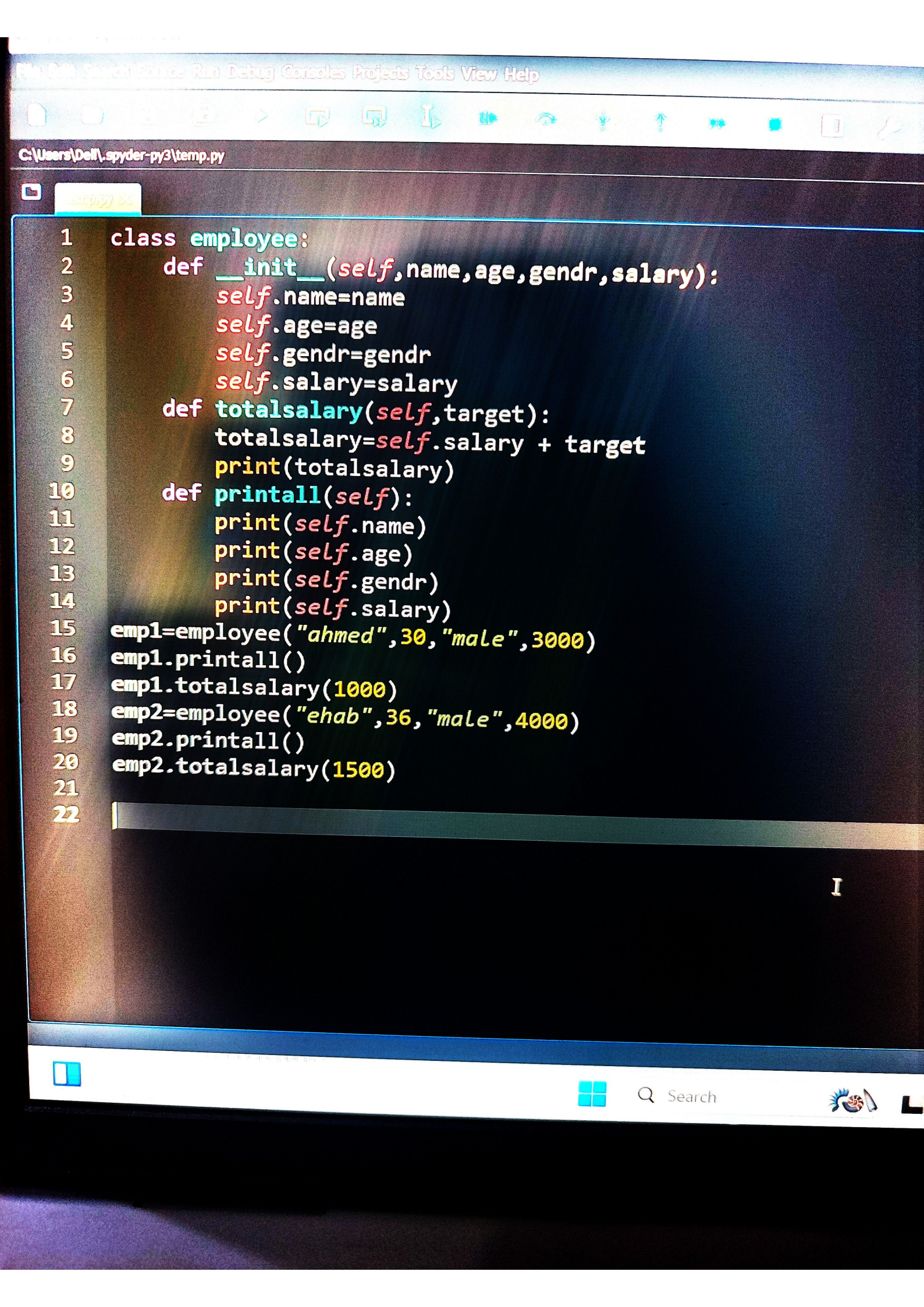
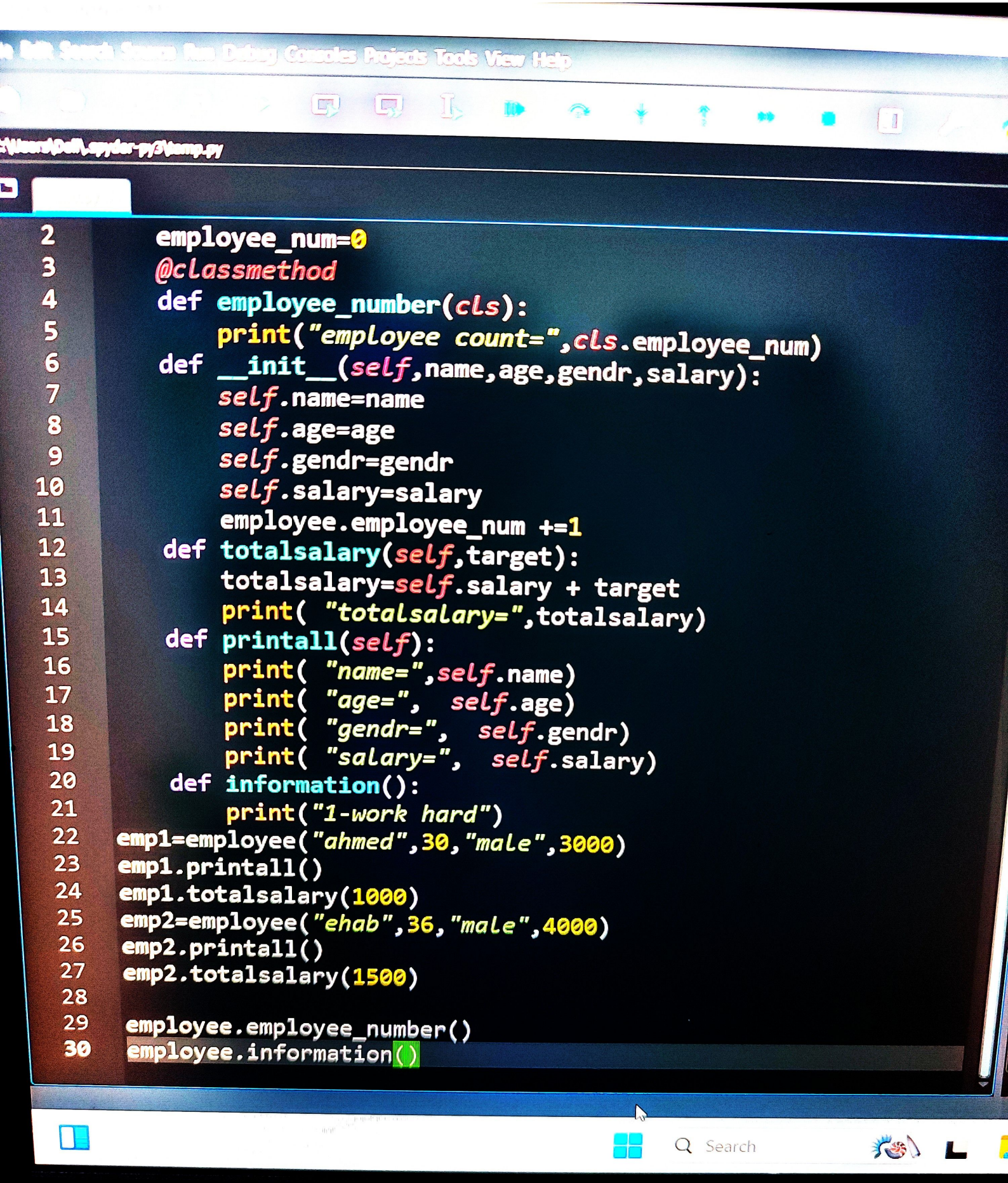
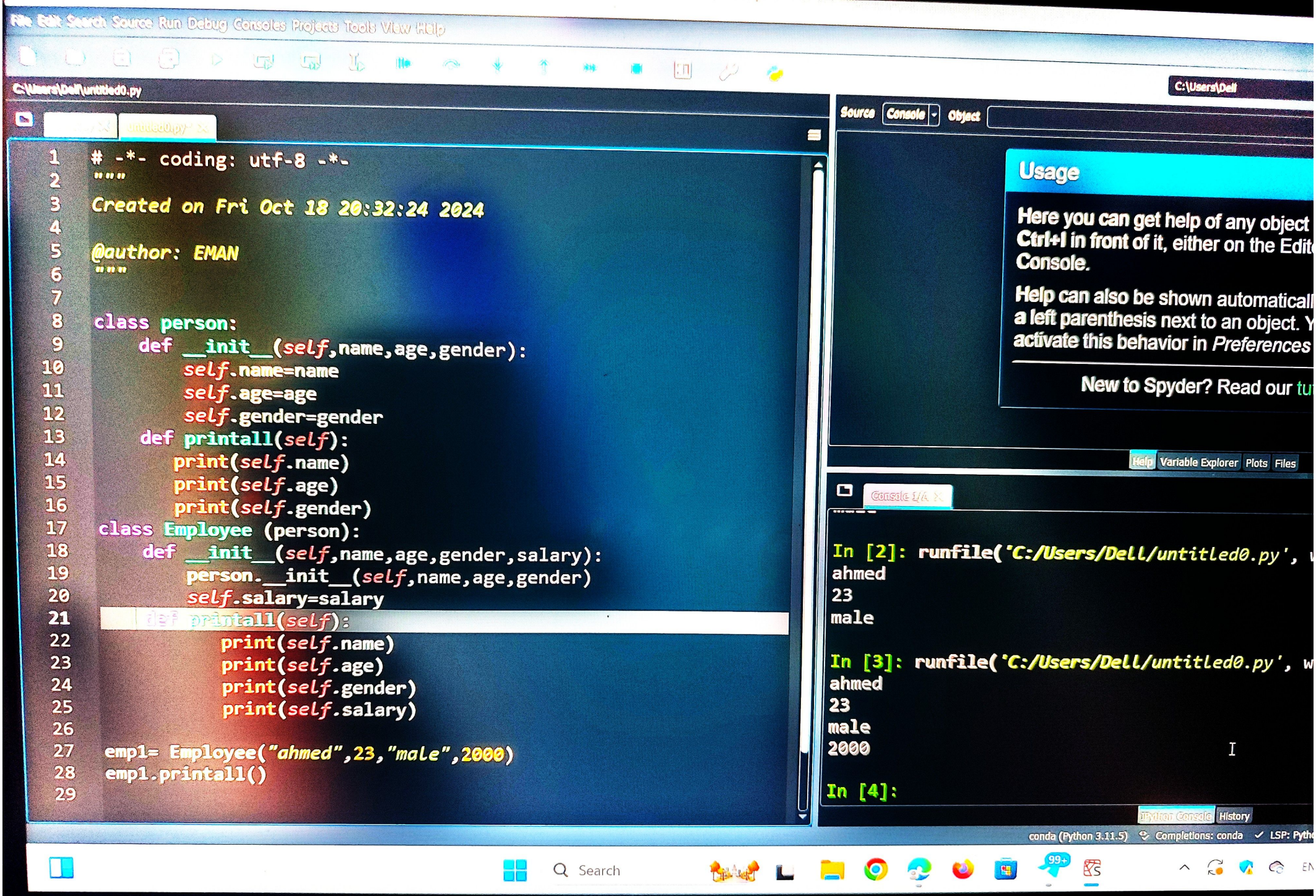
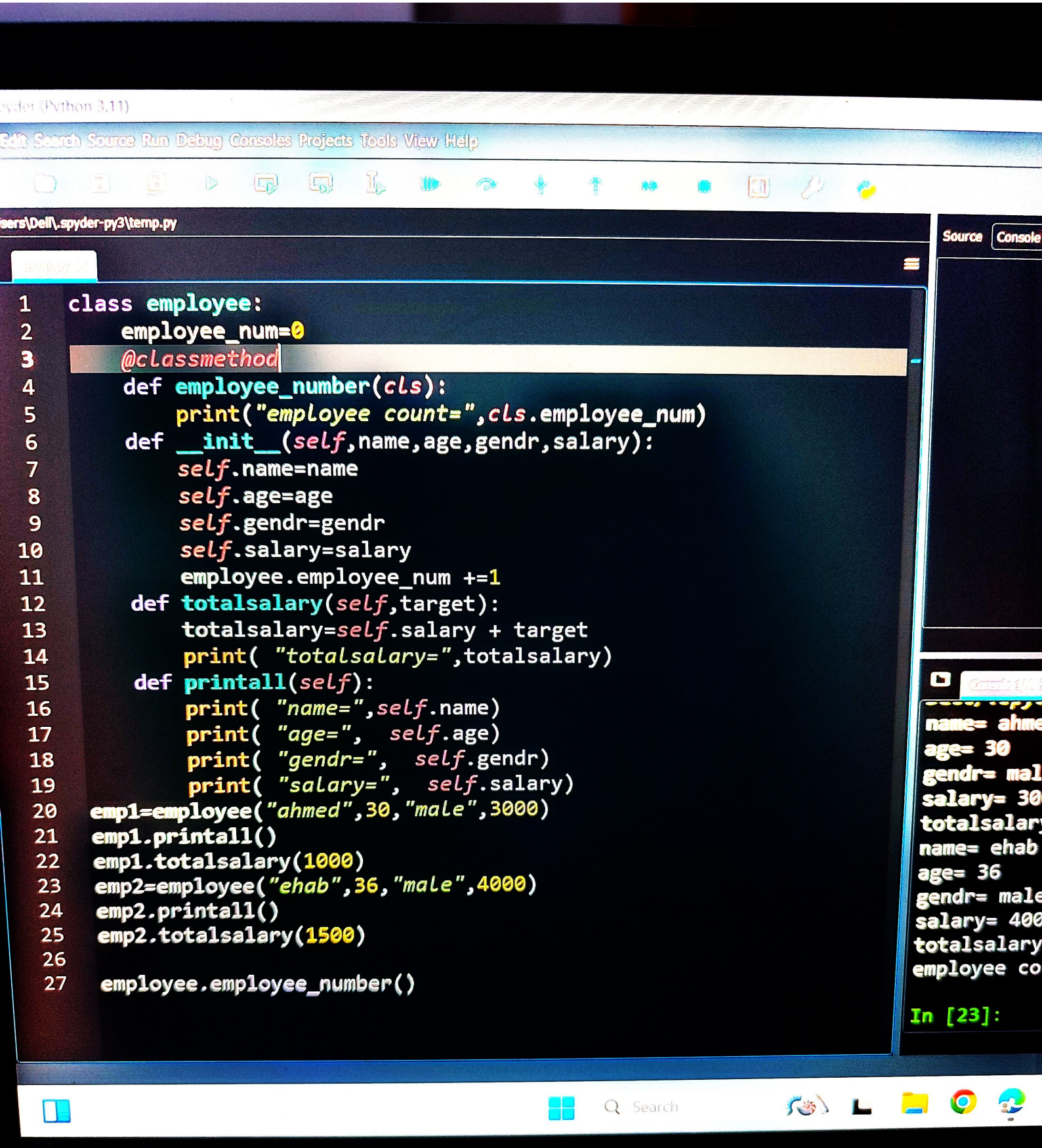

Edit Refactor Source Run Debug Consoles Projects Tools View Help
C:\Users\Dell\.spyder-py3\temp.py
temp.py
class employee:
def __init__(self,name,age,gendr,salary):
self.name=name
self.age=age
self.gendr=gendr
self.salary=salary
def totalsalary(self,target):
totalsalary=self.salary + target
print(totalsalary)
def printall(self):
print(self.name)
print(self.age)
print(self.gendr)
print(self.salary)
emp1=employee("ahmed",30,"male",3000)
emp1.printall()
emp1.totalsalary(1000)
emp2=employee("ehab",36,"male",4000)
