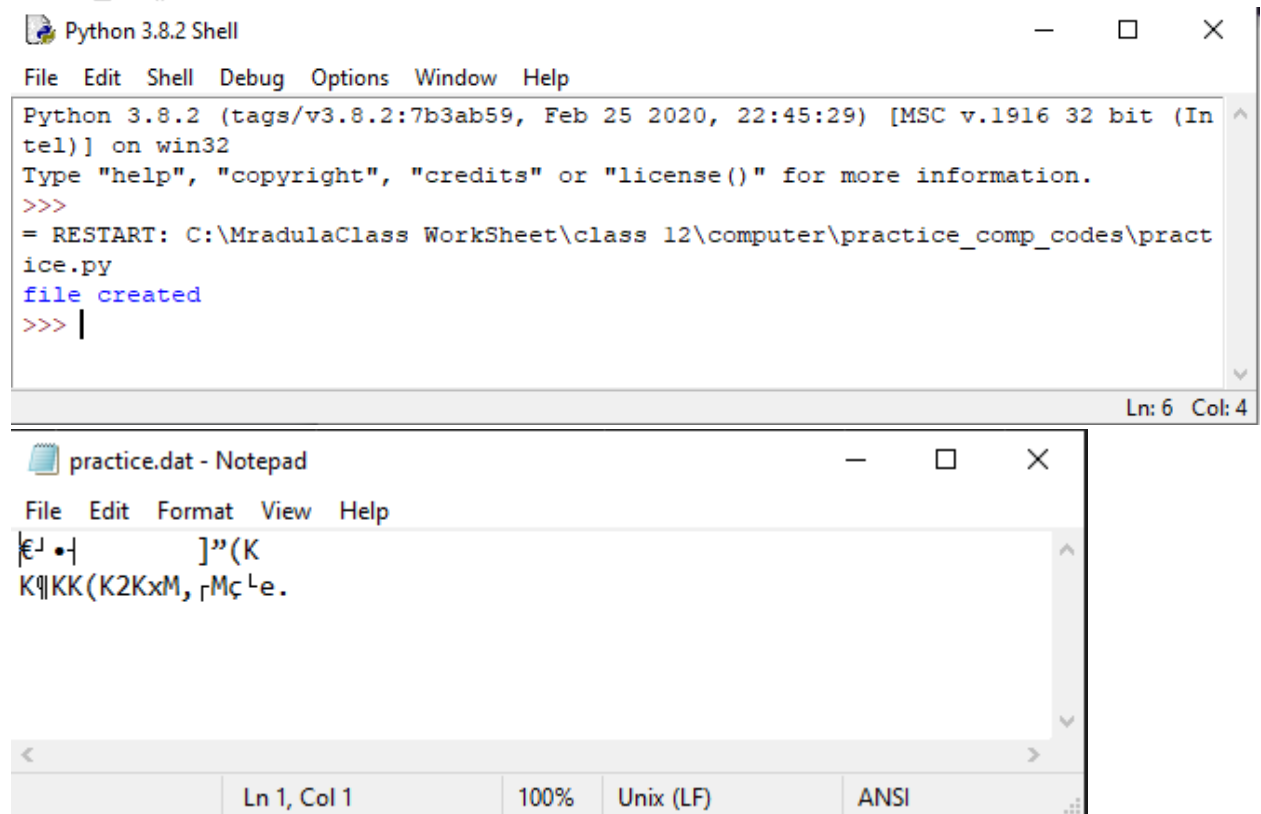


binary file

1. a program to write list sequence in a binary file

```
import pickle
def create_bin():
    mylist=[10,20,30,40,50,120,300,999]
    f=open("practice.dat","wb")
    pickle.dump(mylist,f)
    print("file created")
create_bin()
```



The screenshot shows two windows. The top window is titled 'Python 3.8.2 Shell' and displays the execution of the provided Python code. The output shows the file 'practice.dat' being created successfully. The bottom window is titled 'practice.dat - Notepad' and shows the binary content of the file, which appears as a series of non-readable characters.

```
Python 3.8.2 Shell
File Edit Shell Debug Options Window Help
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\MradulaClass WorkSheet\class 12\computer\practice_comp_codes\practice.py
file created
>>> |
```

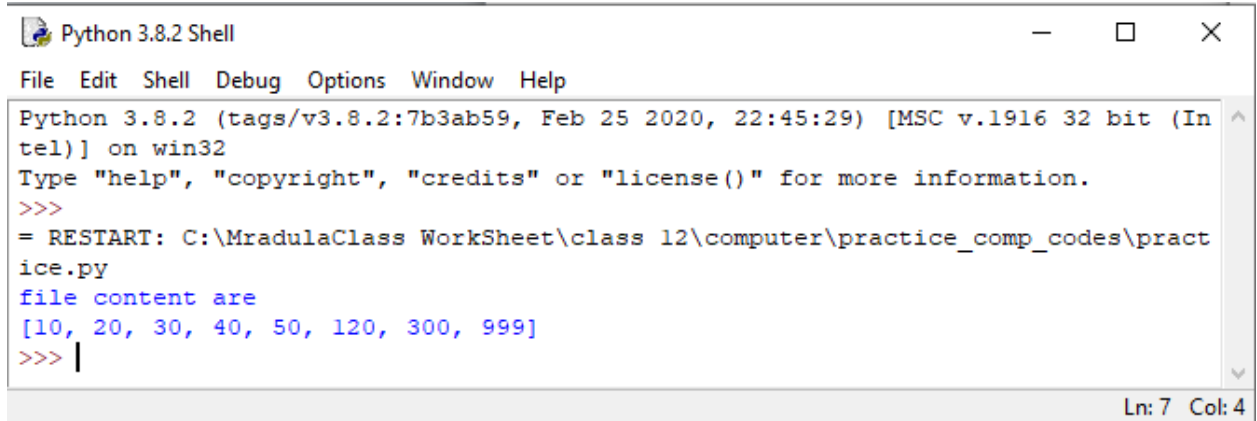
Ln: 6 Col: 4

```
practice.dat - Notepad
File Edit Format View Help
KqKK(K2KxM, rMç Le.
```

Ln 1, Col 1 100% Unix (LF) ANSI

2. pickle.load() method is used to read the binary file.

```
import pickle
def read_bin():
    f=open("practice.dat","rb")
    mylist=pickle.load(f)
    print("file content are")
    print(mylist)
    f.close()
read_bin()
```



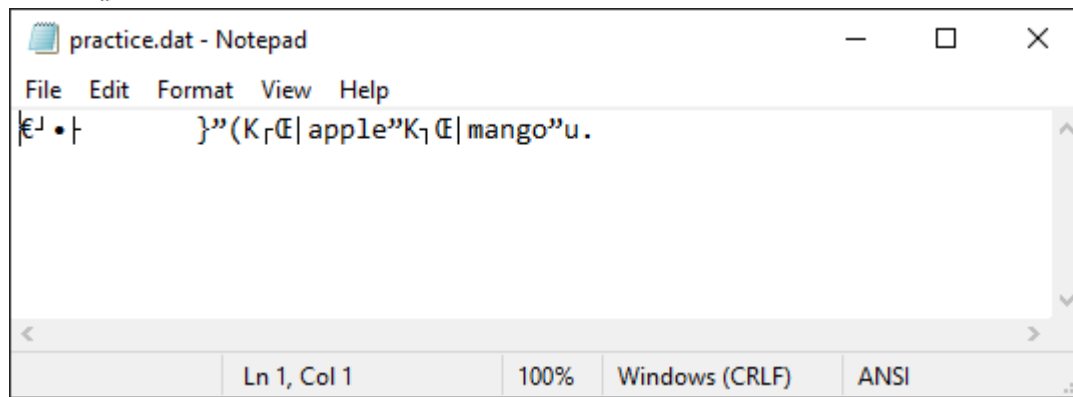
The screenshot shows a Python 3.8.2 Shell window with the following content:

```
Python 3.8.2 Shell
File Edit Shell Debug Options Window Help
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\MradulaClass WorkSheet\class 12\computer\practice_comp_codes\practice.py
file content are
[10, 20, 30, 40, 50, 120, 300, 999]
>>> |
```

The status bar at the bottom right indicates "Ln: 7 Col: 4".

3. using dump method

```
import pickle  
f=open("practice.dat","wb")  
d={1:"apple",2:"mango"}  
pickle.dump(d,f)  
f.close()
```



- ```
import pickle
f=open("practice.dat","wb")
a="new thing"
l=[10,20,30,40]
d={1:"apple",2:"mango"}
pickle.dump(a,f)
pickle.dump(l,f)
pickle.dump(d,f)
f.close()
```

]'(K  
K||KK(e.€' •| }'|(K\_€|apple"K\_€|mango"u.' The status bar at the bottom shows 'Ln 1, Col 1', '100%', 'Macintosh (CR)', and 'ANSI'."/>

practice.dat - Notepad

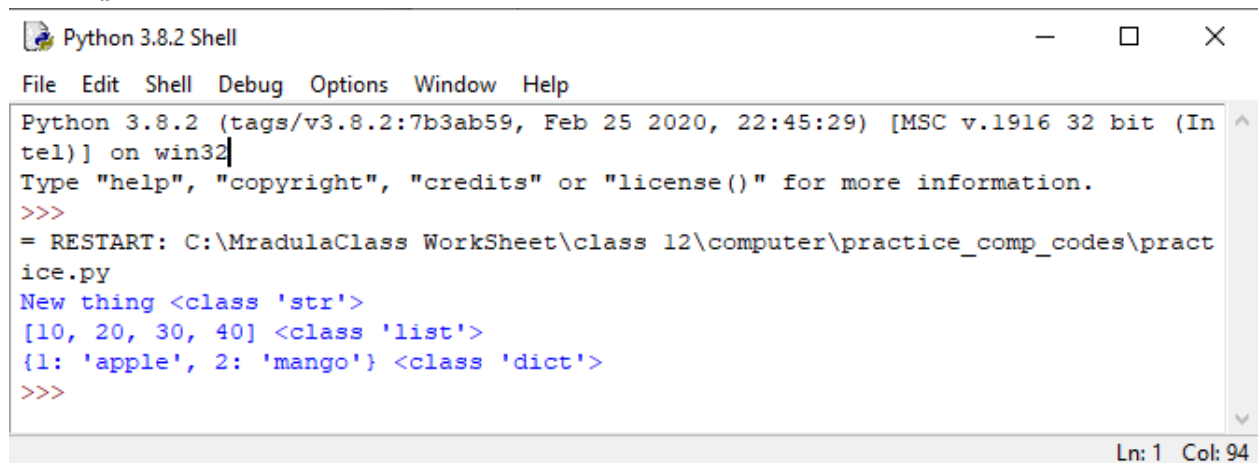
File Edit Format View Help

€' •  
€ New thing".€' •  
]'(K  
K||KK(e.€' •| }'|(K\_€|apple"K\_€|mango"u.'

