PYTHON & CSV FILES

WHY CSV?

- Huge data is extensively handled while using social networking sites and its related applications.
- CSV files organizes data into a structured form to handle them with ease.
- CSV file formats are of plain text format which makes website developers to create applications that implements CSV.

PYTHON & CSV FILES

- Data storage and sharing is the major demand in today's industrial work environment.
- Comma Separated Values (CSV) file is a simple file format used to store tabular data, such as spreadsheet or database.
- They are easier to import into a spreadsheet or database since they are plain text(regardless of the specific software used)
- CSV files can be opened in MS Excel or in any text editor(notepad..) or database which makes them easier to read.

CSV AND XLS FILE FORMAT DIFFERENCE

CSV	EXCEL
CSV format is a plain text format with a series of values separated by commas.	Excel is a binary file that holds information about all the worksheets in a file, including both content and formatting
CSV can be opened with any text editor in Windows like Notepad, MS Excel etc.	XLS files can be only be read by applications that have been especially written to read their format, and can only written in the same way.
CSV is a format for saving tabular information into a delimited text file with extension .csv	Excel is a spreadsheet that saves files into its own proprietary format .xls or .xlsx
Importing CSV files can be much faster, and it also consumes less memory	Excel consumes more memory while importing data

CSV MODULE

- CSV module of Python gives the Programmer the ability to parse CSV files.
- A CSV file is a file is a human readable text file where each line has a number of fields, separated by commas or any delimiter.
- The command to include this module is
  - import csv

ADVANTAGES OF CSV FILES

- CSV files are smaller in size
- Faster to handle
- Human readable and easy to edit manually.
- CSV is easy to generate and import onto a spreadsheet or database.
- CSV is simple to implement and parse.
- CSV is processed by almost all existing applications.

additional qns from the supplementary

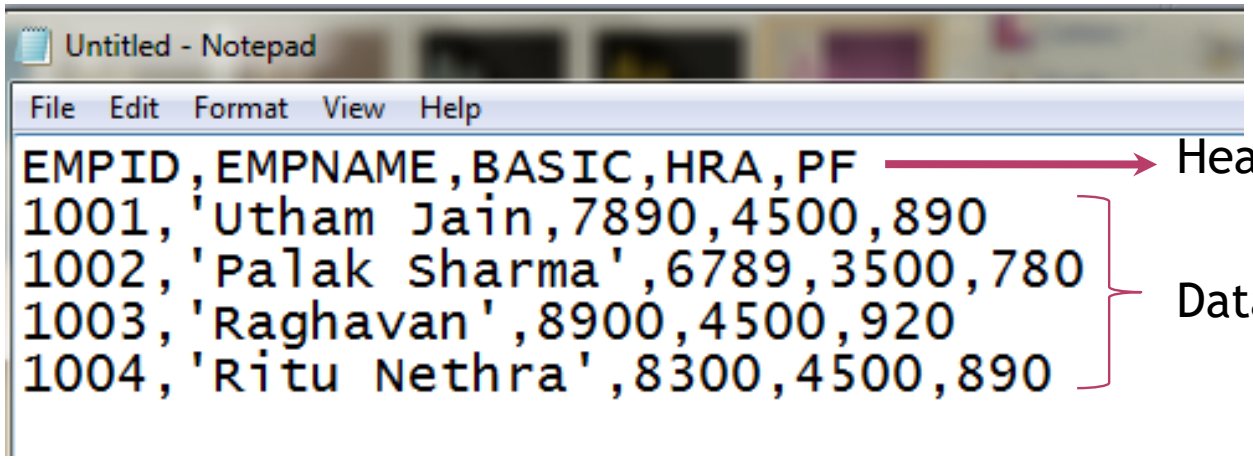
1) When a file is opened for output in append mode, what happens when
   a) mentioned file does not exist
   b) mentioned file does exist
2) How many file objects do we required
   a) to process three files sequentially
   b) to merge two sorted file into a third file
3) Is csv file different from a text file? why/why not
4) Why are csv files popular?
5) WAP that counts the number of character upto the first $ in a text file.
6) WAP that will create an object called filout for writing, associate it with the filename strs.txt. The code should keep on writing strings to it as long as the user wants.

