



```
File Edit Search Source Run Debug Console Projects Tools View Help  
File Edit Search Source Run Debug Console Projects Tools View Help  
C:\Users\DELL\PycharmProjects\Employee.py  
  
2     employee_num=0  
3     @classmethod  
4     def employee_number(cls):  
5         print("employee count=",cls.employee_num)  
6     def __init__(self,name,age,gendr,salary):  
7         self.name=name  
8         self.age=age  
9         self.gendr=gendr  
10        self.salary=salary  
11        employee.employee_num +=1  
12    def totalsalary(self,target):  
13        totalsalary=self.salary + target  
14        print( "totalsalary=",totalsalary)  
15    def printall(self):  
16        print( "name=",self.name)  
17        print( "age=", self.age)  
18        print( "gendr=", self.gendr)  
19        print( "salary=", self.salary)  
20    def information():  
21        print("I-work hard")  
22 emp1=employee("ahmed",30,"male",3000)  
23 emp1.printall()  
24 emp1.totalsalary(1000)  
25 emp2=employee("ehab",36,"male",4000)  
26 emp2.printall()  
27 emp2.totalsalary(1500)  
28  
29 employee.employee_number()  
30 employee.information()
```



Search





C:\Users\DELL\untitled0.py

C:\Users\DELL

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Fri Oct 18 20:32:24 2024
4
5 @author: EMAN
6 """
7
8 class person:
9     def __init__(self, name, age, gender):
10         self.name=name
11         self.age=age
12         self.gender=gender
13     def printall(self):
14         print(self.name)
15         print(self.age)
16         print(self.gender)
17 class Employee (person):
18     def __init__(self, name, age, gender, salary):
19         person.__init__(self, name, age, gender)
20         self.salary=salary
21     def printall(self):
22         print(self.name)
23         print(self.age)
24         print(self.gender)
25         print(self.salary)
26
27 emp1= Employee("ahmed", 23, "male", 2000)
28 emp1.printall()
```

Source

Console

Object

## Usage

Here you can get help of any object  
Ctrl+I in front of it, either on the Edit  
Console.

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

New to Spyder? Read our tutorial

Help Variable Explorer Plots Files

Console 1/A

```
In [2]: runfile('C:/Users/Dell/untitled0.py', w)
ahmed
23
male

In [3]: runfile('C:/Users/Dell/untitled0.py', w)
ahmed
23
male
2000

In [4]:
```

ipython Console History

conda (Python 3.11.5) Completions: conda LSP: Python



Search





users\Dell\spyder-py3\temp.py

Source Console

```
1 class employee:
2     employee_num=0
3     @classmethod
4     def employee_number(cls):
5         print("employee count=",cls.employee_num)
6     def __init__(self,name,age,gendr,salary):
7         self.name=name
8         self.age=age
9         self.gendr=gendr
10        self.salary=salary
11        employee.employee_num +=1
12    def totalsalary(self,target):
13        totalsalary=self.salary + target
14        print( "totalsalary=",totalsalary)
15    def printall(self):
16        print( "name=",self.name)
17        print( "age=", self.age)
18        print( "gendr=", self.gendr)
19        print( "salary=", self.salary)
20    emp1=employee("ahmed",30,"male",3000)
21    emp1.printall()
22    emp1.totalsalary(1000)
23    emp2=employee("ehab",36,"male",4000)
24    emp2.printall()
25    emp2.totalsalary(1500)
26
27    employee.employee_number()
```

```
name= ahmed
age= 30
gendr= male
salary= 3000
totalsalary= 4000
name= ehab
age= 36
gendr= male
salary= 4000
totalsalary= 5500
employee count= 2
```

In [23]:



```
26
27     empl= Employee("ahmed",23,"male",2000)
28     empl.printall()
29
30     class manager:
31         def __init__(self,name,id,location):
32             self.name=name
33             self.id=id
34             self.location=location
35         def printall(self):
36             print(self.name)
37             print(self.id)
38             print(self.location)
39     class ceo(Employee,manager):
40         def __init__(self,name,id,location,age,gender,salary):
41             Employee.__init__(self,name,age,gender,salary)
42             manager.__init__(self,name,id,location)
43         def printall(self):
44             print(self.name)
45             print(self.age)
46             print(self.gender)
47             print(self.salary)
48             print(self.id)
49             print(self.location)
50     ceo1=ceo("ali",20230054,"egypt",40,"male",3000)
51     ceo1.printall()
52
53
54
```

Source Console Object

## 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 a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*

New to Spyder? Read our [tutorial](#)

Variable Explorer Plots Files

Console 1/A

```
In [4]: runfile('C:/Users/DELL/untitled0.py', wdir=
ahmed
23
male
2000
ali
40
male
3000
20230054
egypt
```

In [5]:

conda (Python 3.11.5) Completions: conda ✓ LSP: Python Line



Search



ENG

File Edit Search Source Run Debug Consoles Projects Tools View Help



C:\Users\Del\task.datastructure.py

temp.py X untitled0.py\* X task.datastructure.py X

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Wed Oct 23 19:29:34 2024
4
5 @author: EMAN
6 """
7
8 class calculation_1:
9     def add(self,num1,num2):
10         return (num1 +num2)
11
12 class calculation_2:
13     def multiplication(self,num1,num2):
14         return (num1*num2)
15 class calculation_3(calculation_1,calculation_2):
16     def division(self,num1,num2):
17         if num2!=0:
18             return (num1/num2)
19         else:
20             print("cannot divide by zero")
21 a=calculation_1()
22 a2=calculation_2()
23 a3=calculation_3()
24 print(a.add(2,3))
25 print(a2.multiplication(3,4))
26 print(a3.multiplication(2,3))
27 print(a3.division(2,3))
28
29
```



22°C  
عائم عالي



Search





C:\Users\DELL\untitled0.py

temp.py X unitled0.py\* X task.datastructure.py X

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Wed Oct 23 20:40:15 2024
4
5 @author: EMAN
6 """
7
8 class liquid:
9     def __init__(self, name, formula):
10         self.name = name
11         self.formula = formula
12     def info(self):
13         print(self.name, "", self.formula, "")
14     def property(self, usage):
15         self.usage = usage
16         print(self.usage, "")
17 class water(liquid):
18     def __init__(self, name, formula):
19         liquid.__init__(self, name, formula)
20     def usageofwater(self):
21         return print("water is important for life and used for c
22 a=liquid("benzene","C6H6")
23 a.info()
24 a.property("It is fuel to machines")
25 a2=water("water","H2O")
26 a2.info()
27 a2.usageofwater()
28
29
30
```



22°C

file.pdf



Search