emp2.printall()
emp2.totalsalary(1500)
Search

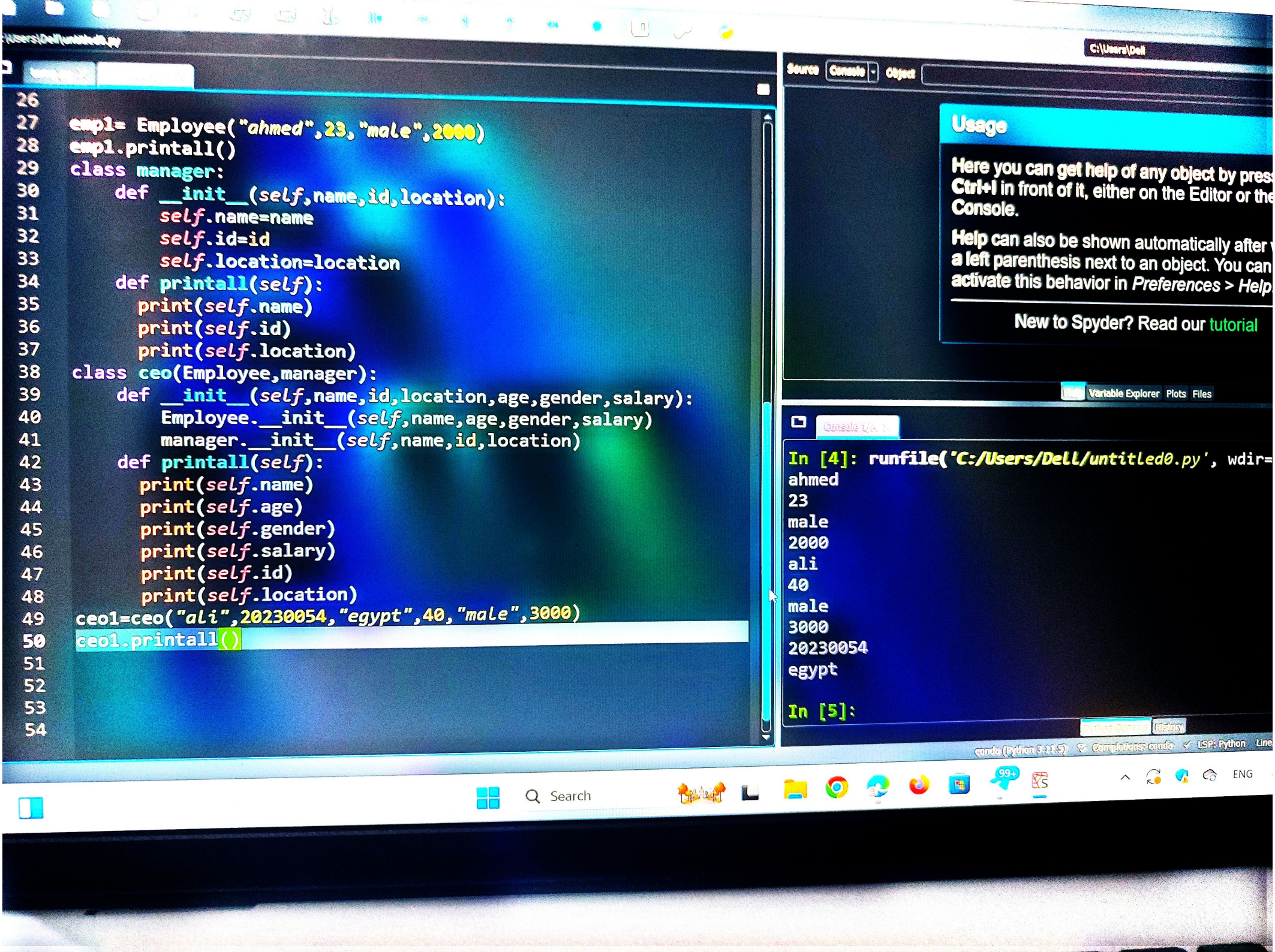


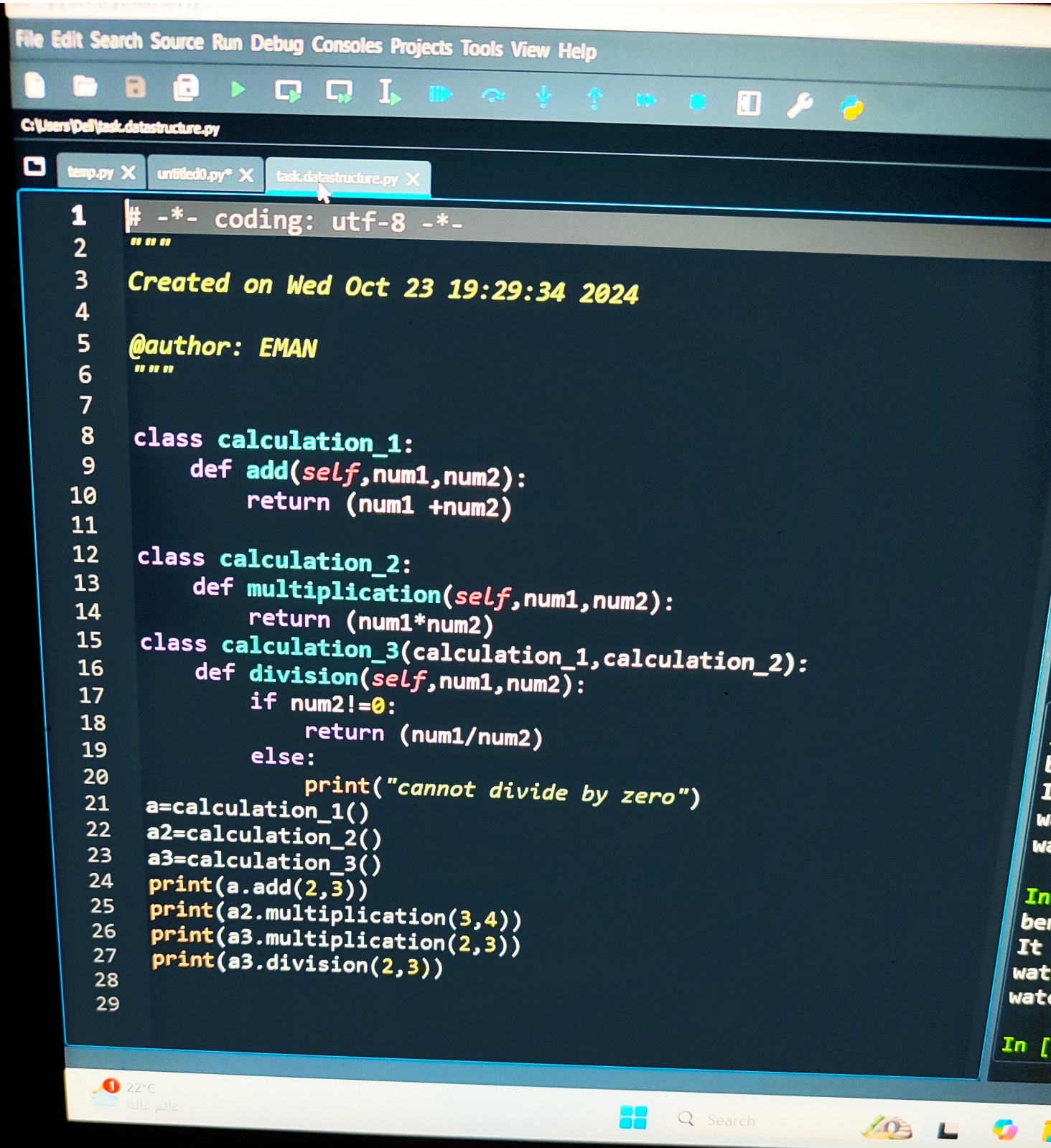
```
    employee_num=0
    @classmethod
    def employee_number(cls):
        print("employee count=",cls.employee_num)
    def __init__(self,name,age,gendr,salary):
        self.name=name
        self.age=age
        self.gendr=gendr
        self.salary=salary
        employee.employee_num +=1
    def totalsalary(self,target):
        totalsalary=self.salary + target
        print( "totalsalary=",totalsalary)
    def printall(self):
        print( "name=",self.name)
        print( "age=",  self.age)
        print( "gendr=",  self.gendr)
