File Handling

Topics to be covered

1. Introduction
2. Types of files
3. Access specifiers in the file
4. File object methods seek( ) and tell( )
5. Text files and its methods
6. Binary file and its methods
7. CSV files and its methods
8. Absolute and Relative path

FILE HANDLING

Files are used to store data permanently and can be retrieved later.

Type of Files

1. Text Files
2. Binary Files
3. CSV Files

Steps for File handling:

1. Open File
2. Read/Write
3. Close File

Open Files: open( ) function is used to open files in python.

There are two ways to open files in python:

1. **file_object = open("file name", " access specifier")**
    a. i.e. f=open("test.txt","r") #here our test file exists in the same directory of python.
    b. i.e. f=open(r"C:\Users\anujd\AppData\Local\Programs\Python\Python311\test.txt","r")
    Use 'r' before path indicates the data within quotes will be read as raw string and no special meaning attached with any character.

    Or

    f=open("C:\\Users\\anujd\\AppData\\Local\\Programs\\Python\\Python311\\\test.txt","r")

    The slash in the path has to be doubled.

    c. In this we need to close the file.
    d. In this mode if we are writing into a file then we have to close the file otherwise our data will not be written into the file till now the file is not closed. The data remains in the output buffer and when we close the file then data is shifted from the output buffer to the file.
    e. flush( ): It forces the data waiting in the buffer immediately written into the file without waiting for closing of file.

2. **with open("file name","access specifier") as file_object:**
    a. i.e. with open("test.txt","r") as f:

b. i.e. with open(r"C:\Users\anujd\AppData\Local\Programs\Python\Python311\test.txt","r") as f:
Or

with open("C:\\Users\\anujd\\AppData\\Local\\Programs\\Python\\Python311\\test.txt","r") as f:

c. In this there is no need to close the file. It will automatically close the file.

Close Files: close( ) function is used to close files in python.

a. file_object.close( )
b. i.e. f.close( )

Access Specifiers in Files:

Access Mode for Text Files	Access Mode for Binary Files	Access Mode for CSV Files	Description	File Pointer Position
r	Rb	r	Read mode. Opens a file for reading.If the file does not exist, open() raises a FileNotFoundError.	Beginning of File
r+	rb+		It opens the file for both reading and writing. If the file does not exist, open() raises a FileNotFoundError.	Beginning of File
w	Wb	w	Write mode. It opens the file for writing only. If the file exists, the content of the file will be removed. If the file does not exist, it is created.	Beginning of File
w+	wb+		The w+ mode opens the file for both writing and reading. Like w, if the file exists, then the content of the file will be removed. If the file does not exist, it is created.	Beginning of File

a	Ab	a	The a mode opens the file for appending. In this the new content is added after the existing content. If the file does not exist, it creates the new file.	End of File
a+	ab+		The a+ mode opens the file for both appending and reading. In this the new content is added after the existing content. If the file does not exist, it is created.	End of File

Default mode for file opening in "r" read mode. If we didn't specify mode during the opening of the file then it will automatically open the file in read mode.

File Object Methods (seek( ) & tell( ))

Method	Prototype	Description
seek( )	**Syntax: <file_object>.seek(<offset>,<from_where>)** where: offset: number of positions to be more forward from_where: it defines to reference point Then it returns the new absolute position. i.e. **f.seek(10,0)** Here, f is the file handle, 10 is the offset (it moves the cursor 10 bytes forward), 0 means reference point at the beginning of file.	seek() function is used to change the position of the **File Handle** to a given specific position. File handle is like a cursor, which defines from where the data has to be read or written in the file. **0**: sets the reference point at the beginning of the file. **(default)** **1**: sets the reference point at the current file position. **2**: sets the reference point at the end of the file **Note:** In the case of a text file we can use only '**0**' as a reference point.
tell( )	**Syntax: <file_object>.tell( )**	tell() function returns the current position of the file object. This

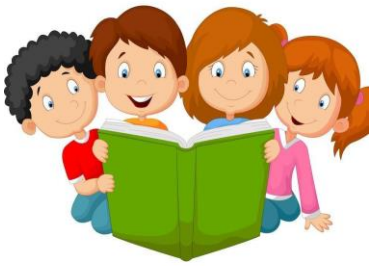
	I.e. position=f.tell( ) Here, position will hold the integer value of the file pointer returned by tell function. f is the file handle.	method takes no parameters and returns an integer value. Initially the file pointer points to the beginning of the file(if not opened in append mode). So, the initial value of tell() is zero.

Text Files:

- It stores information in the form of ASCII or Unicode characters
- Each line of text is terminated with a special character called EOL (End of Line), which is the new line character ('\n') in python by default.
- File extension will be .txt

Working with Text Files:

1. Reading data from a file.
2. Writing data into a file.



Reading data from files

There are three ways to read data from text file:

1. read function i.e. read( )
2. readline function i.e. readline( )
3. readlines function i.e. readlines( )

read( ) : It is used in text files to read a specified number of data bytes from the file. It returns the result in the form of a string.

Syntax: file_object.read( )

file_pointer.read(n): It will read the maximum n bytes/characters from the file.

f.read(7) # it will read 7 bytes/characters from the position of file pointer.

file_pointer.read( ): It will read the entire content of the file.

f.read( ) # it will read all the data of the file from the position of file pointer.

readline( ): It will read one complete line in one go from the file. It returns the data in the form of a string.

Syntax: file_object.readline( )

file_pointer.readline( ): It will read the entire line.

f.readline( ) #it will read one complete line in one go.

file_pointer.readline(n): It will read the first 'n' bytes from the file.

f.readline(5) #it will read the first 5 characters/bytes from the file.

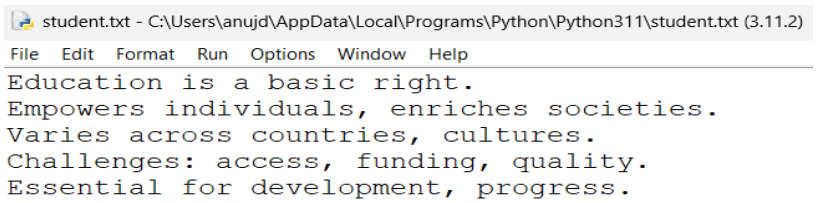
readlines( ): It will return all the lines of the file as the elements of the list. I.e. the 1st line of the file will be the first element of the list and so on.

Syntax: file_object.readlines( )

file_pointer.readlines( ): It will read all the lines of the file as the elements of the list.

f.readlines( ) #it will read all the lines of the file as the elements of the list.