Ln 1, Col 1 100% Macintosh (CR) ANSI

5. reading data from a binary file (unpickling)

```
import pickle
f=open("practice.dat","rb")
m=pickle.load(f)
print(m,type(m))
n=pickle.load(f)
print(n,type(n))
o=pickle.load(f)
print(o,type(o))
f.close()
```



The screenshot shows a Python 3.8.2 Shell window with a menu bar (File, Edit, Shell, Debug, Options, Window, Help) and a toolbar with standard window controls. The main text area displays the output of a script. It starts with the Python version and architecture information, followed by a prompt to type 'help', 'copyright', 'credits', or 'license()'. The script then restarts at the file path 'C:\MradulaClass WorkSheet\class 12\computer\practice\_comp\_codes\practice.py'. The output shows three objects loaded from the file: a string 'New thing', a list [10, 20, 30, 40], and a dictionary {'1': 'apple', '2': 'mango'}. The status bar at the bottom right indicates 'Ln: 1 Col: 94'.

```
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\MradulaClass WorkSheet\class 12\computer\practice_comp_codes\practice.py
New thing <class 'str'>
[10, 20, 30, 40] <class 'list'>
{'1': 'apple', '2': 'mango'} <class 'dict'>
>>>
```

Ln: 1 Col: 94

6. how to read data from binary file where no of records are not known.

```
import pickle
f=open("practice.dat","rb")
while(true):
 obj=pickle.load(f)
 print(obj)
 f.close()
 break;
```

Python 3.8.2 Shell

File Edit Shell Debug Options Window Help

Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32

Type "help", "copyright", "credits" or "license()" for more information.

>>>

= RESTART: C:\MradulaClass WorkSheet\class 12\computer\practice\_comp\_codes\practice.py

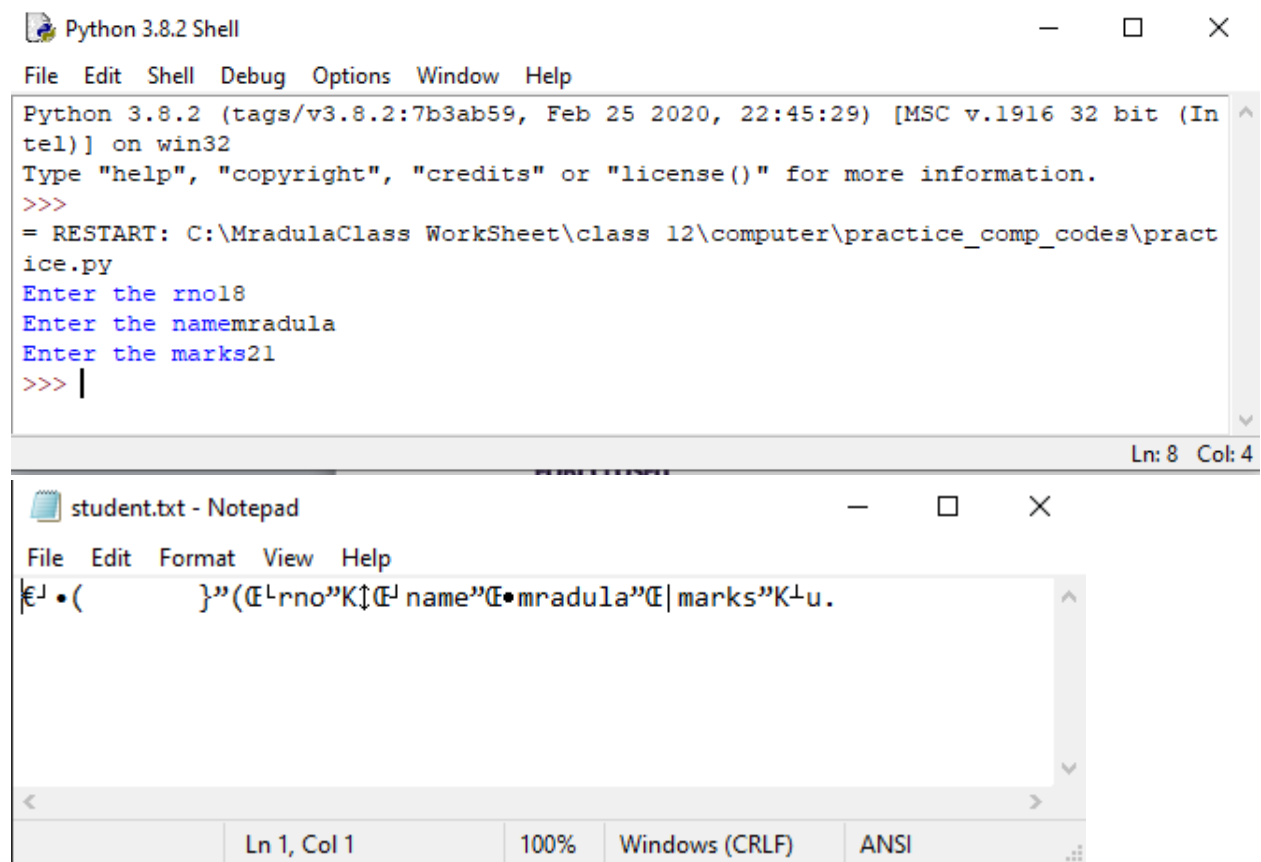
New thing

>>> |

Ln: 6 Col: 4

7. udf save dictionary object containing {rno, name, marks} to a pickled file student.txt.

```
import pickle
def writefile():
 d={}
 fobj=open("student.txt","ab")
 d["rno"]=int(input('enter the rno'))
 d["name"]=input('enter the name')
 d["marks"]=int(input('enter the marks'))
 pickle.dump(d, fobj)
 fobj.close()
writefile()
```



The screenshot displays two windows. The top window is titled "Python 3.8.2 Shell" and shows the execution of a Python script. The script defines a function `writefile()` that creates a dictionary `d` with keys `"rno"`, `"name"`, and `"marks"`, prompts the user for values, and saves the dictionary to a file named `student.txt` in append mode using `pickle.dump()`. The script is then executed, and the user enters `18` for `rno`, `mradula` for `name`, and `21` for `marks`. The bottom window is titled "student.txt - Notepad" and shows the contents of the file, which is a pickled dictionary object: `{'rno': 18, 'name': 'mradula', 'marks': 21}`.

```
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\MradulaClass WorkSheet\class 12\computer\practice_comp_codes\practice.py
Enter the rno18
Enter the namemradula
Enter the marks21
>>> |
```

Ln: 8 Col: 4

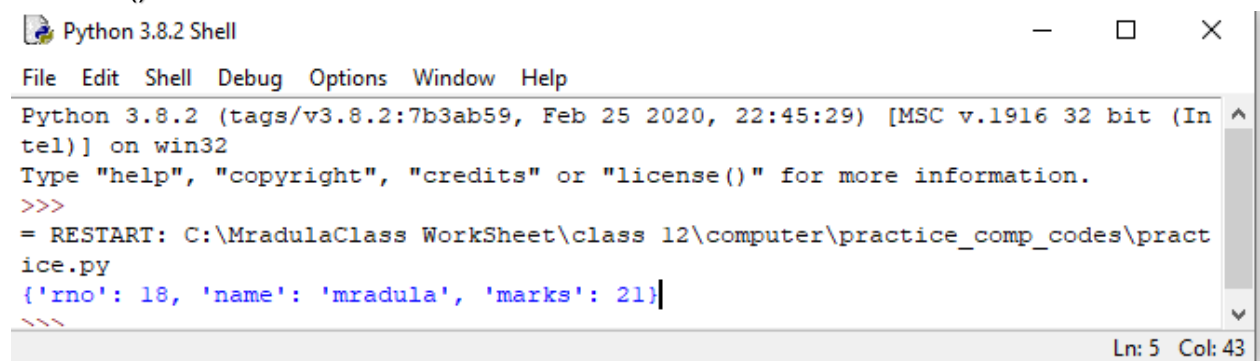
```
student.txt - Notepad
File Edit Format View Help
{'rno': 18, 'name': 'mradula', 'marks': 21}
```

Ln 1, Col 1 100% Windows (CRLF) ANSI

8. udf read all objects from file student.txt

```
import pickle
def readdata():
 f=open("student.txt","rb")
 while(true):
 try:
 obj=pickle.load(f)
 print(obj)
 except eoferror:
 f.close()
 break
```

readdata()



```
Python 3.8.2 Shell
File Edit Shell Debug Options Window Help
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\MradulaClass WorkSheet\class 12\computer\practice_comp_codes\practice.py
{'rno': 18, 'name': 'mradula', 'marks': 21}
```

Ln: 5 Col: 43



9. implement menu driven

```
import pickle
def writefile():
 d={}
 fobj=open("student.txt","ab")
 d["rno"]=int(input('enter the rno'))
 d["name"]=input('enter the name')
 d["marks"]=int(input('enter the marks'))
 pickle.dump(d, fobj)
 fobj.close()
```

```
def readdata():
 f=open("student.txt","rb")
 while(true):
 try:
 obj=pickle.load(f)
 print(obj)
 except eoferror:
 f.close()
 break
```

```
def menu():
 while(true):
 print("1: write data")
 print("2: read data")
 print("3: exit")
 ch=int(input('enter the choice'))
 if(ch==1):
 writefile()
 elif(ch==2):
 readdata()
 elif(ch==3):
 break
 menu()
```

Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32

Type "help", "copyright", "credits" or "license()" for more information.

