student/.spyder-py3')

Please enter the radius of the given circle: 20
 The area of the given circle is:  1256.6370614359173
```



```python
def calculate_lcm(x, y):
    # selecting the greater number
    if x > y:
        greater = x
    else:
        greater = y
    while(True):
        if((greater % x == 0) and (greater % y == 0)):
            lcm = greater
            break
        greater += 1
    return lcm

# taking input from users
num1 = int(input("Enter first number: "))
num2 = int(input("Enter second number: "))
# printing the result for the users
print("The L.C.M. of", num1,"and", num2,"is", calculate_lcm(num1, num2))
```

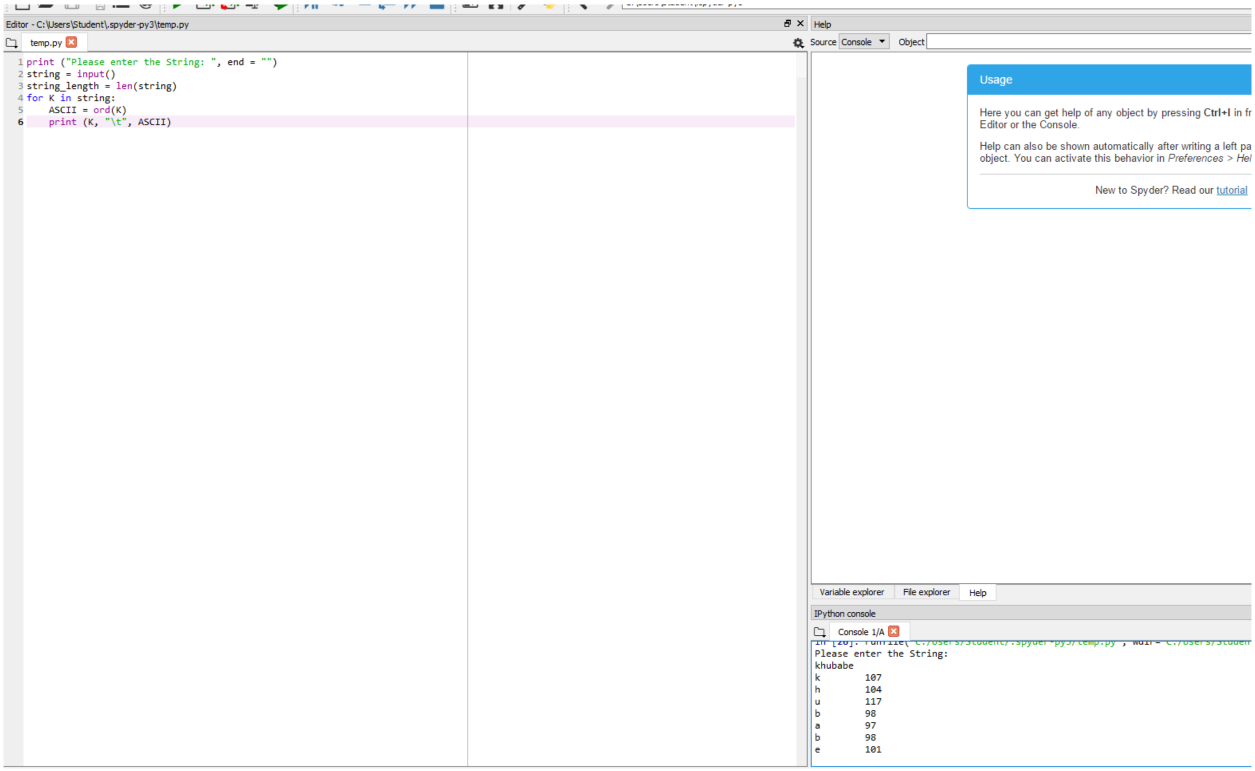
Usage

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

New to Spyder? Read our tutorial

```
 The area of the given circle is:  1256.6370614359173

In [25]: runfile('C:/Users/Student/.spyder-py3/temp.py', wdir='C:/Users/Student/.spyder-py3')