File Content



Code

```
f=open("student.txt","r")
data=f.read()
print(data)
```

Output

```
>>>
== RESTART: C:\Users\anujd\AppData\Local\P
Education is a basic right.
Empowers individuals, enriches societies.
Varies across countries, cultures.
Challenges: access, funding, quality.
Essential for development, progress.
```

```
f=open("student.txt","r")
data=f.read(7)
print(data)
```

```
>>>
= RESTART: C:/Users/anujd/
Educati
>>>
```

```
f=open("student.txt","r")
data=f.readline()
print(data)
data=f.readline()
print(data)
```

```
>>>
= RESTART: C:/Users/anujd/AppData/Local/P
Education is a basic right.

Empowers individuals, enriches societies.
```

```
f=open("student.txt","r")
data=f.readline()
print(data,end="")
data=f.readline()
print(data,end="")
```

```
>>>
= RESTART: C:/Users/anujd/AppData/Local/P
Education is a basic right.
Empowers individuals, enriches societies.
>>>
```

```
f=open("student.txt","r")
data=f.readline()
print(data,end="")
data=f.readline(5)
print(data,end="")
data=f.readline(7)
print(data,end="")
```

```
>>>
========== RESTART: C:/Use
Education is a basic right.
Empowers ind
```

```
f=open("student.txt","r")
data=f.readlines()
print(data)
```

```
>>>
========== RESTART: C:/Users/anujd/AppData/Lo
cal/Programs/Python/Python311/filedemo1.py ====
=======
['Education is a basic right.\n', 'Empowers ind
ividuals, enriches societies.\n', 'Varies acros
s countries, cultures.\n', 'Challenges: access,
funding, quality.\n', 'Essential for developmen
t, progress.\n']
>>>
```

Result is in the form of a list.

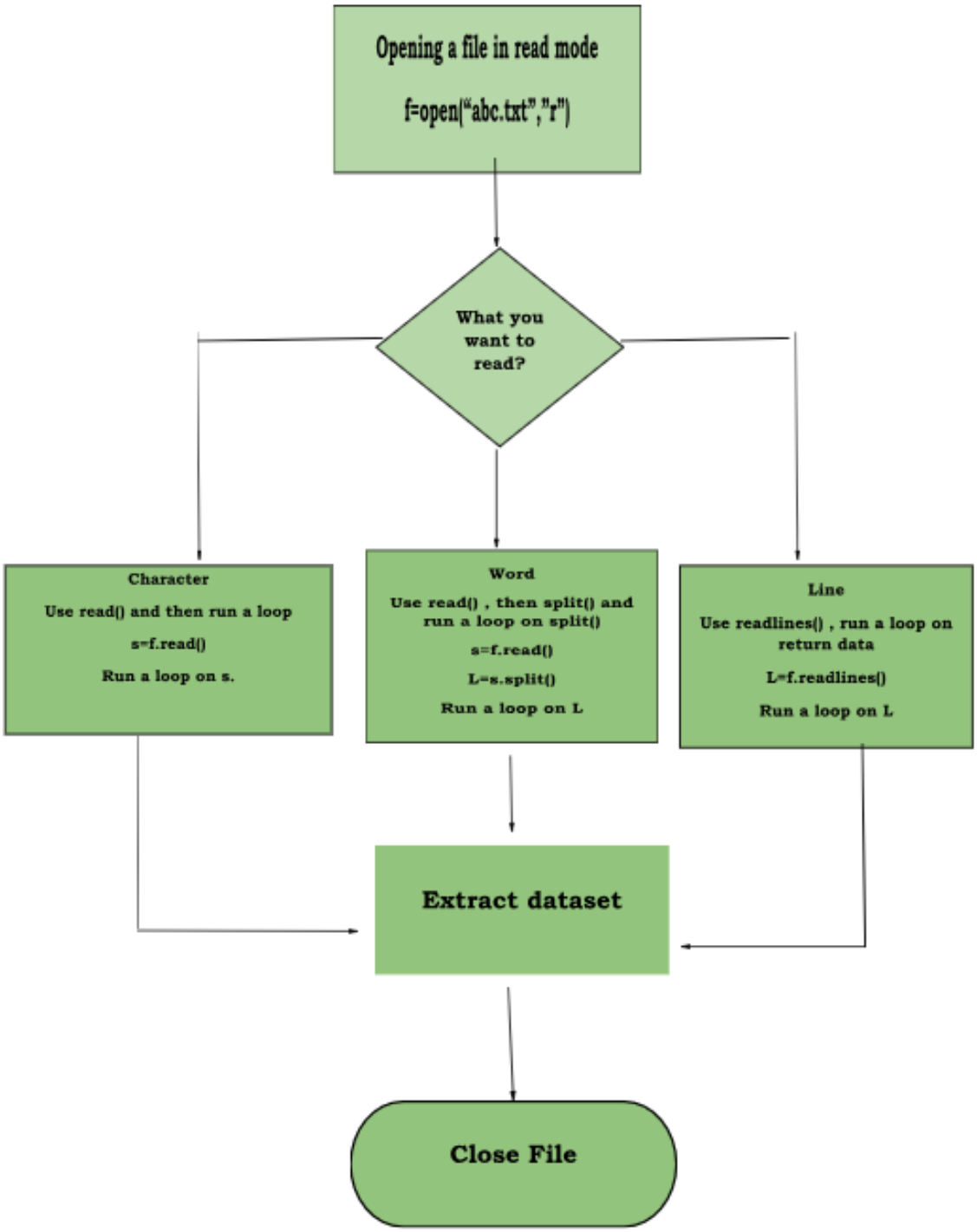
```
f=open("student.txt","r")
data=f.readlines()
for i in data:
    print(i,end="")
```

```
>>>
= RESTART: C:/Users/anujd/AppData/Local/P
s/Python/Python311/filedemo1.py
Education is a basic right.
Empowers individuals, enriches societies.
Varies across countries, cultures.
Challenges: access, funding, quality.
Essential for development, progress.
>>>
```

Tips on writing text file code in exam:

- Let the file name be “abc.txt”.
- Read the question carefully and find out what has to be read from the file.

Follow the below flow chart to write the code.