>>>

= RESTART: C:\MradulaClass WorkSheet\class 12\computer\practice\_comp\_codes\practice.py

1: Write data

2: Read data

3: Exit

Enter the choice1

Enter the rno13

Enter the namecherry

Enter the marks23

1: Write data

2: Read data

3: Exit

Enter the choice2

{'rno': 18, 'name': 'mradula', 'marks': 21}

{'rno': 13, 'name': 'cherry', 'marks': 23}

1: Write data

2: Read data

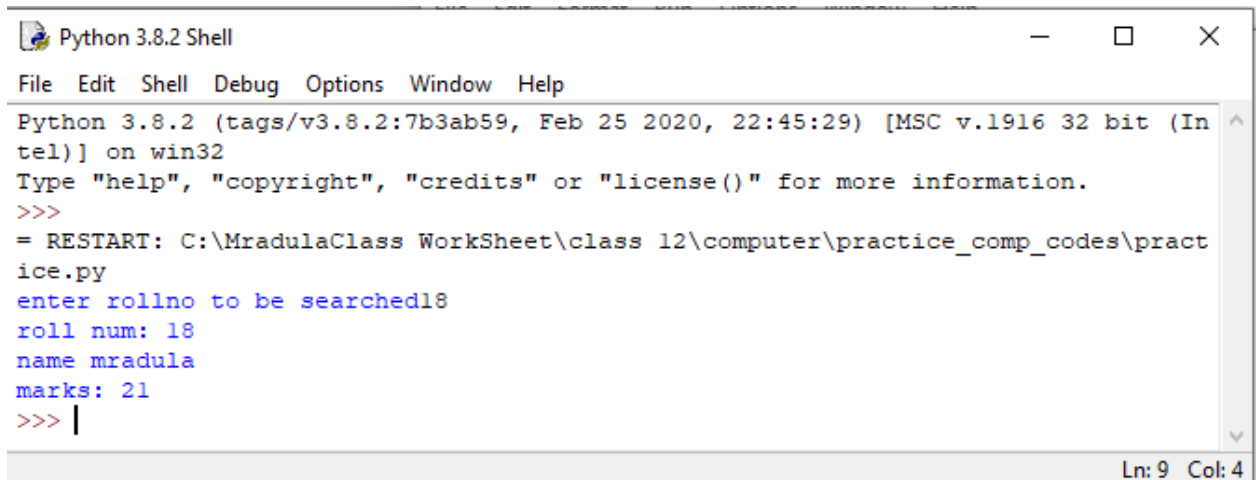
3: Exit

Enter the choice3

>>> |

10. search by roll no

```
import pickle
def search_rno():
 f=open("student.txt","rb")
 flag = False
 r=int(input("enter rollno to be searched"))
 while True:
 try:
 rec=pickle.load(f)
 if rec['rno'] == r:
 print("roll num:",rec["rno"])
 print("name",rec["name"])
 print("marks:",rec["marks"])
 flag = True
 except EOFError:
 break
 if flag == False:
 print("no records found")
 f.close()
search_rno()
```



Python 3.8.2 Shell

File Edit Shell Debug Options Window Help

Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32

Type "help", "copyright", "credits" or "license()" for more information.

>>>

= RESTART: C:\MradulaClass WorkSheet\class 12\computer\practice\_comp\_codes\practice.py

enter rollno to be searched18

roll num: 18

name mradula

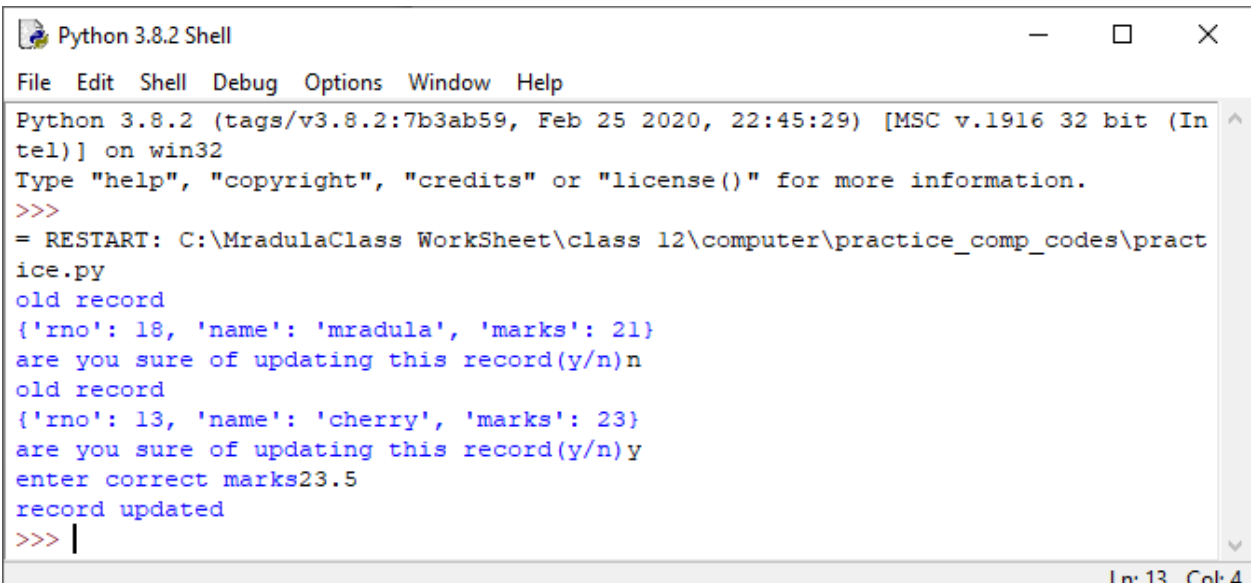
marks: 21

>>> |

Ln: 9 Col: 4

#### 11. update record

```
import pickle
f=open("student.txt","rb")
reclst=[]
while true:
 try:
 rec=pickle.load(f)
 reclst.append(rec)
 except eoferror:
 break
f.close()
flag=false
ans="n"
for i in range (len(reclst)):
 flag=true
 print("old record")
 print(reclst[i])
 ans=input("are you sure of updating this record(y/n)")
 if ans.lower()=="y":
 m=eval(input("enter correct marks"))
 reclst[i]["marks"]=m
if ans=='y':
 f=open("student.txt","wb")
 for x in reclst:
 pickle.dump(x,f)
 print("record updated")
f.close()
if flag==false:
 print("record not found")
```

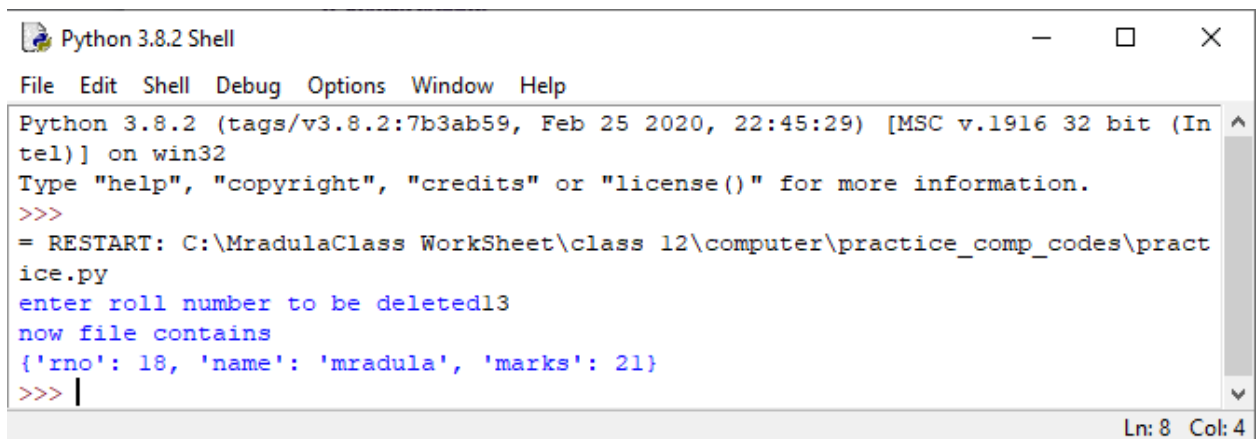


```
Python 3.8.2 Shell
File Edit Shell Debug Options Window Help
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\MradulaClass WorkSheet\class 12\computer\practice_comp_codes\practice.py
old record
{'rno': 18, 'name': 'mradula', 'marks': 21}
are you sure of updating this record(y/n)n
old record
{'rno': 13, 'name': 'cherry', 'marks': 23}
are you sure of updating this record(y/n)y
enter correct marks23.5
record updated
>>> |
```

Ln: 13 Col: 4

## 12. delete record

```
import pickle
f=open("student.txt","rb")
reclst=[]
r=int(input("enter roll number to be deleted"))
while true:
 try:
 rec=pickle.load(f)
 reclst.append(rec)
 except eoferror:
 break
f.close()
f=open("student.txt","wb")
for x in reclst:
 if x["rno"]==r:
 continue
 else:
 pickle.dump(x,f)
f.close()
print("now file contains")
f=open("student.txt","rb")
while true:
 try:
 obj=pickle.load(f)
 print(obj)
 except eoferror:
 f.close()
 break
```



