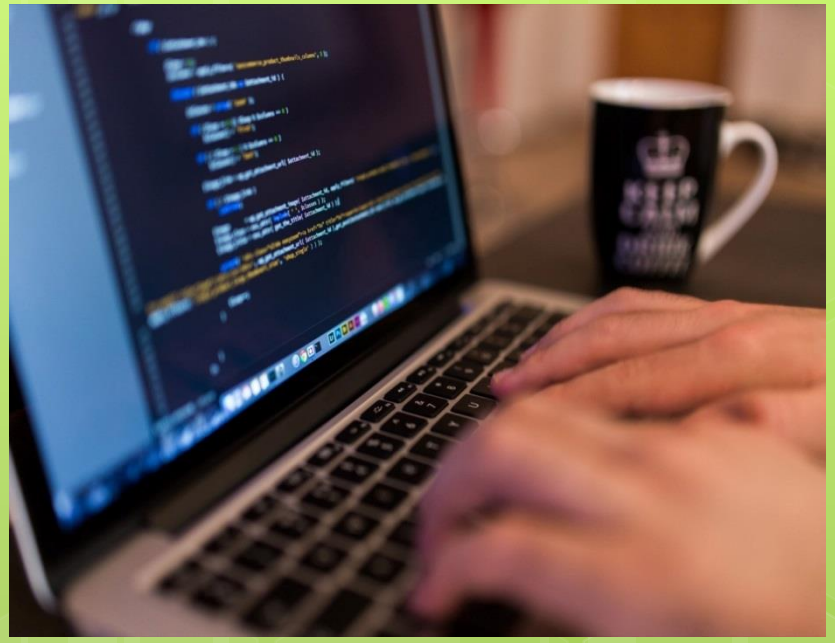


FILE HANDLING IN PYTHON

CLASS-XII

Computer Science

ARUN KUMAR, PGT CS,
KV-1, FARIDABAD

Binary File Operations

The open() function opens a file in text format by default. To open a file in binary format, add 'b' to the mode parameter. Hence the "rb" mode opens the file in binary format for reading, while the "wb" mode opens the file in binary format for writing. Unlike text mode files, binary files are not human readable. When opened using any text editor, the data is unrecognizable.

Reading and Writing to a Binary File

```
fo=open("binary.txt","wb")
fr=open("binary.txt","rb")
num=[5, 10, 15, 20, 25]
arr=bytearray(num)
fo.write(arr)
print("file created and data
fo.close()
num=(fr.read())
print(num)
fr.close()
```

```
Python 3.7.4 (tags/v3.7.4:e
(Intel)] on win32
Type "help", "copyright",
>>>
= RESTART: C:\Users\user\Ap
file created and data writ
b'\x05\n\x0f\x14\x19'
>>>
```

Reading and Writing to a Binary File

```
fo=open("binary.txt","wb")
fr=open("binary.txt","rb")
str="stay home, stay safe"
arr=str.encode("utf-8")
fo.write(arr)
print("file created")
fo.close()
num=(fr.read())
print(num)
fr.close()
```

```
Python 3.7.4 (tags/v3.7.4:e09359
(Intel)] on win32
Type "help", "copyright", "credi
>>>
= RESTART: C:\Users\user\AppData
file created
b'stay home, stay safe'
>>>
```

Pickle Module

Python pickle module is used for serializing and de-serializing python object structures. The process to converts any kind of python objects (list, dict, etc.) into byte streams (0s and 1s) is called pickling or serialization or flattening or marshalling. We can converts the byte stream (generated through pickling) back into python objects by a process called as unpickling.

Only after importing pickle module we can do pickling and unpickling. Importing pickle can be done using the following command –

Import pickle

dump() and load() functions

pickle.dump() function is used to store the object data to the file. It takes 3 arguments. First argument is the object that we want to store. The second argument is the file object we get by opening the desired file in write-binary(wb) mode. And the third argument is the key-value argument. This argument defines the protocol. There are two type of protocol– pickle.HIGHEST_PROTOCOL and pickle.DEFAULT_PROTOCOL.

Pickle.load() function is used to retrieve pickled data. The steps are quite simple. We have to use pickle.load() function to do that. The primary argument of pickle load function is the file object that you get by opening the file in read-binary(rb) mode.