Enter first number: 23

Enter second number: 55
The L.C.M. of 23 and 55 is 1265
```



```
print ("Please enter the String: ", end = "")
string = input()
string_length = len(string)
for K in string:
    ASCII = ord(K)
    print (K, "\t", ASCII)
```

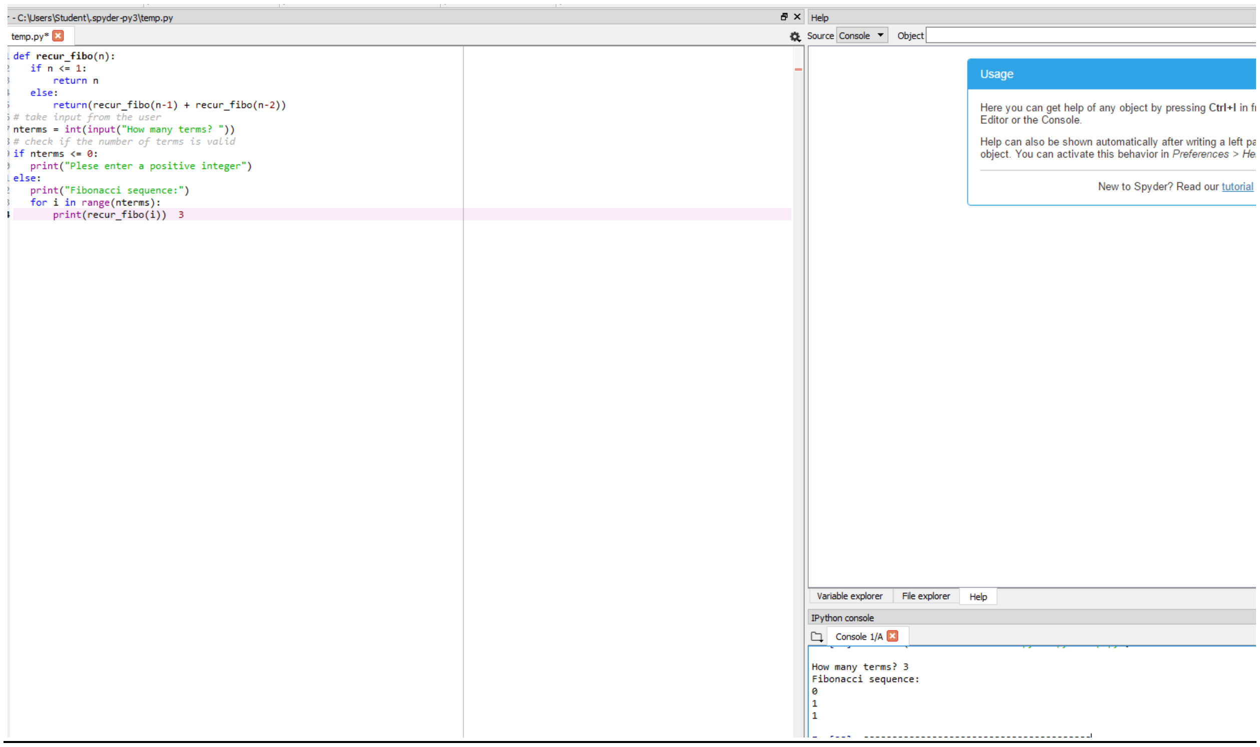
Usage

Here you can get help of any object by pressing **Ctrl+I** in fr
Editor or the Console.

Help can also be shown automatically after writing a left pa
object. You can activate this behavior in *Preferences > Hel*

New to Spyder? Read our tutorial

```
In [26]: runfile('C:/Users/Student/.spyder-py3/temp.py', wdir='C:/Users/Studen
Please enter the String:
khubabe
k       107
h       104
u       117
b       98
a       97
b       98
e       101
```