Example 1:

Write a function count_char() that reads a file named "char.txt" counts the number of times character "a" or "A" appears in it.

```
def count_word():
    c=0 # variable to keep track of count
    f=open("char.txt","r") # step 1: open the file in read mode
    '''since the words are to be counted ,
     we will use read()'''
    s=f.read() # step 2, read entire data in s

    ''' run a loop on s to pick one character at a time'''
    for i in s:
        if i.upper()=="A" :
            c+=1 # increase the value of couter variable by 1
    f.close()
    return c
```

Example 2:

Write a function count_word() that reads a text file named "char.txt" and returns the number of times word " the" exists.

```
def count_word():
    c=0 # variable to keep track of count
    f=open("char.txt","r") # step 1: open the file in read mode
    '''since the words are to be counted ,
     we will use read() to fetch data and use
     a split() on it '''
    s=f.read() # step 2, read entire data in s
    L=s.split() # a list L is created with list of words in entire file
    ''' run a loop on L to pick one word at a time'''
    for i in L:
        if i.upper()=="THE" :
            c+=1 # increase the value of couter variable by 1
    f.close()
    return c
```

Example 3:

Write a function count_line() that reads a text file named "char.txt" and returns the number of lines that start with a vowel.

```
def count_word():
    c=0 # variable to keep track of count
    f=open("char.txt","r") # step 1: open the file in read mode
    '''since the words are to be counted ,
     we will use read()'''
    s=f.readlines() # step 2, read list of lines in s

    ''' run a loop on s to pick one line at a time'''
    for i in s:
        if i[0] in "aeiouAEIOU":
            c+=1
    f.close()
    return c
```

Writing data into Files

If the file doesn't exist then it will create the file.

1. **write( ):** It takes string as an input and writes it into the file.
    a. **Syntax**: file_object.write(string)
    b. i.e. f.write("Hello World")

2. **writelines( ):** It is used to write multiple lines as a list of strings into the file. In this each element of the list will be treated as a separate line in the file.
    a. **Syntax:** file_object.writelines(list of strings)
    b. I.e. data=["I am a student of DOE", "I studies in class 12th"]
    f.writelines(data)

Code	Output

`f=open("toppers.txt","w")`
`f.write("Priya")`
`f.write("Disha")`
`f.write("Tanish")`
`f.write("Krishna")`
`f.write("Aman")`
`f.close()`
	Content in "Topper.txt" file
	toppers.txt - C:\Users\anujd\AppData\Local\Programs\Python\Python311\t
	File Edit Format Run Options Window Help
	PriyaDishaTanishKrishnaAman
	
	But, we want these names in separate lines.

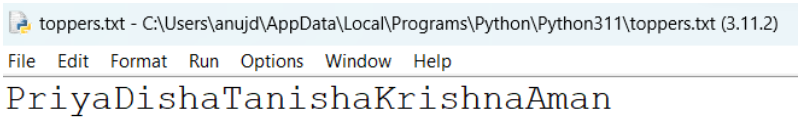
`f=open("toppers.txt","w")`
`f.write("Priya\n")`
`f.write("Disha\n")`
`f.write("Tanish\n")`
`f.write("Krishna\n")`
`f.write("Aman\n")`
`f.close()`
	toppers.txt - C:\Users\anujd\AppData\
	File Edit Format Run Options W
	Priya
	Disha
	Tanish
	Krishna
	Aman

```
with open("topper.txt","w") as f:
    for i in range(5):
        name=input("Enter the name of toppers one by one : ")
        f.write(name)
```

Output:

```
>>>
============ RESTART: C:/Users/anujd/AppData/Local
Enter the name of toppers one by one : Priya
Enter the name of toppers one by one : Disha
Enter the name of toppers one by one : Tanisha
Enter the name of toppers one by one : Krishna
Enter the name of toppers one by one : Aman
>>>
```

Content of "Topper.txt" File:



Question: Write a program in python with reference to above program, the content of the text files should be in different lines.

I.e.
Priya
Disha
Tanisha
Krishna
Aman

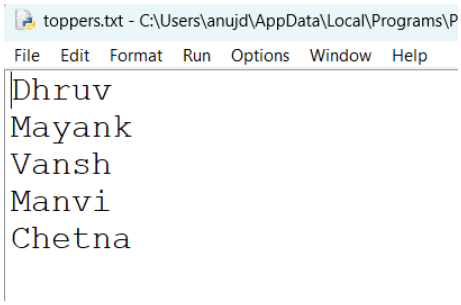
Code

```
with open("toppers.txt","w") as f:
    data=[]
    for i in range(5):
        name=input("Enter the name of toppers one by one : ")
        data.append(name+"\n")
    f.writelines(data)
```

Output

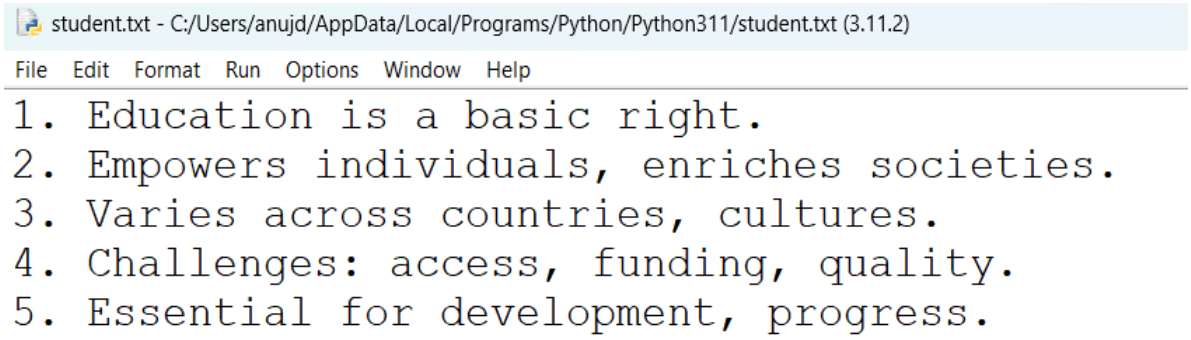
```
>>>
    = RESTART: C:/Users/anujd/AppData/Local/Programs/
    Enter the name of toppers one by one : Dhruv
    Enter the name of toppers one by one : Mayank
    Enter the name of toppers one by one : Vansh
    Enter the name of toppers one by one : Manvi
    Enter the name of toppers one by one : Chetna
>>>
```

Content of "**Toppers.txt**" file:



Write a program in python to count vowels, consonants, digits, spaces, special characters, spaces, words and lines from a text file named "student.txt".