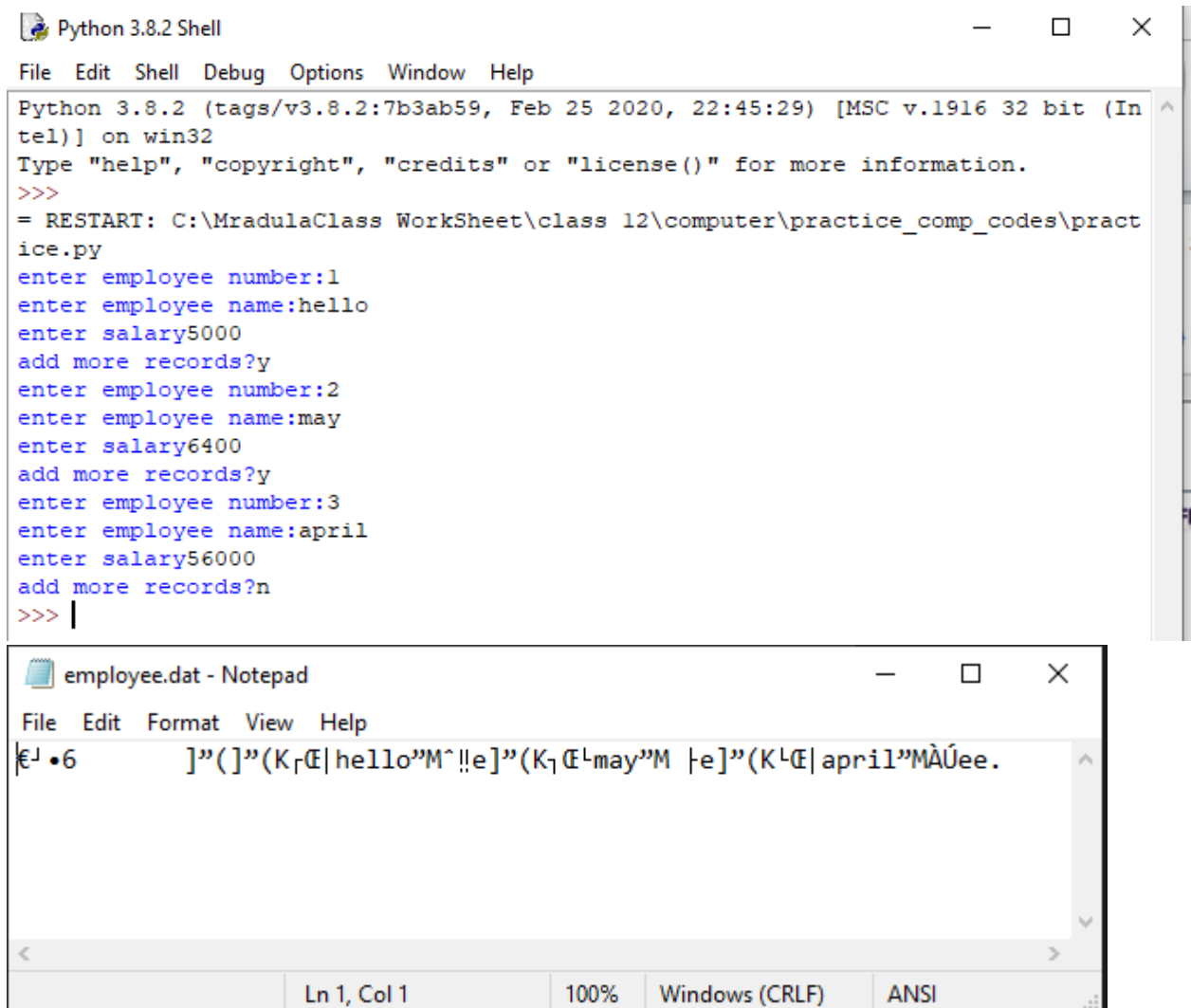
The screenshot shows a Python 3.8.2 Shell window with the following content:

```
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\MradulaClass WorkSheet\class 12\computer\practice_comp_codes\practice.py
enter roll number to be deleted13
now file contains
{'rno': 18, 'name': 'mradula', 'marks': 21}
>>> |
```

The status bar at the bottom right indicates "Ln: 8 Col: 4".

13. pickling – storing employee details in binary file

```
import pickle
emp=[]
f=open("employee.dat","wb")
ans="y"
while ans=="y":
 eno= int(input("enter employee number:"))
 name=input("enter employee name:")
 salary=int(input("enter salary"))
 emp.append([eno,name,salary])
 ans=input("add more records?")
pickle.dump(emp,f)
f.close()
```



The image shows two windows from a Windows operating system. The top window is titled "Python 3.8.2 Shell" and displays the execution of a Python script. The script prompts the user to enter employee details (number, name, salary) and asks if they want to add more records. The user enters three records: (1, 'hello', 5000), (2, 'may', 6400), and (3, 'april', 56000). The bottom window is titled "employee.dat - Notepad" and shows the binary content of the file created by the script, which is a pickled list of the employee records.

```
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\MradulaClass WorkSheet\class 12\computer\practice_comp_codes\practice.py
enter employee number:1
enter employee name:hello
enter salary5000
add more records?y
enter employee number:2
enter employee name:may
enter salary6400
add more records?y
enter employee number:3
enter employee name:april
enter salary56000
add more records?n
>>> |
```

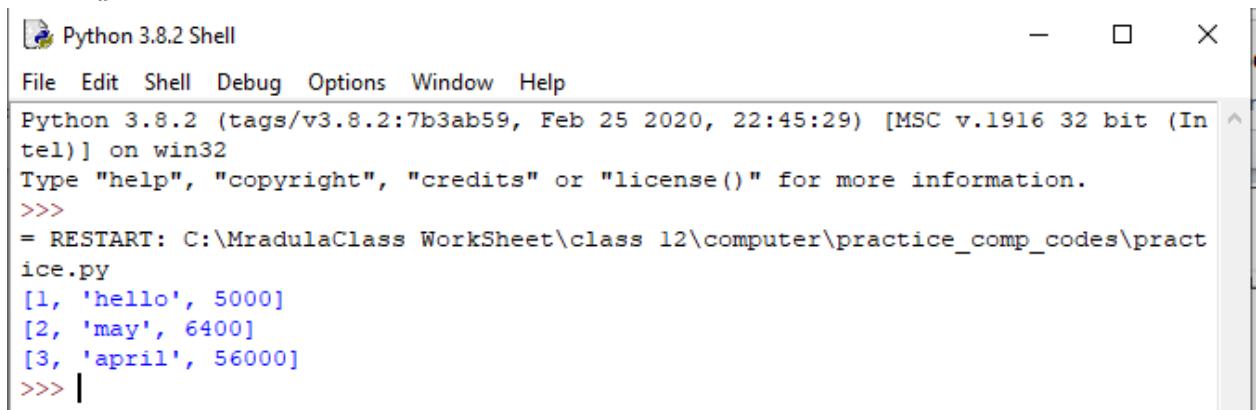
employee.dat - Notepad

```
File Edit Format View Help
[Binary content of employee.dat file]
```

Ln 1, Col 1      100%      Windows (CRLF)      ANSI

14. un-pickling – reading and display record

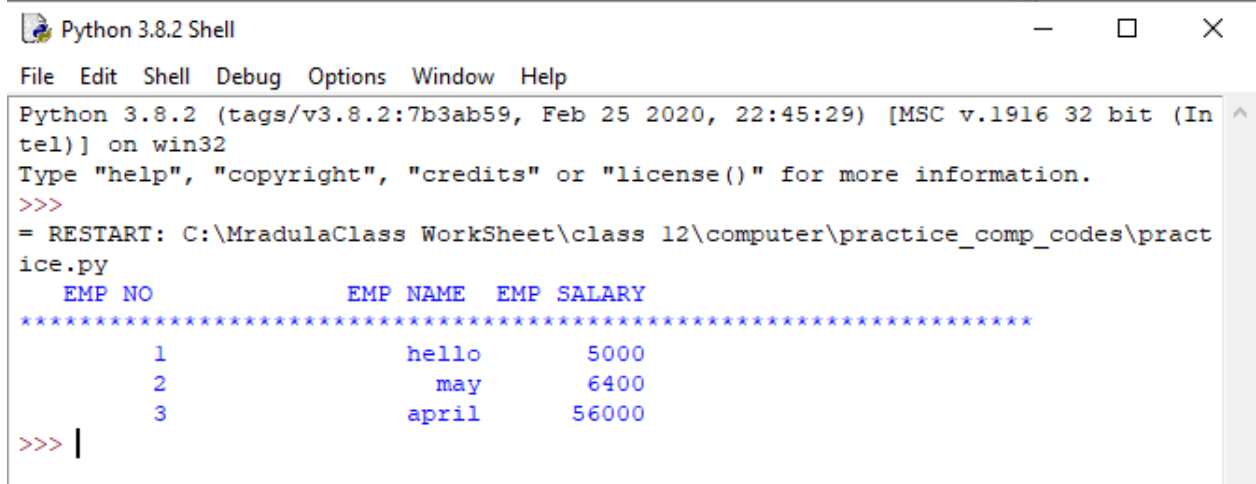
```
import pickle
emp=[]
f=open("employee.dat","rb")
ans="y"
while true:
 try:
 emp=pickle.load(f)
 except eoferror:
 break
for e in emp:
 print(e)
f.close()
```



```
Python 3.8.2 Shell
File Edit Shell Debug Options Window Help
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\MradulaClass WorkSheet\class 12\computer\practice_comp_codes\practice.py
[1, 'hello', 5000]
[2, 'may', 6400]
[3, 'april', 56000]
>>> |
```

15. un-pickling– display record (formatted output)

```
import pickle
emp=[]
f=open("employee.dat","rb")
ans="y"
while true:
 try:
 emp=pickle.load(f)
 except eoferror:
 break
print("%10s"%emp no ,"%20s"%emp name ,"%10s"%emp salary")
print("*****")
for e in emp:
 print("%10s"%e[0],"%20s"%e[1],"%10s"%e[2])
f.close()
```



The screenshot shows a Python 3.8.2 Shell window with the following content:

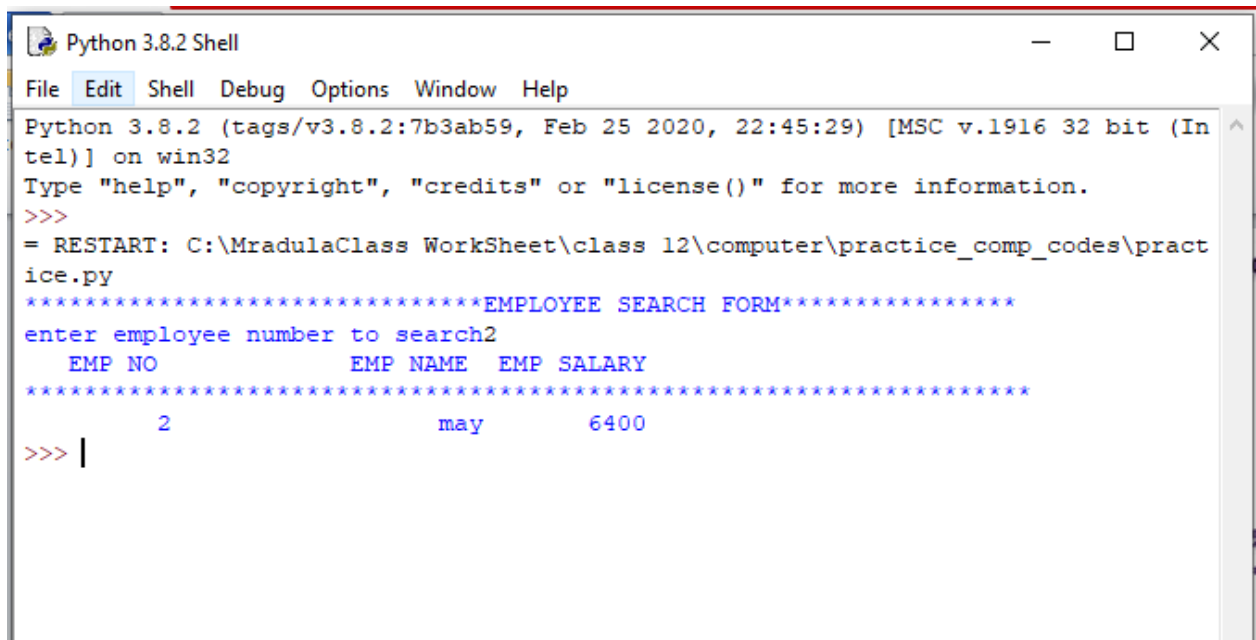
```
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\MradulaClass WorkSheet\class 12\computer\practice_comp_codes\practice.py
 EMP NO EMP NAME EMP SALARY

 1 hello 5000
 2 may 6400
 3 april 56000
>>> |
```