Additional qns from supplementary

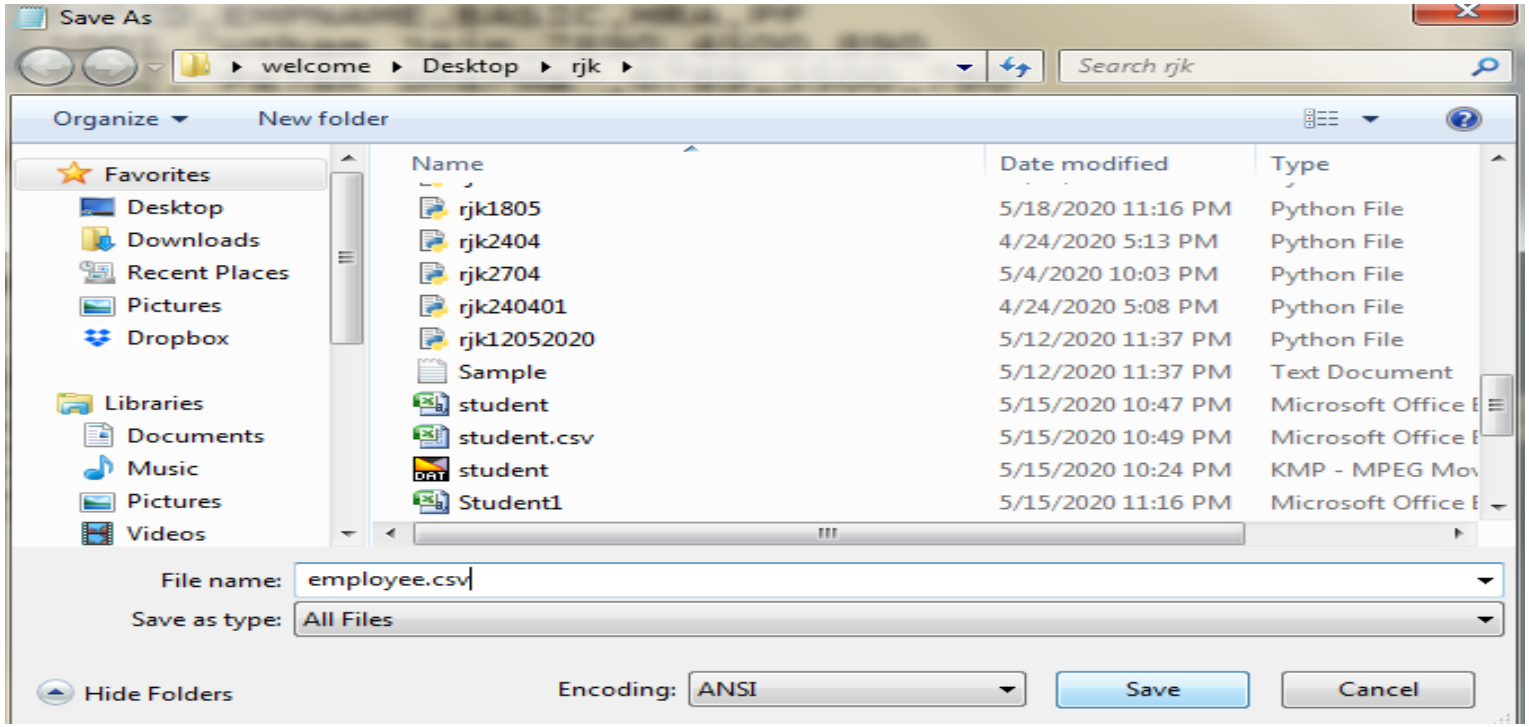
1) WAfunction that reads a csv file and creates another csv file with the same content, but with different delimiter.
2) WAfunction that reads a csv file creates another csv file with the same content except the lines beginning with ‘check’

CREATING CSV FILE USING NOTEPAD

- Open Notepad from command prompt type the following details of employees as shown below:



SAVE THE FILE WITH EXTENSION .CSV



This file is saved as employee.csv

HOW TO READ & WRITE CSV FILE USING PYTHON?