```
def recur_fibo(n):
   if n <= 1:
       return n
   else:
       return(recur_fibo(n-1) + recur_fibo(n-2))
# take input from the user
nterms = int(input("How many terms? "))
# check if the number of terms is valid
if nterms <= 0:
   print("Plese enter a positive integer")
else:
   print("Fibonacci sequence:")
   for i in range(nterms):
       print(recur_fibo(i))  3
```

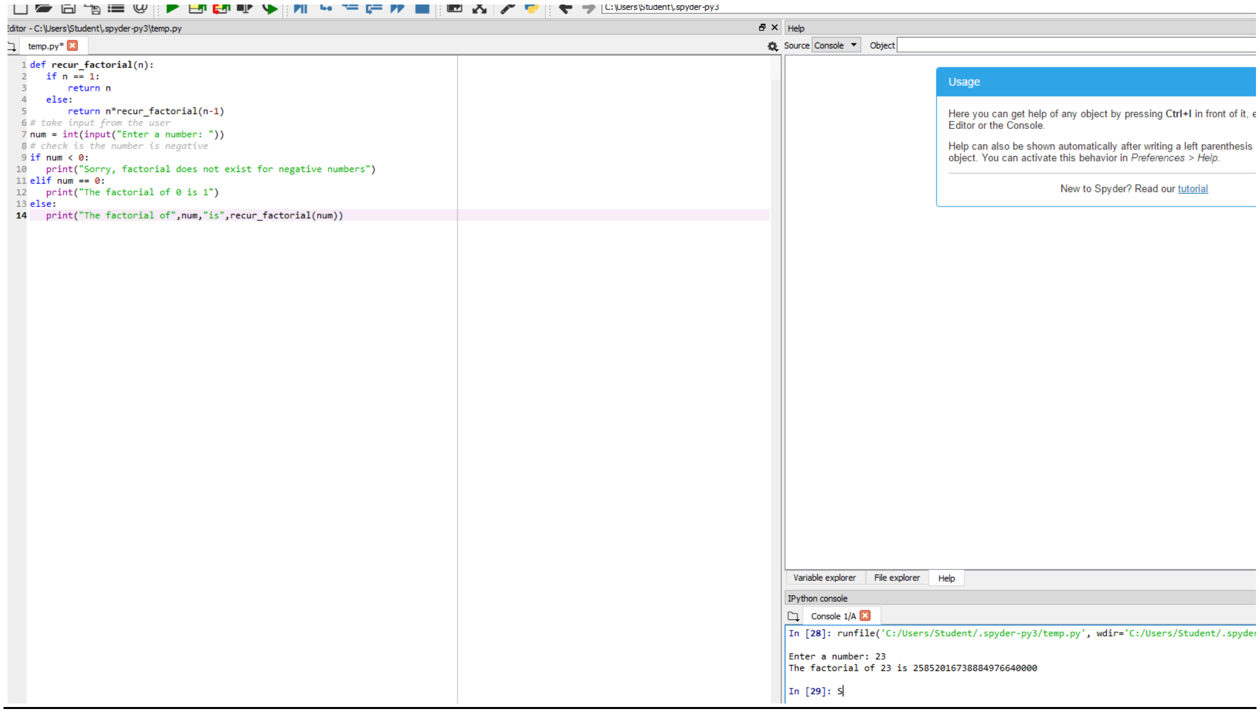
Usage

Here you can get help of any object by pressing **Ctrl+I** in fr
Editor or the Console.

Help can also be shown automatically after writing a left pa
object. You can activate this behavior in *Preferences > He*

New to Spyder? Read our tutorial

```
How many terms? 3
Fibonacci sequence:
0
1
1
```



```
def recur_factorial(n):
   if n == 1:
       return n
   else:
       return n*recur_factorial(n-1)
# take input from the user
num = int(input("Enter a number: "))
# check is the number is negative
if num < 0:
   print("Sorry, factorial does not exist for negative numbers")
elif num == 0:
   print("The factorial of 0 is 1")
else:
   print("The factorial of",num,"is",recur_factorial(num))
```

Here you can get help of any object by pressing **Ctrl+I** in front of it, e
Editor or the Console.

Help can also be shown automatically after writing a left parenthesis
object. You can activate this behavior in *Preferences > Help*.

New to Spyder? Read our tutorial

```
In [28]: runfile('C:/Users/Student/.spyder-py3/temp.py', wdir='C:/Users/Student/.spyder

Enter a number: 23
The factorial of 23 is 25852016738884976640000

In [29]: S
```

2.FLASK PROGRAMS:

```
>>> import emoji

>>> print(emoji.emojize('Python is :thumbs_up:'))

Python is ⍰

>>> print(emoji.emojize('Python is :thumbsup:', language='alias'))




Python is ⍰

>>> print(emoji.demojize('Python is ⍰'))




Python is :thumbs_up:

>>> print(emoji.emojize("Python is fun :red_heart:"))




Python is fun ❤

>>> print(emoji.emojize("Python is fun :red_heart:", variant="emoji_type"))


Python is fun ❤ #red heart, not black heart

>>> print(emoji.is_emoji("⍰"))

True
```

```
>>> print(emoji.emojize('Python es :pulgar_hacia_arriba:', language='es'))

Python es ⍰

>>> print(emoji.demojize('Python es ⍰', language='es'))

Python es :pulgar_hacia_arriba:

>>> print(emoji.emojize("Python é :polegar_para_cima:", language='pt'))

Python é ⍰

>>> print(emoji.demojize("Python é ⍰", language='pt'))

Python é :polegar_para_cima:
```