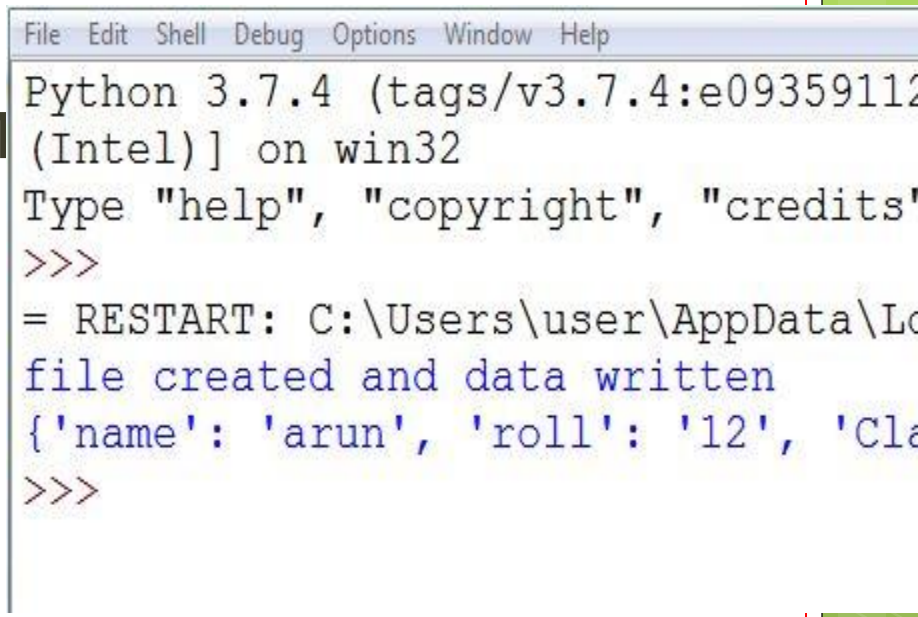
dump() and load() functions

```
#dumping a list
import pickle
fo=open("binary.dat","wb")
fr=open("binary.dat","rb")
num=['a','b','c','d','e']
pickle.dump(num,fo)
print("file created and data written")
fo.close()
mum=pickle.load(fr)
print(num)
fr.close()
```

```
Python 3.7.4 (tags/v3.7.4:e0
(Intel)] on win32
Type "help", "copyright", "c
>>>
= RESTART: C:\Users\user\App
file created and data writte
['a', 'b', 'c', 'd', 'e']
>>>
```

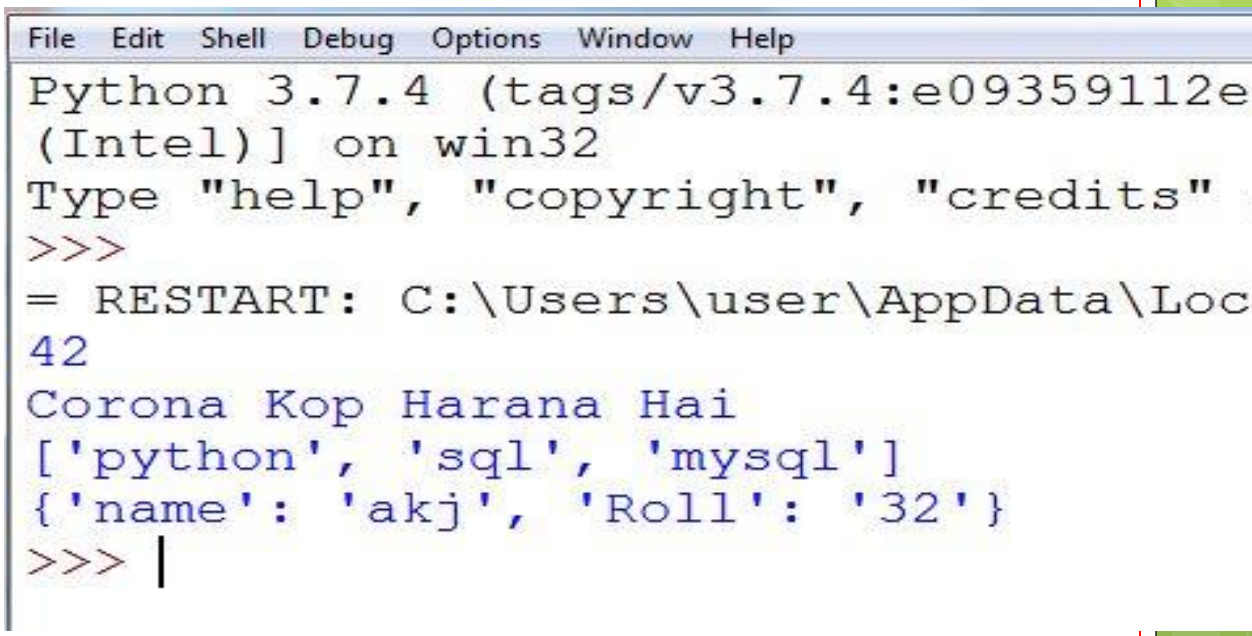
dump() and load() functions

```
#dumping a dictionary
import pickle
fo=open("binary1.dat","wb")
fr=open("binary1.dat","rb")
num={"name":"arun","roll":"1
pickle.dump(num,fo)
print("file created and data
fo.close()
mum=pickle.load(fr)
print(num)
fr.close()
```



```
#Loading using Iteration
import pickle
fo=open("binary2.dat", "wb")
fr=open("binary2.dat", "rb")
num= 42
str= "Corona Kop Harana Hai"
mlist= ["python", "sql", "mysql"]
mdict= { "name": "akj", "Roll": "32" }
pickle.dump(num,fo)
pickle.dump(str, fo)
pickle.dump(mlist, fo)
pickle.dump(mdict, fo)
fo.close()
while True:
    try:
        r=pickle.load(fr)
        print(r)
    except EOFError:
        break

fr.close()
```