- Python's CSV library contains objects and code to read, write and process data from and to CSV files.
- CSV modules are now widely used in e-commerce applications because they are very easy to store and process.
- Two ways to read CSV files in python
  - reader() function
  - dictreader class
- To write into CSV file we use
  - writer() function
  - dictwriter class

FILE OPERATIONS IN PYTHON

- Open a file (csv file)
- Perform read or write operations (reader())
- Close the file

EXAMPLE TO READ THE CSV FILE(rjk120701)

```
# to read the content of a csv file in Python
import csv
fobj = open('employee.csv','r')
robj = csv.reader(fobj)
for row in robj:
    print(row)
print(type(robj))
fobj.close()
```

Output

```
['EMPID', 'EMPNAME', 'BASIC', 'HRA', 'PF']
['1001', "'Utham Jain", '7890', '4500', '890']
['1002', "'Palak Sharma'", '6789', '3500', '780']
['1003', "'Raghavan'", '8900', '4500', '920']
['1004', "'Ritu Nethra'", '8300', '4500', '890']
<class '_csv.reader'>
```

TO DISPLAY THE RECORDS AS IN CSV FORMAT (rjk12072)

```
# to print the contents as created in a csv file
import csv
fobj = open('employee.csv','r')
robj = csv.reader(fobj)
for r in robj:
    print(','.join(r))
fobj.close()
```

Output

```
EMPID,EMPNAME,BASIC,HRA,PF
1001,'Utham Jain,7890,4500,890
1002,'Palak Sharma',6789,3500,780
1003,'Raghavan',8900,4500,920
1004,'Ritu Nethra',8300,4500,890
```

EXAMPLE TO READ SELECTED COLUMNS OF CSV FILE (rjk120703)

```
# to read the SPECIFIC COLUMN of a csv file in
  Python
import csv
fobj = open('employee.csv','r')
robj = csv.reader(fobj)
for col in robj:
    print(col[0],col[1],col[4])
fobj.close()
```

Output

```
EMPID EMPNAME PF
1001 'Utham Jain 890
1002 'Palak Sharma' 780
1003 'Raghavan' 920
1004 'Ritu Nethra' 890
```

TO COUNT THE RECORDS(rjk12074)

```
# to count the records of a csv file in Python
import csv
fobj = open('employee.csv','r')
robj = csv.reader(fobj)
c=list(robj)
print(len(c)-1)
fobj.close()
```

Output

4

TO COUNT THE RECORDS (USING NEXT() FUNCTION)rjk12075

next() – skips the current row and moves the iterator to next row

```
# to count the records of a csv file
import csv
fobj = open('employee.csv','r')
robj = csv.reader(fobj)
next(robj)
count = list(robj)
print(len(count))
fobj.close()
```

Output

4

TO PERFORM SEARCH ON A PARTICULAR EMPID NUMBER(rjk12076)

```
# to perform search operation in a csv file
import csv
def creation():
    fobj = open('employee.csv','r')
    robj = csv.reader(fobj)
    empid = input('Enter search id')
    found = 0
    for r in robj:
                if r[0] == empid :
                print(','.join(r))
                found = 1
                break
    if found == 0:
                print('Search ID is not found ')
fobj.close()
```

```
Enter search id 1003
1003,'Raghavan',8900,4500,920
```

```
Enter search id 1005
Search ID is not found
```

TO CREATE ORDER.CSV USING PYTHON MODULE(rjk12077)