Content of File:



Code:

```
File  Edit  Format  Run  Options  Window  Help
vowel_count = 0
consonant_count = 0
digit_count = 0
special_count = 0
space_count=0
word_count = 0
line_count = 0
f=open("student.txt","r")
for line in f:
    line_count = line_count + 1
    words = line.split()
    word_count = word_count + len(words)
    for i in line:
        if i.lower() in "aeiou":
            vowel_count = vowel_count + 1

        elif i.lower() in "bcdfghjklmnpqrstvwxyz":
            consonant_count = consonant_count + 1

        elif i in "0123456789":
            digit_count = digit_count + 1

        elif i.isspace():
            space_count = space_count+ 1

        else:
            special_count = special_count + 1
print("Vowels:", vowel_count)
print("Consonants:", consonant_count)
print("Digits:", digit_count)
print("Special Characters:", special_count)
print("Spaces:", space_count)
print("Words:", word_count)
print("Lines:", line_count)
```

Output:

```
>>>
    = RESTART: C:/Users/anujd/AppData/I
    Vowels: 59
    Consonants: 89
    Digits: 5
    Special Characters: 16
    Spaces: 27
    Words: 26
    Lines: 6
>>>
```

Exercise

1. Define a function SGcounter() that counts and display number of S and G present in a text file 'A.txt"
   e.g., SAGAR JOON IS GOING TO MARKET.

   It will display S:2 G:2

2. Write a function in Python that counts the number of "is", "am" or "are" words present in a text file "HELLO.TXT". If the "HELLO.TXT" contents are as follows:
   Here are two sentences that contain "is," "am," or "are":

   "She is studying for her final exams.

   We are planning a trip to the mountains next weekend."

   The output of the function should be: Count of is/am/are in file: 2

3. Write a method in python to read lines from a text file HELLO.TXT to find and display the occurrence of the word "hello".

4. Write a user-defined function named Count() that will read the contents of a text file named "India.txt" and count the number of lines which start with either "I" or "T".
   E.g. In the following paragraph, there are 2 lines starting with "I" or "T":

   "The Indian economy is one of the largest and fastest-growing in the world, characterized by a diverse range of industries including agriculture, manufacturing, services, and information technology. It boasts a sizable consumer base and a dynamic entrepreneurial spirit. However, it also faces challenges such as income inequality, poverty, and infrastructure gaps, which the government continually addresses through policy reforms and initiatives to foster sustainable growth and development."

5. Write a method in python to read lines from a text file AZAD.TXT and display those lines, which are starting with an alphabet 'T'.

Binary Files:

1. Binary files are made up of non-human readable characters and symbols, which require specific programs to access its contents.
2. In this translation is not required because data is stored in bytes form.
3. Faster than text files.

4. pickle module is used for working with binary files
    a. import pickle
5. File extension will be .dat
6. There is no delimiter to end the file.

Working in Binary files:

Pickle module: pickle module is used in binary file for load( ) and dump( ) methods which are used for reading and writing into binary file respectively.

Pickling: It is the process of converting python object into byte stream. Pickling is done at the time of writing into a binary file.

Unpickling: It is the process of converting a byte stream into python object. Unpickling is done at the time reading from a binary file.

dump( ): it is used to write data into binary file.

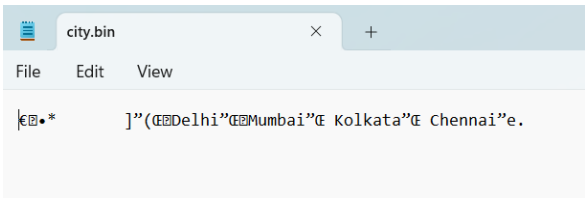
Syntax: identifier = pickle.dump(data , file_pointer)

Example: a= “My name is Anuj”

pickle.dump(a,f) #here ‘a’ contains data and ‘f’ is a file pointer.

```
#Write a program to write city names in binary file

import pickle
f=open("city.bin","wb")
city=["Delhi","Mumbai","Kolkata","Chennai"]
pickle.dump(city,f)
f.close()
```



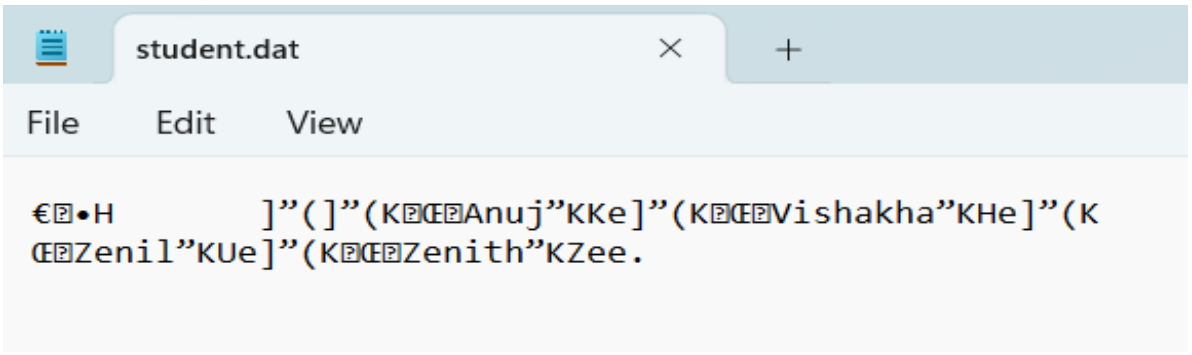
Program Code

```
#Write a program in python
#To write student details into a binary file
import pickle
f=open("student.dat","wb")
record=[]
while True:
    rno=int(input("Enter the Roll No : "))
    name=input("Enter the Name : ")
    marks=int(input("Enter the Marks : "))
    x=[rno,name,marks]
    record.append(x)
    ch=input("Press 'y' for input more details : ")
    if ch not in "yY":
        break
pickle.dump(record,f)
f.close()
```

Input Details

```
>>>
========== RESTART: C:/Users
Enter the Roll No : 1
Enter the Name : Anuj
Enter the Marks : 75
Press 'y' for input more details : y
Enter the Roll No : 2
Enter the Name : Vishakha
Enter the Marks : 72
Press 'y' for input more details : y
Enter the Roll No : 10
Enter the Name : Zenil
Enter the Marks : 85
Press 'y' for input more details : y
Enter the Roll No : 15
Enter the Name : Zenith
Enter the Marks : 90
Press 'y' for input more details : n
>>>
```

File Content



load( ): it is used to read data from binary file.