        print( "salary=",  self.salary)
    def information():
        print("1-work hard")
emp1=employee("ahmed",30,"male",3000)
emp1.printall()
emp1.totalsalary(1000)
emp2=employee("ehab",36,"male",4000)
emp2.printall()
emp2.totalsalary(1500)

employee.employee_number()
employee.information()
```

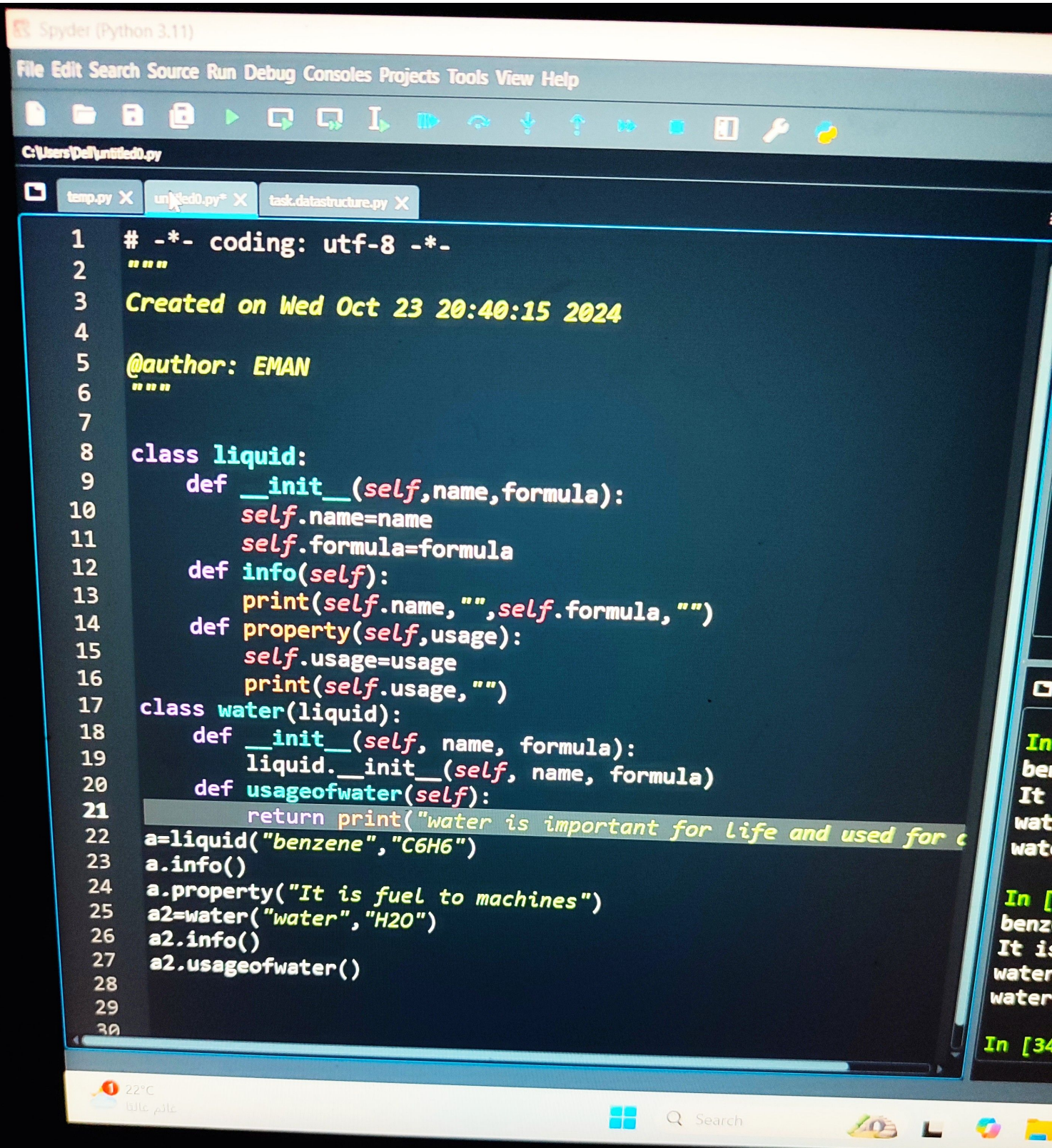



```python
# -*- coding: utf-8 -*-
"""
Created on Fri Oct 18 20:32:24 2024

@author: EMAN
"""

class person:
    def __init__(self,name,age,gender):
        self.name=name
        self.age=age
        self.gender=gender
    def printall(self):
        print(self.name)
        print(self.age)
        print(self.gender)
class Employee (person):