## 16. searching in binary file

```
import pickle
emp=[]
f=open("employee.dat","rb")
ans="y"
print("*****employee search form*****")
en=int(input("enter employee number to search"))
found=false
while true:
 try:
 emp=pickle.load(f)
 except eoferror:
 break
print("%10s"%emp no ","%20s"%emp name ","%10s"%emp salary")
print("*****")
for e in emp:
 if (e[0]==en):
 print("%10s"%e[0],"%20s"%e[1],"%10s"%e[2])
 found=true
 break
if found==false:
 print("##sorry employee number not found##")
f.close()
```

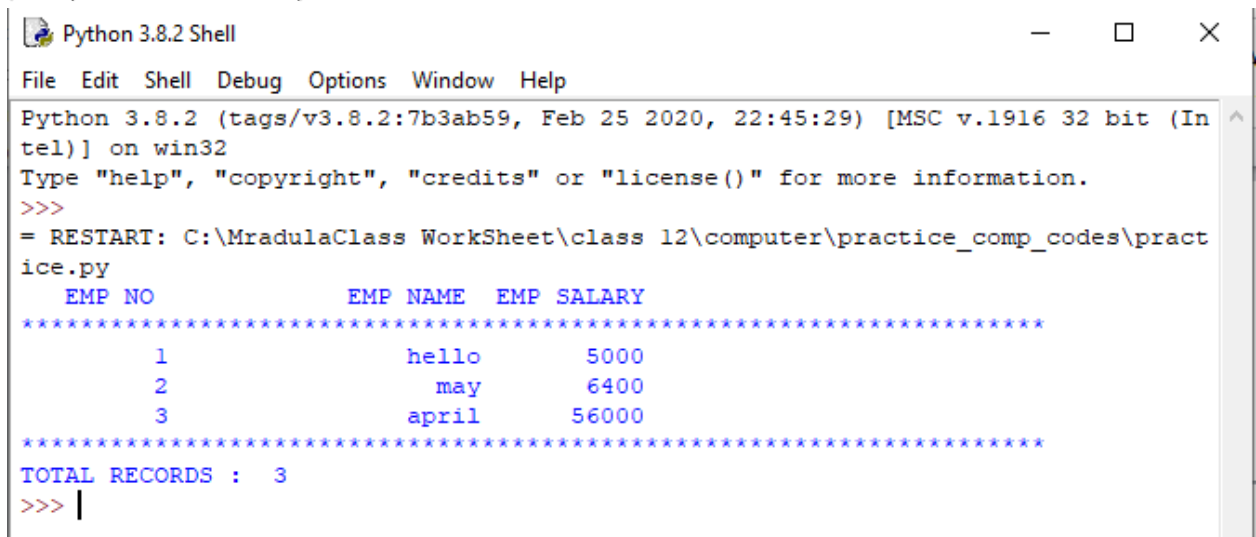


```
Python 3.8.2 Shell
File Edit Shell Debug Options Window Help
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\MradulaClass WorkSheet\class 12\computer\practice_comp_codes\practice.py
*****EMPLOYEE SEARCH FORM*****
enter employee number to search2
EMP NO EMP NAME EMP SALARY

 2 may 6400
>>> |
```

17. finding number of record in binary file

```
import pickle
emp=[]
f=open("employee.dat","rb")
emp=pickle.load(f)
l=len(emp)
while true:
 try:
 emp=pickle.load(f)
 except eoferror:
 break
print("%10s"% "emp no ", "%20s"% "emp name ", "%10s"% "emp salary")
print("*****")
for e in emp:
 print("%10s"%e[0], "%20s"%e[1], "%10s"%e[2])
print("*****")
print("total records : ",l)
```



Python 3.8.2 Shell

File Edit Shell Debug Options Window Help

Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32  
Type "help", "copyright", "credits" or "license()" for more information.  
>>>  
= RESTART: C:\MradulaClass WorkSheet\class 12\computer\practice\_comp\_codes\practice.py  
EMP NO EMP NAME EMP SALARY  
\*\*\*\*\*  
1 hello 5000  
2 may 6400  
3 april 56000  
\*\*\*\*\*  
TOTAL RECORDS : 3  
>>> |

## 18. updating employee record

```
import pickle
emp=[]
f=open("employee.dat","rb")
emp=pickle.load(f)
print("##employee records##")
print(emp)
print("-----")
f.close()
f=open("employee.dat","wb")
found=false
en=int(input("enter employee number to update :"))
for i in range(len(emp)):
 if emp[i][0]==en:
 sl=int(input("enter new salary:"))
 emp[i][2]=sl
 found=true
 print("##record updated##")
if not found:
 print("## no such employee number ##")
pickle.dump(emp,f)
f.close()
f=open("employee.dat","rb")
emp=pickle.load(f)
print("## employee records after update ##")
print(emp)
print("-----")
```

Python 3.8.2 Shell

File Edit Shell Debug Options Window Help

Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32

Type "help", "copyright", "credits" or "license()" for more information.

>>>

= RESTART: C:\MradulaClass WorkSheet\class 12\computer\practice\_comp\_codes\practice.py

##EMPLOYEE RECORDS##

[[1, 'hello', 5000], [2, 'may', 6400], [3, 'april', 56000]]

-----

enter employee number to update :3

enter new salary:5400

##RECORD UPDATED##

## EMPLOYEE RECORDS AFTER UPDATE ##

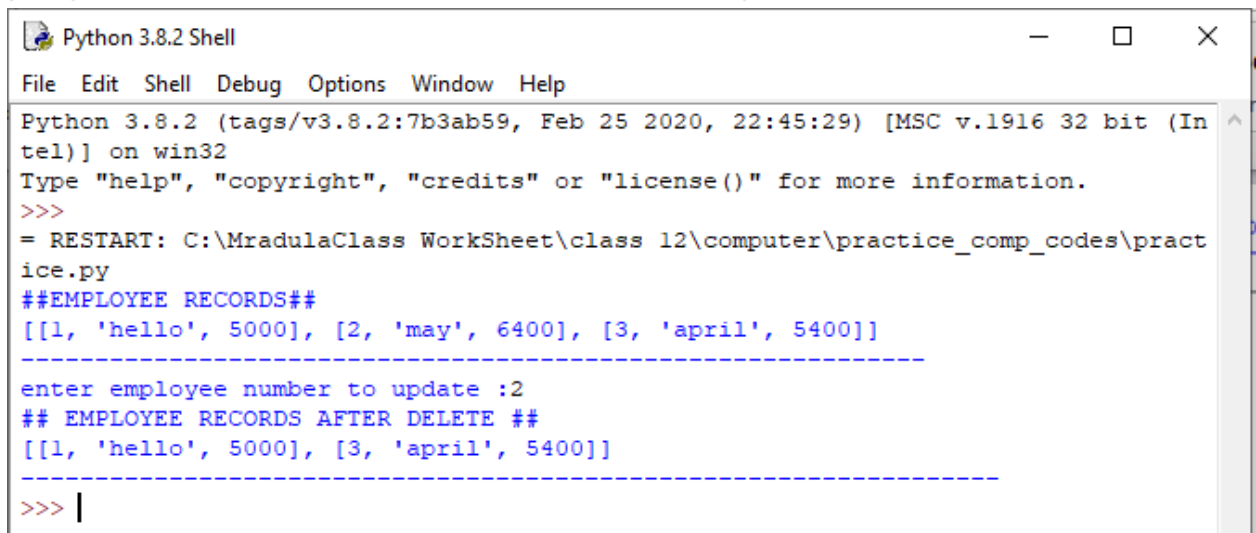
[[1, 'hello', 5000], [2, 'may', 6400], [3, 'april', 5400]]

-----

...

#### 19. deleting employee record

```
import pickle
emp=[]
f=open("employee.dat","rb")
emp=pickle.load(f)
print("##employee records##")
print(emp)
print("-----")
f.close()
f=open("employee.dat","wb")
found=false
en=int(input("enter employee number to update :"))
emp2=[]
for i in range(len(emp)):
 if emp[i][0]!=en:
 emp2.append(emp[i])
pickle.dump(emp2,f)
f.close()
f=open("employee.dat","rb")
emp=pickle.load(f)
print("## employee records after delete ##")
print(emp)
print("-----")
```



```
Python 3.8.2 Shell
File Edit Shell Debug Options Window Help
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\MradulaClass WorkSheet\class 12\computer\practice_comp_codes\practice.py
##EMPLOYEE RECORDS##
[[1, 'hello', 5000], [2, 'may', 6400], [3, 'april', 5400]]

enter employee number to update :2
EMPLOYEE RECORDS AFTER DELETE
[[1, 'hello', 5000], [3, 'april', 5400]]

>>> |
```