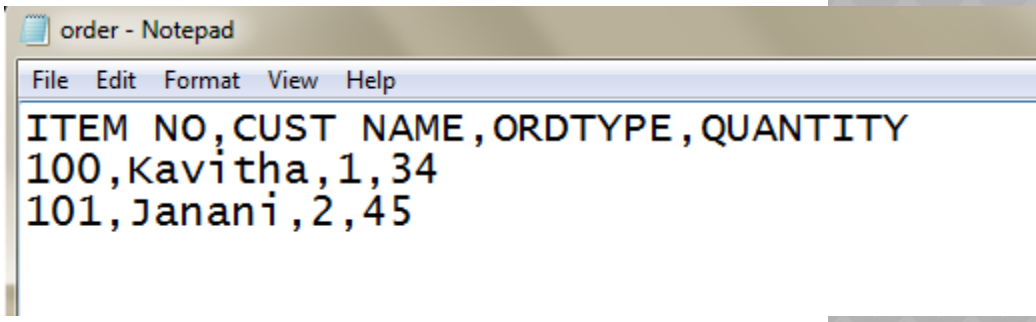
```
# to create a csv file using python csv module
import csv
header = ['ITEM NO','CUST NAME','ORDTYPE','QUANTITY']
rows=[]
wish = 'y'
while wish == 'y' :
    itemno=int(input('Enter item number '))
    custname = input('Type ur name ')
    ordtype = input('Enter 1 for vegs 2 for fruits ')
    qty = int(input('Enter quantity required '))
    r = [itemno,custname,ordtype,qty]
    rows.append(r)
    wish = input('Enter y to continue any other letter to quit ')
```

TO CREATE ORDER.CSV USING PYTHON MODULE

```
fname='order.csv'
with open(fname,'w') as fhandle:
    row = csv.writer(fhandle,delimiter=',')
    row.writerow(header)
    row.writerows(rows)
```

OUTPUT

```
Enter item number 100

Type ur name Kavitha

Enter 1 for vegs 2 for fruits 1

Enter quantity required 34

Enter y to continue any other letter to quit y

Enter item number 101

Type ur name Janani

Enter 1 for vegs 2 for fruits 2

Enter quantity required 45

Enter y to continue any other letter to quit n
```



PRACTICALS

Write a menu driven program in Python to create a CSV file with following data

- Roll no
- Name of student
- Mark in Sub1 • Mark in sub2 • Mark in sub3 • Mark in sub4 • Mark in sub5

Perform following operations on the CSV file after reading it.

- Calculate total and percentage for each student.
- Display the name of student if in any subject marks are greater than 80% (Assume marks are out of 100)

```
import csv
def creation():
    header =
['STUDID','STUDNAME','SUB1','SUB2','SUB3','SUB4','SUB5']
    rows=[]
    wish = 'y'
    while wish == 'y' :
        studid=int(input('Enter student id  '))
        for i in rows:
            if studid in i:    # validation
                print('Already exist!! No duplicates ')
                studid=int(input('Enter student id  '))
```

```
studname = input('Enter student name ')
sub1=eval(input('Enter Eng marks '))
sub2=eval(input('Enter Mat marks '))
sub3=eval(input('Enter Phy marks '))
sub4=eval(input('Enter Che marks '))
sub5=eval(input('Enter Csc marks '))
tot = sub1+sub2+sub3+sub4+sub5
per = tot/5
r =[studid,studname,sub1,sub2,sub3,sub4,sub5,tot,per]
rows.append(r)
wish = input('Enter y to continue other letter to quit ')
```

```
# writing into the file
fname='student.csv'
with open(fname,'w') as fhandle:
    row = csv.writer(fhandle,delimiter=',')
    row.writerow(header)
    row.writerows(rows)
```

```
# to read from the file and display the records
def traversing():
    fname ='student.csv'
    with open(fname) as fhandle:
        rows = csv.reader(fhandle)

        print('\nID\tName\tEng\tMat\tPhy\tChe\tCS\tTot\tAvg')
        next(rows)
        stu=[]
        for rec in rows:
            print('\t'.join(rec))
            stu.append(' '.join(rec))
```

```
print('Students who have secured >=80 in one subject\n')
for x in range(1,len(stu)):
    y=stu[x].split()
    for i in range(2,len(y)-2):
        if int(y[i])>=80:
            print('Name of the student ',y[1])
            break
```

```
while True:
    print('\n\t\t\t M E N U C H O I C E S ')
    print('\n\t\t\t\t1 Creation')
    print('\n\t\t\t\t2 Traversing')
    print('\n\t\t\t\t3 Quit')
    ch = int(input('\t\t\tEnter ur choice '))
   if ch == 1:
        creation()
    elif ch == 2:
        traversing()
    else:
        print('End of the program')
        break
```