Syntax: identifier = pickle.load(file_pointer)

Example: data = pickle.load(f) #Here ‘data’ is an identifier and ‘f’ is a file pointer.

```
#Write a program to read data from city.bin
import pickle
f=open("city.bin","rb")
city=pickle.load(f)
print("City Names are : ",city)
f.close()
```

```
>>>
== RESTART: C:/Users/anujd/AppData/Local/Programs/Python/Pythor
City Names are :  ['Delhi', 'Mumbai', 'Kolkata', 'Chennai']
>>>
```

```
#Write a program in python to read
#student details from a binary file named "student.dat"

import pickle
f=open("student.dat","rb")
record=pickle.load(f) #record contains the whole content of file
for i in record:    #accessing records one by one
    print(i)
f.close()
```

```
>>>
== RESTART: C:/Users/anujd/AppI
[1, 'Anuj', 75]
[2, 'Vishakha', 72]
[10, 'Zenil', 85]
[15, 'Zenith', 90]
>>>
```

Question: Write a menu based program in python which contain student details in binary file and should have following facilities:

1. Writing student details.
2. Display all students' details
3. Search particular student details
4. Update any student details
5. Delete any student details
6. Exit

```
import pickle
import os

def insert():
    f=open("student.dat","ab")
    roll_no=int(input("Enter the Roll Number : "))
    name=input("Enter the Name : ")
    marks=int(input("Enter the Marks : "))
    student={"Roll_No":roll_no,"Name":name,"Marks":marks}
    pickle.dump(student,f)
    f.close()
    print("student details inserted successfully\n")

def display():
    f=open("student.dat","rb")
    try:
        while True:
            student=pickle.load(f)
            print(student)
    except:
        f.close()

def search():
    f=open("student.dat","rb")
    r=int(input("Enter the Roll Number to be searched : "))
    counter=0
```

```
    try:
        while True:
            student = pickle.load(f)
            if student["Roll_No"] == r:
                print("Student Details:")
                print("Roll Number:", student["Roll_No"])
                print("Name:", student["Name"])
                print("Marks:", student["Marks"])
                counter=1
    except:
        pass
    finally:
        if counter==0:
            print("Student Not Found")
        f.close()

def update():
    f=open("student.dat","rb")
    r = int(input("Enter Roll Number to update: "))
    temp_file = open("temp.dat", "wb")

    try:
        while True:
            student = pickle.load(f)
            if student["Roll_No"] == r:
                print("Current Details:")
                print("Roll Number:", student["Roll_No"])
                print("Name:", student["Name"])
                print("Marks:", student["Marks"])
                name = input("Enter The New Name: ")
                marks = int(input("Enter The New Marks: "))
                student["Name"] = name
                student["Marks"] = marks
                pickle.dump(student, temp_file)
                print("Student details updated successfully.")
            else:
                pickle.dump(student, temp_file)
    except :
        pass
    finally:
        f.close()
        temp_file.close()
        os.remove("student.dat")
        os.rename("temp.dat", "student.dat")

def delete():
    f=open("student.dat","rb")
    r = int(input("Enter The Roll Number to be deleted: "))
    temp_file = open("temp.dat", "wb")

    try:
        while True:
```

```
            student = pickle.load(f)
            if student["Roll_No"] != r:
                pickle.dump(student, temp_file)
    except EOFError:
        pass
    finally:
        f.close()
        temp_file.close()
        os.remove("student.dat")
        os.rename("temp.dat", "student.dat")
        print("Student deleted successfully.")

ch='y'
while ch in "yY":
    print("\nMenu:")
    print("1. Insert Student Details")
    print("2. Display All Students Details")
    print("3. Search Student Details")
    print("4. Update Student Details")
    print("5. Delete Student Details")
    print("6. Exit\n")
    choice=int(input("Enter your choice : "))
    if choice==1:
        insert()
    elif choice==2:
        display()
    elif choice==3:
        search()
    elif choice==4:
        update()
    elif choice==5:
        delete()
    elif choice==6:
        print("\n\t------ THE END --------")
        break
    else:
        print("\n\t------ Invalid Choice ------")
    ch=input("\nPress 'y' if you want to continue again : ")
```

Comma Separated Value (CSV) Files:

1. It is a plain text file which stores data in a tabular format, where each row represents a record and each column represents a field and fields are separated by comma in csv files.
2. csv module is used for working with csv files
    a. import csv
3. File extension for csv file will be .csv
4. CSV module provides two main classes for working with csv files are:

a. reader
b. writer

5. CSV reader object can be used to iterate through the rows of the CSV file.
6. CSV writer object that can be used to write row(s) to the CSV file.
    a. writerow(): This method is used to write a single row to the CSV file.
    b. writerows(): This method is used to write multiple rows to the CSV file.
7. We can read csv file data in excel file also.

1. Reader Function:

a. For reading data from csv file we require **csv. reader( )** function.

2. Writer Function:

a. For writing data to the csv file we take a file object as input and write the data into the file.
b. writerow( ): It writes a single row of data to the CSV file.
c. writerows( ): It writes a list of rows of data to the CSV file.

WRITING INTO CSV FILES

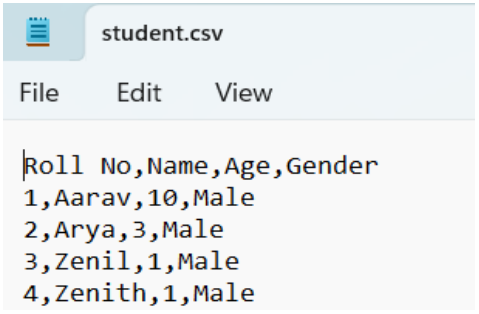
Code:

```
import csv
f=open("student.csv","w",newline="")
data=csv.writer(f)
data.writerow(["Roll No","Name","Age","Gender"])
data.writerow([1,"Aarav",10,"Male"])
data.writerow([2,"Arya",3,"Male"])
data.writerow([3,"Zenil",1,"Male"])
data.writerow([4,"Zenith",1,"Male"])
print("Data Inserted Successfully")
f.close()
```

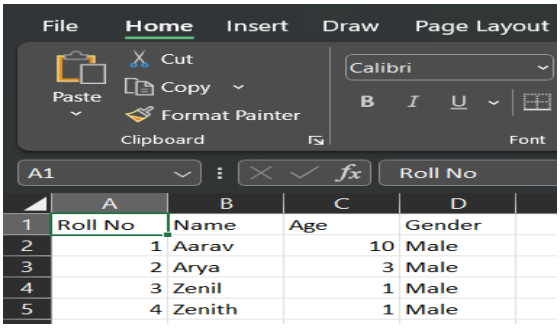
Output in IDLE:

```
>>>
    == RESTART: C:/Users/anujd/
    Data Inserted Successfully
>>>
```