20. following is the structure of each record in a data file named "product.dat". {"prod\_code":value, "prod\_desc":value, "stock":value}  
the values for prod\_code and prod\_desc are strings, and the value for stock is an integer. write a function in python to update the file with a new value of stock. the stock and the product\_code, whose stock is to be updated, are to be input during the execution of the function.

```
import pickle
def update():
 cd=input("enter product code whose stock is to be updated")
 f=open("product.dat",'rb')
 flag=false
 rec_list=[]
 while true:
 try:
 rec=pickle.load(f)
 rec_list.append(rec)
 except eoferror:
 f.close()
 break
 for i in range(len(rec_list)):
 if rec_list[i]['code']==cd:
 flag=true
 rec_list[i]['stock']=int(input("enter new value of stock"))
 break
 if flag==false:
 print("product code not found")
 else:
 f=open("product.dat",'wb')
 for rec in rec_list:
 pickle.dump(rec,f)
 print("record updated")
 f.close()
```

```
def enter_data():
 d={}
 f=open("product.dat",'ab')
 pc=input("enter product code")
 pd=input("enter product description")
 st=int(input("enter stock"))
 d['code']=pc
 d['desc']=pd
 d['stock']=st
 pickle.dump(d,f)
```

```
f.close()
print("record inserted")

def disp_all():
 f=open("product.dat",'rb')
 while true:
 try:
 rec=pickle.load(f)
 print(rec)
 except eoferror:
 break
 f.close()
#driver code
while true:
 print("1: enter record")
 print("2: display all")
 print("3: update record")
 print("0: exit")
 ch=int(input("enter choice"))
 if ch==1:
 enter_data()
 elif ch==2:
 disp_all()
 elif ch==3:
 update()
 elif ch==0:
 break
```

```
= RESTART: C:\MradulaClass WorkSheet\class 12\computer\practice_comp_codes\practice.py
1: Enter record
2: Display all
3: Update record
0: Exit
Enter choice1
enter product codec01
enter product descriptionshampoo
enter stock25
record inserted
1: Enter record
2: Display all
3: Update record
0: Exit
Enter choice2
{'code': 'a01', 'desc': 'chocolate', 'stock': 20}
{'code': 'a02', 'desc': 'strawberry', 'stock': 90}
{'code': 'b01', 'desc': 'lip gloss', 'stock': 25}
{'code': 'b02', 'desc': 'earrings', 'stock': 30}
{'code': 'c01', 'desc': 'shampoo', 'stock': 25}
1: Enter record
2: Display all
3: Update record
0: Exit
Enter choice3
enter product code whose stock is to be updatedc01
enter new value of stock35
record updated
1: Enter record
2: Display all
3: Update record
0: Exit
Enter choice2
{'code': 'a01', 'desc': 'chocolate', 'stock': 20}
{'code': 'a02', 'desc': 'strawberry', 'stock': 90}
{'code': 'b01', 'desc': 'lip gloss', 'stock': 25}
{'code': 'b02', 'desc': 'earrings', 'stock': 30}
{'code': 'c01', 'desc': 'shampoo', 'stock': 35}
1: Enter record
2: Display all
3: Update record
-
1: Enter record
2: Display all
3: Update record
0: Exit
Enter choice2
{'code': 'a01', 'desc': 'chocolate', 'stock': 20}
{'code': 'a02', 'desc': 'strawberry', 'stock': 90}
{'code': 'b01', 'desc': 'lip gloss', 'stock': 25}
{'code': 'b02', 'desc': 'earrings', 'stock': 30}
{'code': 'c01', 'desc': 'shampoo', 'stock': 35}
1: Enter record
2: Display all
3: Update record
0: Exit
Enter choice0
>>>
```

---

21. given a binary file "stuq2.dat", containing records of the following type: {'s\_admno':value, 's\_name':value, 'percentage':value} where these three values are: s\_admno – admission number of student (string) s\_name – name of student (string) percentage – marks percentage of student (float) write a function in python that would read contents of the file "student.dat" and display the details of those students whose percentage is equal to or above 75.

```
import pickle
def distinction():
 f=open("stuq2.dat",'rb')
 flag=false
 rec={}
 while true:
 try:
 rec=pickle.load(f)
 if rec['perc']>=75:
 print(rec)
 except eoferror:
 f.close()
 break
def enter_data():
 d={}
 f=open("stuq2.dat",'ab')
 ad=input("enter admn no")
 nm=input("enter stu name")
 pr=int(input("enter perc"))
 d['adm']=ad
 d['name']=nm
 d['perc']=pr
 pickle.dump(d,f)
 f.close()
 print("record inserted")
def disp_all():
 f=open("stuq2.dat",'rb')
 while true:
 try:
 rec=pickle.load(f)
 print(rec)
 except eoferror:
 break
 f.close()
#driver code
while true:
 print("1: enter record")
```



```

print("2: display all")
print("3: display distinction holders")
print("0: exit")
ch=int(input("enter choice"))
if ch==1:
 enter_data()
elif ch==2:
 disp_all()
elif ch==3:
 distinction()
elif ch==0:
 break

```

```

Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\MradulaClass WorkSheet\class 12\computer\practice_comp_codes\practice.py
1: Enter record
2: Display all
3: display distinction holders
0: Exit
Enter choice1
enter admn no11234
enter stu namepriya
enter perc94
record inserted
1: Enter record
2: Display all
3: display distinction holders
0: Exit
Enter choice2
{'adm': '11189', 'name': 'mradula', 'perc': 86}
{'adm': '43855', 'name': 'hello', 'perc': 54}
{'adm': '9985', 'name': 'chandru', 'perc': 87}
{'adm': '09567', 'name': 'april', 'perc': 94}
{'adm': '11234', 'name': 'priya', 'perc': 94}
1: Enter record
2: Display all
3: display distinction holders
0: Exit
Enter choice3
{'adm': '11189', 'name': 'mradula', 'perc': 86}
{'adm': '9985', 'name': 'chandru', 'perc': 87}
{'adm': '09567', 'name': 'april', 'perc': 94}
{'adm': '11234', 'name': 'priya', 'perc': 94}
1: Enter record
2: Display all
3: display distinction holders
0: Exit
Enter choice0
>>> |

```

22. assuming the tuple vehicle as follows: ( vehicletype, no\_of\_wheels) where vehicletype is a string and no\_of\_wheels is an integer. write a function showfile() to read all the records present in an already existing binary file speed\_tuple.dat and display them on the screen, also count the number of records present in the file.

```
import pickle
def enter_data():
 t=()
 f=open("speed_tuple.dat",'ab')
 vt=input("enter vehicle type")
 now=int(input("enter no. of wheels"))
 t=(vt,now)
 pickle.dump(t,f)
 f.close()
 print("record inserted")
def showfile():
 f=open("speed_tuple.dat",'rb')
 ctr=0
 while True:
 try:
 rec=pickle.load(f)
 print(rec)
 ctr+=1
 except EOFError:
 break
 f.close()
 print("no. of records read:",ctr)
#driver code
while True:
 print("1: enter record")
 print("2: display all")
 print("0: exit")
 ch=int(input("enter choice"))
 if ch==1:
 enter_data()
 elif ch==2:
 showfile()
 elif ch==0:
 break
```

```
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\MradulaClass WorkSheet\class 12\computer\practice_comp_codes\practice.py
1: Enter record
2: Display all
0: Exit
Enter choice1
enter vehicle typeauto rikshaw
enter no. of wheels3
record inserted
1: Enter record
2: Display all
0: Exit
Enter choice2
('car', 4)
('truck', 8)
('bike', 2)
('auto rikshaw', 3)
No. of records read: 4
1: Enter record
2: Display all
0: Exit
Enter choice0
>>> |
```

23. write a function in python to search for a bookno from a binary file "book\_dict.dat", assuming the binary file is containing the records of the following type: {"bookno":value, "book\_name":value} assume that bookno is an integer.

```
import pickle
def search():
 bn=input("enter book number whose details are to be searched")
 f=open("book_dict.dat",'rb')
 flag=false
 rec_list=[]
 while true:
 try:
 rec=pickle.load(f)
 rec_list.append(rec)
 except eoferror:
 f.close()
 break
 for i in range(len(rec_list)):
 if rec_list[i]["bookno"]==bn:
 flag=true
 print(rec_list[i])
 break
 f.close()
 if flag==false:
 print("book number not found")
def enter_data():
 d={}
 f=open("book_dict.dat",'ab')
 bno=input("enter book number")
 bn=input("enter book name")
 d["bookno"]=bno
 d["book_name"]=bn
 pickle.dump(d,f)
 f.close()
 print("record inserted")
def disp_all():
 f=open("book_dict.dat",'rb')
 while true:
 try:
 rec=pickle.load(f)
 print(rec)
 except eoferror:
 break
```

```

f.close()
while true:
 print("1: enter record")
 print("2: display all")
 print("3: search book no")
 print("0: exit")
 ch=int(input("enter choice"))
 if ch==1:
 enter_data()
 elif ch==2:
 disp_all()
 elif ch==3:
 search()
 elif ch==0:
 break

```

```


Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\MradulaClass WorkSheet\class 12\computer\practice_comp_codes\practice.py
1: Enter record
2: Display all
3: Search book no
0: Exit
Enter choice1
enter book number5
enter book namewings of fire
record inserted
1: Enter record
2: Display all
3: Search book no
0: Exit
Enter choice2
{'BookNo': '1', 'Book_name': 'harry potter'}
{'BookNo': '2', 'Book_name': 'nancy drew'}
{'BookNo': '3', 'Book_name': '13 reasons why'}
{'BookNo': '4', 'Book_name': 'the fault in our stars'}
{'BookNo': '5', 'Book_name': 'wings of fire'}
1: Enter record
2: Display all
3: Search book no
0: Exit
Enter choice3
enter book number whose details are to be searched3
{'BookNo': '3', 'Book_name': '13 reasons why'}
1: Enter record
2: Display all
3: Search book no
0: Exit
Enter choice0
>>> |

```

24. assuming that a binary file vintage.dat contains records of the following type, write a function in python to read the data vintage.dat and display those vintage vehicles, which are priced between 200000 and 250000. [vno, vdesc, price]

```
import pickle
def disp_price_range():
 f=open("vintage.dat",'rb')
 flag=false
 rec=[]
 while true:
 try:
 rec=pickle.load(f)
 if (200000<=rec[2]<=250000):
 print(rec)
 except eoferror:
 f.close()
 break
def enter_data():
 l=[]
 f=open("vintage.dat",'ab')
 vn=int(input("enter vehicle no"))
 vd=input("enter vehicle description")
 pr=eval(input("enter price"))
 l=[vn,vd,pr]
 pickle.dump(l,f)
 f.close()
 print("record inserted")
def disp_all():
 f=open("vintage.dat",'rb')
 while true:
 try:
 rec=pickle.load(f)
 print(rec[0],rec[1],rec[2])
 except eoferror:
 break
 f.close()
#driver code
while true:
 print("1: enter record")
 print("2: display all")
 print("3: display according to range")
 print("0: exit")
 ch=int(input("enter choice"))
```

```
if ch==1:
 enter_data()
elif ch==2:
 disp_all()
elif ch==3:
 disp_price_range()
elif ch==0:
 break
```



```
Python 3.8.2 Shell
File Edit Shell Debug Options Window Help
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\MradulaClass WorkSheet\class 12\computer\practice_comp_codes\practice.py
1: Enter record
2: Display all
3: display according to range
0: Exit
Enter choice1
enter vehicle no3803
enter vehicle descriptiontruck
enter price470000
record inserted
1: Enter record
2: Display all
3: display according to range
0: Exit
Enter choice2
1234 rfde 4356
3232 car 54000000
4354 scooty 54000
3803 truck 470000
1: Enter record
2: Display all
3: display according to range
0: Exit
Enter choice3
1: Enter record
2: Display all
3: display according to range
0: Exit
Enter choice0
>>>
```

25. write a function in python to search for a laptop from a binary file "laptop.dat" containing the records of following type. the user should enter the model number and the function should display the details of the laptop. [modelno, ram, hdd, details] where modelno, ram, hdd are integers, and details is a string.