csv reader()

csv reader()

Once the file is opened in read mode.

The file object is converted to csv.reader object.

The reader object is used to read records as lists from a csv file.

Syntax :

obj = csv.reader(file object)

csv writer()

csv writer()

To write to a csv file in Python, we use csv.writer() functions.

The csv.writer() function returns a writer object that converts the user's data into a delimited string.

This string can later be used to write into CSV files using the writerow() function.

writerow() method allows to write a list of fields to the file.

writerow() is used to write each row.

writerows() is used write all rows in one go.

csv writer()

csv writer()

Syntax :

obj = csv.writer(filehandle,delimiter=value)

obj.writerow(single row of values)

obj.writerows(multiple rows of values)

line_num object of CSV file

_csv.reader object has a method called line_num that returns the number of lines in the CSV file.

line_num is nothing but a counter which returns the number of rows which have been iterated.

line_num object of CSV file

```
import csv
fhandle= open('student.csv','r')
rec = csv.reader(fhandle)
print('ROLLNO\t\tNAME\t\tENG\t\tMAT\t\tPHY\t\tCHE\t\tCS\t\tTOT\t\tAVG')
print('-'*80)
for i in rec:
    if rec.line_num==1 :
        continue
    print('\t\t'.join(i))
print('-'*80)
print(rec.line_num)
fhandle.close()
```

Write a program to append into csv file

```
import csv
f=open('emp.csv',’a')
ans='y'
rec = csv.writer(f,delimiter=',')
while ans=='y':
    eno=int(input('Enter employee number '))
    name=input('Enter name ')
    rec.writerow([eno,name])
    ans = input('Press y to continue ')
f.close()
```

1. Write user-defined function to perform read and write operation onto a student.csv file having fields as roll number, name, stream and marks.

1. Write a Python program to read specific columns from department.csv and print the contents of dept_id, dept_loc.

   [dept_id,dept_name,dept_loc]

```
import csv
def read():
    f = open('stud.csv')
    rec = csv.reader(f)
    for i in rec:
        print(i)
    f.close()
def write():
    f = open('stud.csv','w')
    obj = csv.writer(f,delimiter=',')
    obj.writerow(['studid','name','stream','marks'])
    ch = 'Y'
```

```
while ch=='Y':
        studid =int(input('Enter rollnum'))
        name = input('Enter name ')
        stream=input('Enter stream ')
        marks=float(input('Enter marks '))
        obj.writerow([studid,name,stream,marks])
        ch =input('Enter Y to continue ')
    f.close()
opt=int(input('1 for read 2 for write'))
if opt == 1:
    read()
elif opt==2:
    write()
```

```
import csv
f = open('department.csv','w')
fobj=csv.writer(f,delimiter=',')
fobj.writerow(['department_id','department_name','members','loc'])
fobj.writerow([1001,'csc',10,'main block'])
fobj.writerow([1002,'math',12,'annex block'])
fobj.writerow([1003,'phy',14,'main block'])
f.close()
with open('department.csv') as f:
    data = csv.reader(f)
    print("-------------------------------")
    for i in data:
        print(i[0],'\t\t', i[3])
        print("-------------------------------")
```