Result in Notepad:



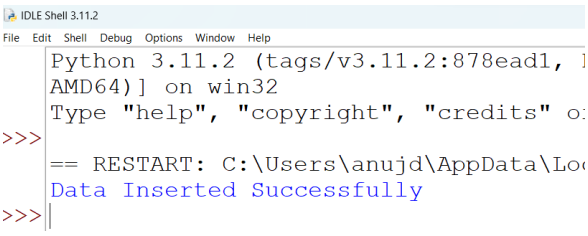
Result in Excel:



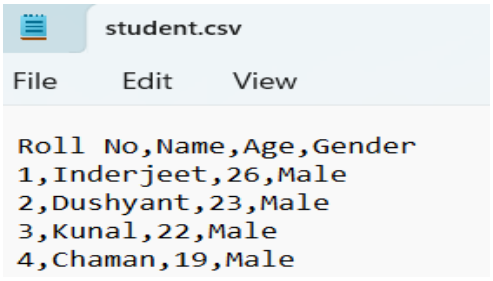
Code:

```
import csv
f=open("student.csv","w",newline="")
data=csv.writer(f)
data.writerow(["Roll No","Name","Age","Gender"])
data.writerows([[1,"Inderjeet",26,"Male"],
            [2,"Dushyant",23,"Male"],
            [3,"Kunal",22,"Male"],
            [4,"Chaman",19,"Male"]])
print("Data Inserted Successfully")
f.close()
```

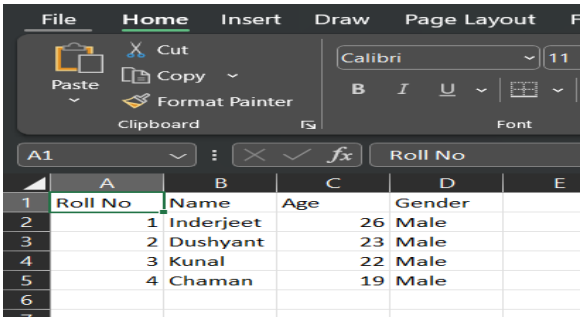
Output in IDLE:



Result in Notepad:



Result in Excel:



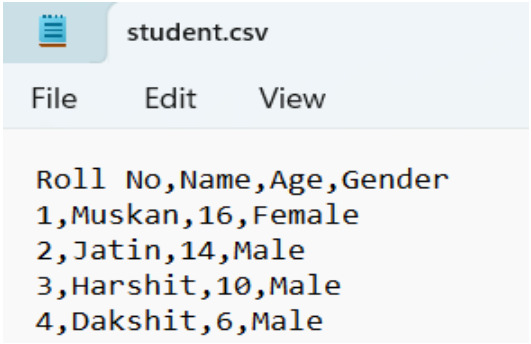
Code:

```
import csv
f=open("student.csv","w",newline="")
data=csv.writer(f)
data.writerow(["Roll No","Name","Age","Gender"])
ch="y"
details=[]
while ch in "yY":
    rno=int(input("Enter the Roll NO: "))
    name=input("Enter the Name : ")
    age=int(input("Enter the Age : "))
    gender=input("Enter the Gender : ")
    info=[rno,name,age,gender]
    details.append(info)
    ch=input("Press 'Y' if you want to continue: ")
data.writerows(details)
print("Data Inserted Successfully")
f.close()
```

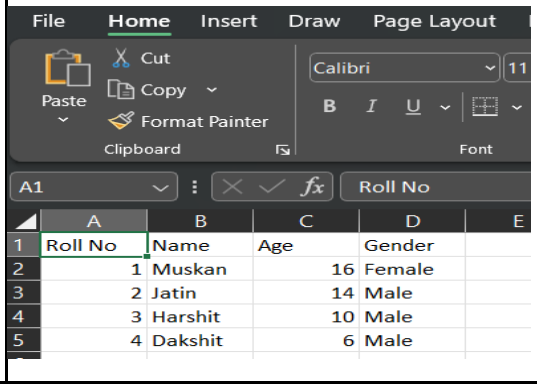
Output in IDLE:

```
== RESTART: C:/Users/anujd/AppData/Local/Pro
Enter the Roll NO: 1
Enter the Name : Muskan
Enter the Age : 16
Enter the Gender : Female
Press 'Y' if you want to continue: y
Enter the Roll NO: 2
Enter the Name : Jatin
Enter the Age : 14
Enter the Gender : Male
Press 'Y' if you want to continue: y
Enter the Roll NO: 3
Enter the Name : Harshit
Enter the Age : 10
Enter the Gender : Male
Press 'Y' if you want to continue: y
Enter the Roll NO: 4
Enter the Name : Dakshit
Enter the Age : 6
Enter the Gender : Male
Press 'Y' if you want to continue: n
Data Inserted Successfully
```

Result in Notepad:

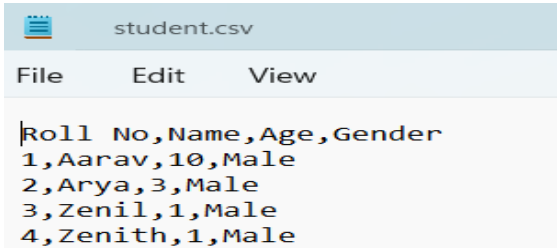


Result in Excel:



READING FROM CSV FILES

File Content:



Code: To read all records of csv file

```
import csv
f=open("student.csv","r")
data=csv.reader(f)
for i in data:
    print(i)
```

Output:

```
== RESTART: C:/Users/anujd/AppData/L
['Roll No', 'Name', 'Age', 'Gender']
['1', 'Aarav', '10', 'Male']
['2', 'Arya', '3', 'Male']
['3', 'Zenil', '1', 'Male']
['4', 'Zenith', '1', 'Male']
>>>
```

Code: To display all the records in which name starts with 'A'