    def __init__(self,name,age,gender,salary):
        person.__init__(self,name,age,gender)
        self.salary=salary
    def printall(self):
        print(self.name)
        print(self.age)
        print(self.gender)
        print(self.salary)

emp1= Employee("ahmed",23,"male",2000)
emp1.printall()

```


Edit Search Source Run Debug Consoles Projects Tools View Help
sers\Dell\.spyder-py3\temp.py
Source
Console
class employee:
employee_num=0
@classmethod
def employee_number(cls):
print("employee count=",cls.employee_num)
def __init__(self,name,age,gendr,salary):
self.name=name
self.age=age
self.gendr=gendr
self.salary=salary
employee.employee_num +=1
def totalsalary(self,target):
totalsalary=self.salary + target
print("totalsalary=",totalsalary)
def printall(self):
print("name=",self.name)
print("age=", self.age)
print("gendr=", self.gendr)
print("salary=", self.salary)
emp1=employee("ahmed",30,"male",3000)
emp1.printall()
emp1.totalsalary(1000)
emp2=employee("ehab",36,"male",4000)
emp2.printall()
emp2.totalsalary(1500)
employee.employee_number()
name= ahme
age= 30
gendr= mal
salary= 30
totalsalary
name= ehab
age= 36
gendr= male
salary= 400
totalsalary
employee co
In [23]:
Search


empl= Employee("ahmed",23,"male",2000)
empl.printall()
class manager:
def __init__(self,name,id,location):
self.name=name
self.id=id
self.location=location
def printall(self):