PRACTICAL - 9

Write a menu driven program in Python using function to read a text file and

Count number of characters

Count number of words

Count number of vowels

Count number of lines

Count number of digits

Count number of special characters

```
# char, word, vowels, lines, digits, special char
# Practical 9 - Text Files - count
def creation(fname):
    fobj = open(fname,'w')
    choice = 'Y'
    while choice=='Y':
        txt = input('Enter the text ')
        fobj.write(txt+'\n')
        choice = input('Enter Y to continue ')
    fobj.close()
```

```
def accessing(fname):
    fobj = open(fname,'r')
    char=word=vowel=line=digit=spchar=0
    txt = fobj.read()
    for i in txt:
        if i==' ' or i=='\n':
            word +=1
        if i !=' ' and i !='\n':
            char +=1
        if i in 'aeiouAEIOU':
            vowel +=1
        if i == '\n':
            line +=1
```

```
    if i.isdigit():
        digit +=1
    if not(i.isalnum()):
        spchar +=1
        if i.isupper():
                upper+=1
print('Total characters :',char)
print('Total words :',word)
print('Total vowels :',vowel)
print('Total lines :',line)
print('Total digits :',digit)
print('Total special characters :',spchar-line)
fobj.close()
```

```
while True:
    print('\n \t\t MENU CHOICES ')
    print('\n\t\t 1 : Creation ')
    print('\n\t\t 2 : Accessing ')
    print(‘\n\t\t3 : Quit’)
    ch = eval(input('Enter ur choice :'))
    if ch == 1:
        creation('sample.txt')
    elif ch == 2:
        accessing('sample.txt')
    elif ch == 3:
        break
```

PRACTICAL - 10

Write a menu driven program in Python using function to read a text file and

Display number of times each word appears in the text

Display words with maximum and minimum length

Display words that start with capital letter

```
def accessing(fname):
    fobj = open(fname,'r')
    txt = fobj.read()
    txt=txt.split()
    print()
    # each word count
    c={}
    for i in range(len(txt)):
        x = str(txt[i])
        c[x]=txt.count(x)
    for i in c:
        print(i,'\tappears\t',c[i],'\ttimes ')
    print()
```

```
# maximum length word
    lst = list(txt)
    max =min= lst[0]
    for i in range(1,len(lst)):
        if lst[i][0].isupper():
            print(lst[i],' word starts with capital letter')
        if len(max) <len(lst[i]):
            max = lst[i]
        if len(min) > len(lst[i]):
            min = lst[i]
    print('\nWord with minimum length ',min)
    print('\nWord with maximum length ',max)
    fobj.close()
```

PRACTICAL - 11

Write a menu driven program in Python using pickle to create a binary file with the following structure