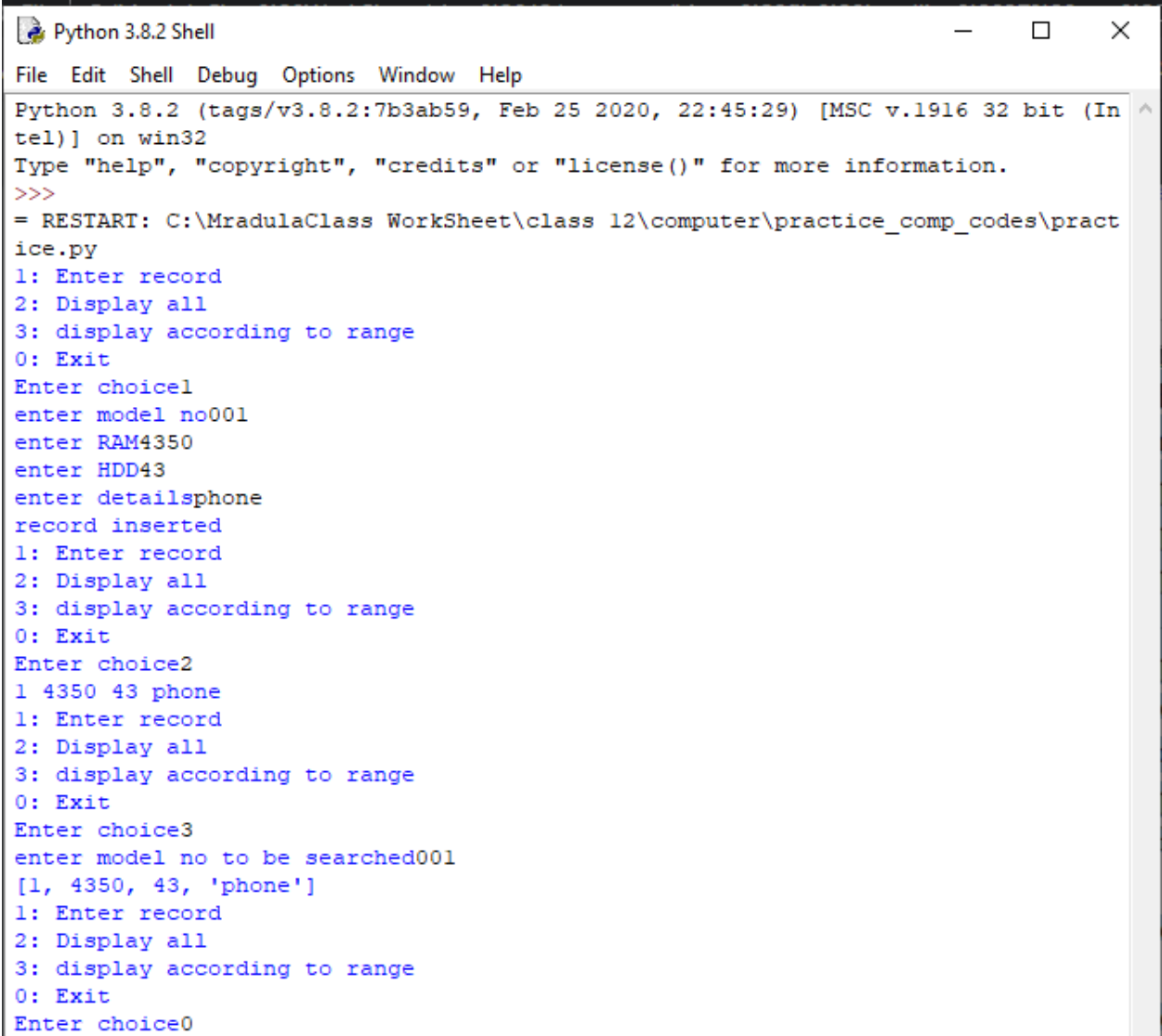
```
import pickle
def disp_model_details():
 f=open("laptop.dat",'rb')
 flag=false
 rec=[]
 m=int(input("enter model no to be searched"))
 while true:
 try:
 rec=pickle.load(f)
 if rec[0]==m:
 print(rec)
 flag=true
 except eoferror:
 f.close()
 break
 if flag==false:
 print("invalid model number")
def enter_data():
 l=[]
 f=open("laptop.dat",'ab')
 mn=int(input("enter model no"))
 rm=int(input("enter ram"))
 hd=int(input("enter hdd"))
 det=input("enter details")
 l=[mn,rm,hd,det]
 pickle.dump(l,f)
 f.close()
 print("record inserted")
def disp_all():
 f=open("laptop.dat",'rb')
 while true:
 try:
 rec=pickle.load(f)
 print(rec[0],rec[1],rec[2],rec[3])
 except eoferror:
 break
 f.close()
while true:
```



```

print("1: enter record")
print("2: display all")
print("3: display according to range")
print("0: exit")
ch=int(input("enter choice"))
if ch==1:
 enter_data()
elif ch==2:
 disp_all()
elif ch==3:
 disp_model_details()
elif ch==0:
 break

```



```

Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\MradulaClass WorkSheet\class 12\computer\practice_comp_codes\practice.py
1: Enter record
2: Display all
3: display according to range
0: Exit
Enter choice1
enter model no001
enter RAM4350
enter HDD43
enter detailsphone
record inserted
1: Enter record
2: Display all
3: display according to range
0: Exit
Enter choice2
1 4350 43 phone
1: Enter record
2: Display all
3: display according to range
0: Exit
Enter choice3
enter model no to be searched001
[1, 4350, 43, 'phone']
1: Enter record
2: Display all
3: display according to range
0: Exit
Enter choice0

```

26. write a function in python to search for the details (number and calls) of those mobile phones which have more than 1000 calls from a binary file "mobile.dat". assuming that this binary file contains records of the following type: (number,calls)

```
import pickle
def search_file():
 f=open("mobile.dat",'rb')
 while true:
 try:
 rec=pickle.load(f)
 if rec[1]>1000:
 print(rec)
 except eoferror:
 break
 f.close()
def enter_data():
 t=()
 f=open("mobile.dat",'ab')
 mn=input("enter mobile number")
 noc=int(input("enter no. of calls made"))
 t=(mn,noc)
 pickle.dump(t,f)
 f.close()
 print("record inserted")
def showfile():
 f=open("mobile.dat",'rb')
 ctr=0
 while true:
 try:
 rec=pickle.load(f)
 print(rec)
 except eoferror:
 break
 f.close()
while true:
 print("1: enter record")
 print("2: display all")
 print("3: display deatils for calls more than 1000")
 print("0: exit")
 ch=int(input("enter choice"))
 if ch==1:
 enter_data()
 elif ch==2:
```

```

 showfile()
elif ch==3:
 search_file()
elif ch==0:
 break

```

```
Python 3.8.2 Shell
File Edit Shell Debug Options Window Help
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\MradulaClass WorkSheet\class 12\computer\practice_comp_codes\practice.py
1: Enter record
2: Display all
3: display deatils for calls more than 1000
0: Exit
Enter choice1
enter mobile number8076988546
enter no. of calls made2134
record inserted
1: Enter record
2: Display all
3: display deatils for calls more than 1000
0: Exit
Enter choice2
('9854545453', 123)
('9897456567', 1234)
('9871473334', 654)
('8076988546', 2134)
1: Enter record
2: Display all
3: display deatils for calls more than 1000
0: Exit
Enter choice3
('9897456567', 1234)
('8076988546', 2134)
1: Enter record
2: Display all
3: display deatils for calls more than 1000
0: Exit
Enter choice0
```

27. write a function in python to read the records from binary file games.dat and display the details of those games, which are meant for children of age range "8 to 13". assume that the file games.dat contains records of the following type: [gamecode, gamename, agerange]

```
import pickle
def show_age_range():
 f=open("games.dat",'rb')
 flag=false
 rec=[]
 while true:
 try:
 rec=pickle.load(f)
 if rec[2]=="8 to 13":
 print(rec)
 except eoferror:
 f.close()
 break
def enter_data():
 l=[]
 f=open("games.dat",'ab')
 gc=input("enter game code")
 gn=input("enter game name")
 ar=input("enter age range")
 l=[gc,gn,ar]
 pickle.dump(l,f)
 f.close()
 print("record inserted")
def disp_all():
 f=open("games.dat",'rb')
 while true:
 try:
 rec=pickle.load(f)
 print(rec)
 except eoferror:
 break
 f.close()
while true:
 print("1: enter record")
 print("2: display all")
 print("3: display games of age range 8 to 13")
 print("0: exit")
 ch=int(input("enter choice"))
 if ch==1:
```

```
 enter_data()
elif ch==2:
 disp_all()
elif ch==3:
 show_age_range()
elif ch==0:
 break
```

Python 3.8.2 Shell

File Edit Shell Debug Options Window Help

Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32

Type "help", "copyright", "credits" or "license()" for more information.