print(self.name)
print(self.id)
print(self.location)
class ceo(Employee,manager):
def __init__(self,name,id,location,age,gender,salary):
Employee.__init__(self,name,age,gender,salary)
manager.__init__(self,name,id,location)
def printall(self):
print(self.name)
print(self.age)
print(self.gender)
print(self.salary)
print(self.id)
print(self.location)
ceo1=ceo("ali",20230054,"egypt",40,"male",3000)
ceo1.printall()
Usage
Here you can get help of any object by press
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
In [4]: runfile('C:/Users/Dell/untitled0.py', wdir=
ahmed
23
male
2000
ali
40
male
3000
20230054
egypt
In [5]:
Search
ENG


File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\Dell\task.datastructure.py
temp.py
untitled0.py*
task.datastructure.py
-*- coding: utf-8 -*-
"""
Created on Wed Oct 23 19:29:34 2024
@author: EMAN
"""
class calculation_1:
def add(self,num1,num2):
return (num1 +num2)
class calculation_2:
def multiplication(self,num1,num2):
return (num1*num2)
class calculation_3(calculation_1,calculation_2):
def division(self,num1,num2):
if num2!=0:
return (num1/num2)
else:
print("cannot divide by zero")
a=calculation_1()
a2=calculation_2()
a3=calculation_3()
print(a.add(2,3))
print(a2.multiplication(3,4))
print(a3.multiplication(2,3))
print(a3.division(2,3))
22°C
Search


Spyder (Python 3.11)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\Dell\untitled0.py
temp.py
untitled0.py*
task.datastructure.py
-*- coding: utf-8 -*-
"""
Created on Wed Oct 23 20:40:15 2024
@author: EMAN
"""
class liquid:
def __init__(self,name,formula):
self.name=name
self.formula=formula
def info(self):
print(self.name,"",self.formula,"")
def property(self,usage):
self.usage=usage
print(self.usage,"")
class water(liquid):
def __init__(self, name, formula):
liquid.__init__(self, name, formula)
def usageofwater(self):
return print("water is important for life and used for c
a=liquid("benzene","C6H6")
a.info()
a.property("It is fuel to machines")
a2=water("water","H2O")
a2.info()
a2.usageofwater()
22°C
Search