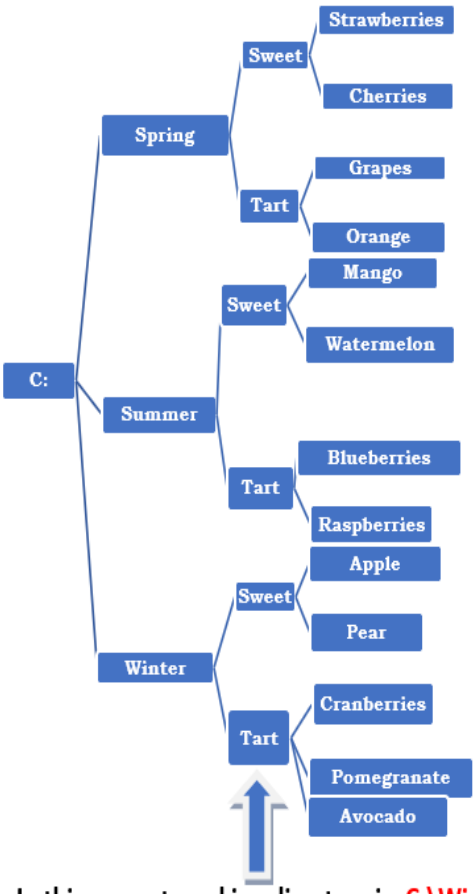
```
import csv
f=open("student.csv","r")
data=csv.reader(f)
for i in data:
    if i[1][0] in "aA":
        print(i)
```

Output:

```
== RESTART: C:/Users/anujd/A
['1', 'Aarav', '10', 'Male']
['2', 'Arya', '3', 'Male']
>>>
```

ABSOLUTE AND RELATIVE PATH:

1. Absolute path is a path that starts from the root directory of the file system or we can say that it describes how to access a given file or directory, from the starting of the file system.
2. Relative path is a path that is relative to the current working directory or we can say that is interpreted from the perspective of the current working directory.



Relative Path	Absolute Path
..\..\Spring\Sweet\Strawberries	C:\Spring\Sweet\Strawberries
..\..\Spring\Sweet	C:\Spring\Sweet
..\..\Spring\Sweet\Cherries	C:\Spring\Sweet\Cherries
..\..\Spring	C:\Spring
..\..\Spring\Tart\Grapes	C:\Spring\Tart\Grapes
..\..\Spring\Tart\	C:\Spring\Tart
..\..\Spring\Tart\Orange	C:\Spring\Tart\Orange
..\..\Spring\Sweet\Mango	C:\Summer\Sweet\Mango
..\..\Spring\Sweet	C:\Summer\Sweet
..\..\Spring\Sweet\Watermelon	C:\Summer\Sweet\Watermelon
..\..	C:
..\..\Summer	C:\Summer
..\..\Summer\Tart\Blueberries	C:\Summer\Tart\Blueberries
..\..\Summer\Tart	C:\Summer\Tart
..\..\Summer\Tart\Raspberries	C:\Summer\Tart\Raspberries
..\Sweet\Apple	C:\Winter\Sweet\Apple
..\Sweet	C:\Winter\Sweet
..\Sweet\Pear	C:\Winter\Sweet\Pear
..	C:\Winter
Cranberries	C:\Winter\Tart\Cranberries
.	C:\Winter\Tart
Pomegranate	C:\Winter\Tart\Pomegranate
Avocado	C:\Winter\Tart\Avocado

In this current working directory is: C:\Winter\Tart

Questions

1. Write a statement to send the file pointer position 10 bytes forward from current location of file ,consider fp as file object.
   a) fp.seek(10) b) fp.seek(10,1) c) fp.tell(10) d) fp.seek(1,10)

2. Which function is used to open a file in Python?
   a. open() b. read() c. write() d. close()

3. Which of the following mode in file opening statement results or generates an error if the file does not exist?
   a) a+ b) r+ c) w+ d) None of the above

4. If the csv file 'item.csv' has a header and 3 records in 3 rows and we want to display only the header after opening the file with open() function, we should write
   a). print(file.readlines())
   b). print(file.read())
   c) print(file.readline())
   d). print(file.header())

5. Identify the missing part in the code to write the list object in the file
   >>> import pickle
   >>> x=[1,3,5,7]
   >>> f=open('w.dat','wb')
   >>> pickle._____(x,f)
   >>> f.close()
   a). write() b). writeline() c). load() d). dump()

6. Which of the following statements are True
   (a) When you open a file for reading, if the file does not exist, an error occurs
   (b) When you open a file for writing, if the file does not exist, a new file is created
   (c) When you open a file for writing, if the file exists, the existing file is overwritten with the new file
   (d) All of the mentioned

7. Write a statement to send the file pointer position 10 bytes forward from current location of file , consider fp as file object.
   a) fp.seek(10) b) fp.seek(10,1) c) fp.tell(10) d) fp.seek(1,10)

8. myfile.txt is a file stored in the working directory. Total number of characters including spaces are 106 and there are 5 lines. What will be the output of len(file.readlines()) after the file was opened as file=open('myfile.txt')
   a). 4 b). 106 c). 5 d). 1

9. Which of the following option is not correct?
   a. if we try to read a text file that does not exist, an error occurs.
   b. if we try to read a text file that does not exist, the file gets created.

c. if we try to write on a text file that does not exist, no error occurs.
d. if we try to write on a text file that does not exist, the file gets Created.

10. A text file readme.txt is opened in Python. What type of data is stored in f?
```
>>> file=open('readme.txt')
>>> f=file.readlines()
```
a). String b). Tuple c). List d). None of the above

11. Find P and Q from the options while performing object serialization
```
>>> import pickle
>>> spc=open("yoyo.dat","wb")
>>> x=500
>>> pickle.dump(P,Q)
```
a). x,spc b). spc, x c). 'yoyo.dat',500 d). 'yoyo.dat','500'

12. A text file myfile0.txt has two lines of text, what will be stored in the variable ctr when the following code is executed?
```
>>> ctr=0
>>> spc=open("myfile0.txt")
>>> while(spc.readline()):
        ctr += 1
```
a). 0 b). 1 c). 2 d). 3

13. How many lines does the file myfile00.txt has after the code is executed?
```
>>> mylist=['India', '\nis', '\nmy', '\ncountry', '\nand', '\nI', '\nam', '\na', '\nproud', '\ncitizen']
>>> spc=open("myfile00.txt","w")
>>> spc.writelines(mylist)
```
a). 2 b). 10 c). 9 d). 1

14. Which of the following options can be used to read the first line of a text file Myfile.txt?
a. myfile = open('Myfile.txt'); myfile.read()
b. myfile = open('Myfile.txt','r'); myfile.read(n)
c. myfile = open('Myfile.txt'); myfile.readline()
d. myfile = open('Myfile.txt'); myfile.readlines()

15. Assume that the position of the file pointer is at the beginning of 3rd line in a text file. Which of the following option can be used to read all the remaining lines?
a). myfile.read() b). myfile.read(n) c). myfile.readline() d). myfile.readlines()