>>>

= RESTART: C:\MradulaClass WorkSheet\class 12\computer\practice\_comp\_codes\practice.py

1: Enter record

2: Display all

3: display games of age range 8 to 13

0: Exit

Enter choice1

enter game code33433

enter game namechess

enter age rangel0 +

record inserted

1: Enter record

2: Display all

3: display games of age range 8 to 13

0: Exit

Enter choice2

['1111', 'hello', '8 - 15']

['1123', 'plot 4', '5+']

['master mind', 'hacker', '7+']

['33433', 'chess', '10 +']

1: Enter record

2: Display all

3: display games of age range 8 to 13

0: Exit

Enter choice3

1: Enter record

2: Display all

3: display games of age range 8 to 13

0: Exit

Enter choice0

>>> |

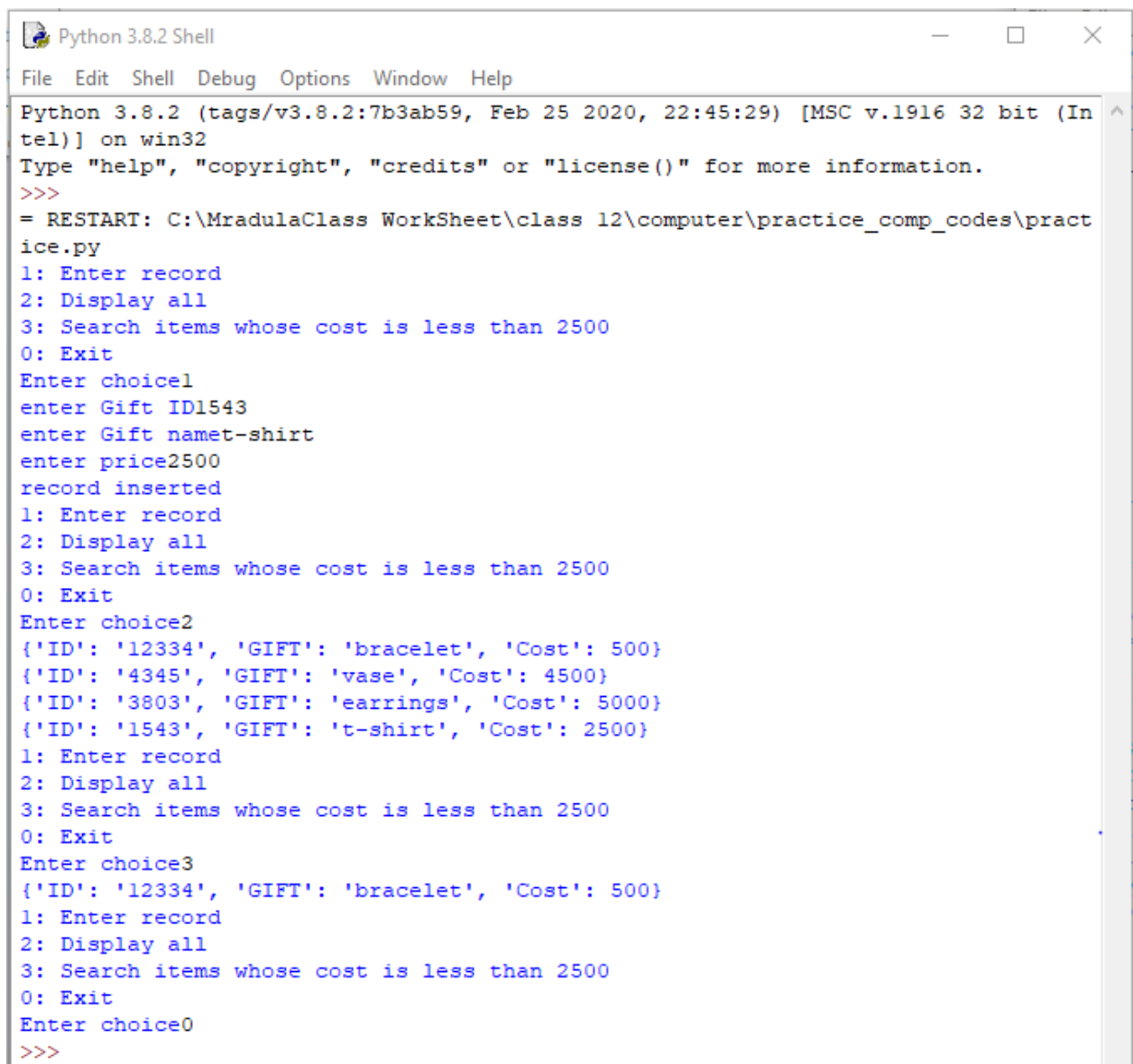
28. write a function in python to read each record of a binary file items.dat, find and display those items which costs less than 2500. assume that the file items.dat is created with the help of objects of the following type: {"id":string, "gift":string, "cost":integer}

```
import pickle
def search():
 f=open("items_q9.dat",'rb')
 flag=false
 rec_list=[]
 while true:
 try:
 rec=pickle.load(f)
 rec_list.append(rec)
 except eoferror:
 f.close()
 break
 for i in range(len(rec_list)):
 if rec_list[i]['cost']<2500:
 flag=true
 print(rec_list[i])
 break
 if flag==false:
 print("not found")
def enter_data():
 d={}
 f=open("items_q9.dat",'ab')
 gid=input("enter gift id")
 gn=input("enter gift name")
 pr=int(input("enter price"))
 d['id']=gid
 d['gift']=gn
 d['cost']=pr
 pickle.dump(d,f)
 f.close()
 print("record inserted")
def disp_all():
 f=open("items_q9.dat",'rb')
 while true:
 try:
 rec=pickle.load(f)
 print(rec)
 except eoferror:
 break
```

```

f.close()
while true:
 print("1: enter record")
 print("2: display all")
 print("3: search items whose cost is less than 2500")
 print("0: exit")
 ch=int(input("enter choice"))
 if ch==1:
 enter_data()
 elif ch==2:
 disp_all()
 elif ch==3:
 search()
 elif ch==0:
 break

```



```

Python 3.8.2 Shell
File Edit Shell Debug Options Window Help
Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 22:45:29) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\MradulaClass WorkSheet\class 12\computer\practice_comp_codes\practice.py
1: Enter record
2: Display all
3: Search items whose cost is less than 2500
0: Exit
Enter choice1
enter Gift ID1543
enter Gift name t-shirt
enter price2500
record inserted
1: Enter record
2: Display all
3: Search items whose cost is less than 2500
0: Exit
Enter choice2
{'ID': '12334', 'GIFT': 'bracelet', 'Cost': 500}
{'ID': '4345', 'GIFT': 'vase', 'Cost': 4500}
{'ID': '3803', 'GIFT': 'earrings', 'Cost': 5000}
{'ID': '1543', 'GIFT': 't-shirt', 'Cost': 2500}
1: Enter record
2: Display all
3: Search items whose cost is less than 2500
0: Exit
Enter choice3
{'ID': '12334', 'GIFT': 'bracelet', 'Cost': 500}
1: Enter record
2: Display all
3: Search items whose cost is less than 2500
0: Exit
Enter choice0
>>>

```

29. write a definition for function bumper() in python to read each object of a binary file items.dat, find and copy details of those items, whose cost is less than 2000 to the file "discount.dat". assume that the file items.dat is created with the help of objects of the following type: {"id":string, "gift":string, "cost":integer} display the contents of the file "items.dat" and "discount.dat"

```
import pickle
def bumper():
 fr=open("items_q9.dat",'rb')
 fw=open("discount.dat",'wb')
 flag=false
 rec_list=[]
 while true:
 try:
 rec=pickle.load(fr)
 rec_list.append(rec)
 except eoferror:
 fr.close()
 break
 for i in range(len(rec_list)):
 if rec_list[i]['cost']<2000:
 flag=true
 pickle.dump(rec_list[i],fw)
 fr.close()
 if flag==false:
 print("not found")
 fw.close()
 disp_all("items_q9.dat")
 disp_all("discount.dat")
def search():
 f=open("items_q9.dat",'rb')
 flag=false
 rec_list=[]
 while true:
 try:
 rec=pickle.load(f)
 rec_list.append(rec)
 except eoferror:
 f.close()
 break
 for i in range(len(rec_list)):
 if rec_list[i]['cost']<2500:
 flag=true
 print(rec_list[i])
```



```

 break
 f.close()
 if flag==false:
 print("not found")
def enter_data():
 d={}
 f=open("items_q9.dat",'ab')
 gid=input("enter gift id")
 gn=input("enter gift name")
 pr=int(input("enter price"))
 d['id']=gid
 d['gift']=gn
 d['cost']=pr
 pickle.dump(d,f)
 f.close()
 print("record inserted")
def disp_all(fname):
 f=open(fname,'rb')
 print("details of file",f.name)
 while true:
 try:
 rec=pickle.load(f)
 print(rec)
 except eoferror:
 break
 f.close()
while true:
 print("1: enter record")
 print("2: display all")
 print("3: search items whose cost is less than 2500")
 print("4: copy items whose cost is less than 2000")
 print("0: exit")
 ch=int(input("\nenter choice"))
 if ch==1:
 enter_data()
 elif ch==2:
 fname=input("name of file whose contents are to shown")
 disp_all(fname)
 elif ch==3:
 search()
 elif ch==4:
 bumper()

```

```

elif ch==0:
 break
= RESTART: C:\MradulaClass WorkSheet\class 12\computer\practice_comp_codes\practice.py
1: Enter record
2: Display all
3: Search items whose cost is less than 2500
4: Copy items whose cost is less than 2000
0: Exit
Enter choice1
enter Gift ID7890
enter Gift namestick note
enter price600
record inserted
1: Enter record
2: Display all
3: Search items whose cost is less than 2500
4: Copy items whose cost is less than 2000
0: Exit
Enter choice2
name of file whose contents are to showndiscount.dat
details of file discount.dat
{'ID': '12334', 'GIFT': 'bracelet', 'Cost': 500}
{'ID': '56785', 'GIFT': 'key chain', 'Cost': 550}
1: Enter record
2: Display all
3: Search items whose cost is less than 2500
4: Copy items whose cost is less than 2000
0: Exit
Enter choice3
{'ID': '12334', 'GIFT': 'bracelet', 'Cost': 500}
1: Enter record
2: Display all
3: Search items whose cost is less than 2500
4: Copy items whose cost is less than 2000
0: Exit
1: Enter record
2: Display all
3: Search items whose cost is less than 2500
4: Copy items whose cost is less than 2000
0: Exit
Enter choice4
details of file items_q9.dat
{'ID': '12334', 'GIFT': 'bracelet', 'Cost': 500}
{'ID': '4345', 'GIFT': 'vase', 'Cost': 4500}
{'ID': '3803', 'GIFT': 'earrings', 'Cost': 5000}
{'ID': '1543', 'GIFT': 't-shirt', 'Cost': 2500}
{'ID': '1234567', 'GIFT': 'necklace', 'Cost': 10000}
{'ID': '56785', 'GIFT': 'key chain', 'Cost': 550}
{'ID': '7890', 'GIFT': 'stick note', 'Cost': 600}
details of file discount.dat
{'ID': '12334', 'GIFT': 'bracelet', 'Cost': 500}
{'ID': '56785', 'GIFT': 'key chain', 'Cost': 550}
{'ID': '7890', 'GIFT': 'stick note', 'Cost': 600}
1: Enter record
2: Display all
3: Search items whose cost is less than 2500
4: Copy items whose cost is less than 2000
0: Exit
Enter choice0
>>> |

```