Admission Number

Student name

Age

Display the contents of the binary file

Display the student whose age is above the user given value

Search a student when the admission number is given by the user

```
# Practical 11 - Binary Files - student
import pickle
def creation(fname):
    fobj = open(fname,'wb')
    rec=[]
    while True:
        admno=int(input('Enter Admission Number '))
        stuname=input('Enter ur name ')
        age = int(input('Enter ur age '))
        temp=[admno,stuname,age]
        rec.append(temp)
        ch=input('Enter Y to continue ')
        if ch !='Y':
            break
    pickle.dump(rec,fobj)
    fobj.close()
```

```
def accessing(fname):
    fobj=open(fname,'rb')
    rec = pickle.load(fobj)
    for i in rec:
        print('Admission Number ',i[0])
        print('Name ',i[1])
        print('Age ',i[2])
        print()
    fobj.close()
```

```
def agesearch(fname,age):
    fobj=open(fname,'rb')
    rec = pickle.load(fobj)
    flag = 0
    for i in rec:
        if i[2]>=age:
            print('Admission Number ',i[0])
            print('Name ',i[1])
            print('Age ',i[2])
            print()
            flag = 1
    if flag == 0 :
        print('No records above the age specified')
    fobj.close()
```

```
def search(fname,admno):
    fobj=open(fname,'rb')
    rec = pickle.load(fobj)
    flag = 0
    for i in rec:
        if i[0]==admno:
            print('Admission Number ',i[0])
            print('Name ',i[1])
            print('Age ',i[2])
            print()
            flag = 1
            break
    if flag == 0:
        print('Record Not found ')
    fobj.close()
```

```
while True:
    print('\n \t\t MENU CHOICES ')
    print('\n\t\t 1 : Creation of records ')
    print('\n\t\t 2 : Display of records ')
    print('\n\t\t 3 : Records age exceed the user value')
    print('\n\t\t 4 : Search by Admission Number')
    print('\n\t\t 5 : Quit')
    ch = eval(input('Enter ur choice :'))
    if ch == 1:
        creation('sample.dat')
    elif ch == 2:
        accessing('sample.dat')
    elif ch == 3:
        age = int(input('Enter the age limit '))
        agesearch('sample.dat',age)
    elif ch == 4:
        admno=int(input('Enter the admission number to search '))
        search('sample.dat',admno)
    elif ch == 5:
        break
```

Practical 12

Write a menu driven program in Python using pickle library to create a binary file with following structure :

Travel id

From

To

Append a record to the file

Delete a record based on Travel id

Update a record based on Travel id

```
# Practical 12 - Binary Files - travel
import pickle
import os
def creation(fname):
    fobj = open(fname,'wb+')
    dic={}
    d1={}
    while True:
        travelid=int(input('Enter Travel Id '))
        src=input('Enter source location ')
        dest= input('Enter destination location ')
        d1[travelid]=[src,dest]
        dic.update(d1)
        ch=input('Enter Y to continue ')
        if ch !='Y':
            break
    pickle.dump(dic,fobj)
    fobj.close()
```

```
def accessing(fname):
    fobj=open(fname,'rb')
    rec = pickle.load(fobj)
    for i in rec.keys():
        print('Travel ID :',i)
        print('Source Location :',rec[i][0])
        print('Destination Location :',rec[i][1])
        print()
    fobj.close()
```

```
def deletion(fname,travelid):
    fobj = open(fname,'rb')
    fobj1=open('temp.dat','wb')
    rec = pickle.load(fobj)
    if travelid in rec.keys():
        rec.pop(travelid)
    else:
        print('Travel ID is invalid')
    pickle.dump(rec,fobj1)
    fobj.close()
    fobj1.close()
```

```
def update(fname,travelid):
    fobj = open(fname,'rb')
    fobj1=open('temp.dat','wb')
    rec = pickle.load(fobj)
    ndict={}
    if travelid in rec.keys():
            src =input('Enter new source ')
            des = input('Enter new destination ')
            ndict[travelid]=[src,des]
    else:
        print('Travel ID is invalid')
    rec.update(ndict)
    pickle.dump(rec,fobj1)
    fobj.close()
    fobj1.close()
```

```
while True:
    print('\n \t\t MENU CHOICES ')
    print('\n\t\t 1 : Creation/appending of records ')
    print('\n\t\t 2 : Displaying of records ')
    print('\n\t\t 3 : Deleting of record ')
    print('\n\t\t 4 : Updation of record ')
    print('\n\t\t 5 : Quit')
    ch = eval(input('Enter ur choice :'))
    if ch == 1:
        creation('travel.dat')
    elif ch == 2:
        accessing('travel.dat')
    elif ch == 3:
        travelid = int(input('Enter travel id to delete '))
        deletion('travel.dat',travelid)
        os.remove('travel.dat')
        os.rename('temp.dat','travel.dat')
    elif ch == 4:
        travelid=int(input('Enter travel id to update '))
        update('travel.dat',travelid)
        os.remove('travel.dat')
        os.rename('temp.dat','travel.dat')
    elif ch == 5:
        break
```