Append data to binary file

```
import pickle
rollno= int(input('Enter roll number:'))
name = input('Enter Name:')
marks = int(input('Enter Marks'))
rec = {'Rollno':rollno,'Name':name,'Marks':marks}
#Creating the dictionary
f = open("student.dat",'ab')
pickle.dump(rec,f)
print(" file created and data inserted")
f.close()
```

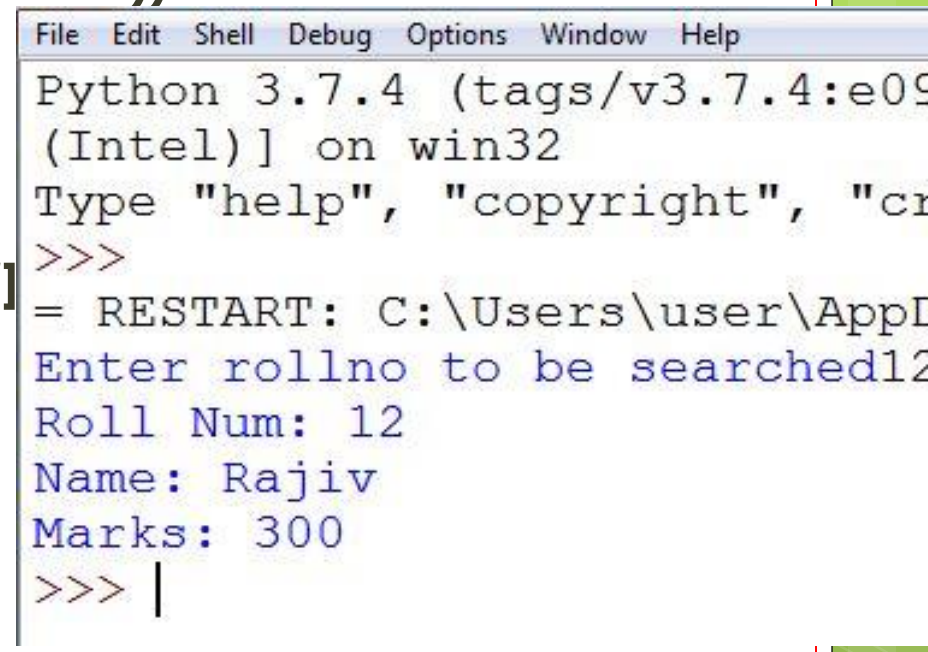
Append data to binary file

```
fr = open("student.dat",'rb')
while True:
    try:
        rec = pickle.load(fr)
        print('Roll Num:',rec['Rollno'])
        print('Name:',rec['Name'])
        print('Marks:',rec['Marks'])
    except EOFError:
        break
fr.close()
```

```
File  Edit  Shell  Debug  Options  Window  Help
Python 3.7.4 (tags/v3.7.4:e
(Intel)] on win32
Type "help", "copyright",
>>>
= RESTART: C:\Users\user\Ap
Enter roll number:12
Enter Name:Rajiv
Enter Marks300
 file created and data inse
Roll Num: 56
Name: pritam
Marks: 500
Roll Num: 67
Name: Neeraj
Marks: 567
Roll Num: 12
Name: Rajiv
Marks: 300
>>>
```

Searching data in binary file

```
import pickle
fr = open('student.dat','rb')
flag = False
r=int(input("Enter rollno to be searched"))
while True:
    try:
        rec = pickle.load(fr)
        if rec['Rollno'] == r:
            print('Roll Num:',rec['Rollno'])
            print('Name:',rec['Name'])
            print('Marks:',rec['Marks'])
        flag = True
    except EOFError:
        break
if flag == False:
    print('No Records found')
fr.close()
```



Updating data in binary file

```
import pickle
fr = open('student.dat','rb')
blist= []
r=int(input("enter roll no to be updated"))
m=int(input("enter correct marks"))
while True:
    try:
        rec = pickle.load(fr)
        blist.append(rec)
    except EOFError:
        break
fr.close()
for i in range(len(blist)):
    if blist[i]['Rollno']==r:
        blist[i]['Marks'] = m
```

Updating data in binary file

```
fw = open('student.dat','wb')
for x in blist:
    pickle.dump(x,fw)
fw.close()

fr = open("student.dat",'rb') #readi
while True:
    try:
        rec = pickle.load(fr)
        print('Roll Num:',rec['Rollno'])
        print('Name:',rec['Name'])
        print('Marks:',rec['Marks'])
    except EOFError:
        break
fr.close()
```

```
File  Edit  Shell  Debug  Options  Window  Help
Python 3.7.4 (tags/v3.7.4:e0
(Intel)] on win32
Type "help", "copyright", "c
>>>
= RESTART: C:\Users\user\App
enter roll no to be updated1
enter correct marks450
Roll Num: 56
Name: pritam
Marks: 500
Roll Num: 67
Name: Neeraj
Marks: 567
Roll Num: 12
Name: Rajiv
Marks: 450
>>>
```