16. A text file student.txt is stored in the storage device. Identify the correct option out of the following options to open the file in read mode.
i. myfile = open('student.txt','rb')
ii. myfile = open('student.txt','w')
iii. myfile = open('student.txt','r')
iv. myfile = open('student.txt')

a). only i b). both i and iv c). both iii and iv d). both i and iii

17. Which of the following statement is incorrect in the context of binary files?
    a) Information is stored in the same format in which the information is held in memory.
    b) No character translation takes place
    c) Every line ends with a new line character
    d) pickle module is used for reading and writing

18. Which of the following options can be used to read the first line of a text file Myfile.txt?
    a) myfile = open('Myfile.txt'); myfile.read()
    b) myfile = open('Myfile.txt','r'); myfile.read(n)
    c) myfile = open('Myfile.txt'); myfile.readline()
    d) myfile = open('Myfile.txt'); myfile.readlines()

19. Which of the following statement is incorrect in the context of binary files?
    a) Information is stored in the same format in which the information is held in memory.
    b) No character translation takes place
    c) Every line ends with a new line character
    d) pickle module is used for reading and writing

20. Which of the following statement is true?
    a) pickling creates an object from a sequence of bytes
    b) pickling is used for object serialization
    c) pickling is used for object deserialization
    d) pickling is used to manage all types of files in Python

21. Syntax of seek function in Python is myfile.seek(offset, reference_point) where myfile is the file object. What is the default value of reference_point?
    a) 0 b) 1 c) 2 d) 3

22. Which of the following character acts as default delimiter in a csv file?
    a) (colon) : b) (hyphen) - c) (comma) , d) (vertical line) |

23. Syntax for opening a Student.csv file in write mode is myfile = open("Student.csv","w",newline=''). What is the importance of newline=''?
    a) A newline gets added to the file
    b) Empty string gets appended to the first line.
    c) Empty string gets appended to all lines.
    d) EOL translation is suppressed

24. What is the correct expansion of CSV files?
    a) Comma Separable Values
    b) Comma Separated Values
    c) Comma Split Values

d) Comma Separation Values

25. Which of the following is not a function / method of csv module in Python?
a) read() b) reader() c) writer() d) writerow()

26. Which of the following statement opens a binary file record.bin in write mode and writes data from a list
lst1 = [1,2,3,4] on the binary file?
a) with open('record.bin','wb') as myfile:
pickle.dump(lst1,myfile)
b)with open('record.bin','wb') as myfile:
pickle.dump(myfile,lst1)
c) with open('record.bin','wb+') as myfile:
pickle.dump(myfile,lst1)
d) with open('record.bin','ab') as myfile:
pickle.dump(myfile,lst1)

27. Which of the following functions changes the position of file pointer and returns its new position?
a) flush() b) tell() c) seek() d) offset()

28. Suppose content of 'Myfile.txt' is:

Twinkle twinkle little star
How I wonder what you are
Up above the world so high
Like a diamond in the sky

What will be the output of the following code?

```
myfile = open("Myfile.txt")
data = myfile.readlines()
print(len(data))
myfile.close()
```

a) 3 b) 4 c) 5 d) 6

29. Identify the missing part in the code to write the list object in the file
```
>>> import pickle
>>> x=[1,3,5,7]
>>> f=open('w.dat','wb')
>>> pickle._____(x,f)
>>> f.close()
```
a) write() b) writeline() c) load() d) dump()

30. What is the significance of the tell() method?
a) tells the path of file

b) tells the current position of the file pointer within the file
c) tells the end position within the file
d) checks the existence of a file at the desired location

31. If the csv file 'item.csv' has a header and 3 records in 3 rows and we want to display only the header after opening the file with open() function, we should write
a). print(file.readlines())
b). print(file.read())
c). print(file.readline())
d). print(file.header())

32. The syntax of seek() is :
file_object.seek(offset [, reference_point]) What is the default value of reference_point?
a) 0 b) 1 c) 2 d) 3

33. Assume the content of text file, 'student.txt' is:
Arjun Kumar
Ismail Khan
Joseph B
Hanika Kiran

What will be the data type of data_rec?
myfile = open("Myfile.txt")
data_rec = myfile.readlines()
myfile.close()

a) string b) list c) tuple d) dictionary

34. Mr. Manish Kumar is writing a program to create a CSV file "user.csv" which will contain user name and password for some entries. He has written the following code. As a programmer, help him to successfully execute the given task.

```
import _____ # Line 1
def addCsvFile(UserName,PassWord): # to write / add data into the CSV file
    f=open(' user.csv','_____') # Line 2

    #csv file reading code
def readCsvFile(): # to read data from CSV file with
    open(' user.csv','r') as newFile:
        newFileReader = csv._____(newFile) # Line 3
        for row in newFileReader:
            print (row[0],row[1])
            newFile._____ # Line 4
addCsvFile("Arjun","123@456")
addCsvFile("Arunima","aru@nima")
addCsvFile("Frieda","myname@FRD")
readCsvFile() #Line 5
```

a) Name the module he should import in Line 1.
b) In which mode, Manish should open the file to add data into the file
c) Fill in the blank in Line 3 to read the data from a csv file.
d) Fill in the blank in Line 4 to close the file.

e) Write the output he will obtain while executing Line 5.

35. Roshni of class 12 is writing a program in Python for her project work to create a CSV file "Teachers.csv" which will contain information for every teacher's identification Number , Name for some entries.She has written the following code.However, she is unable to figure out the correct statements in few lines of code,hence she has left them blank. Help her write the statements correctly for the missing parts in the code.

```
import _________                                          # Line 1
def addrec(Idno, Name): # to add record into the CSV file
    f=open("Teachers.csv", _________)                     # Line 2
    Filewriter = CSV.writer(f)
    Filewriter.writerow([Idno,name])
    f.close()
def readfile(): # to read the data from CSV file
    f=open("Teachers.csv", _________)                     # Line 3
    FileReader = CSV._________ (f)                        # Line 4
    for row in FileReader:
        print(row)
    f._________                                           # Line 5
```

36. Rohit, a student of class 12, is learning CSV File Module in Python. During the examination, he has been assigned an incomplete python code (shown below) to create a CSV File 'Student.csv' (content shown below). Help him in completing the code which creates the desired CSV File.

CSV File
1,AKSHAY,XII,A
2,ABHISHEK,XII,A
3,ARVIND,XII,A
4,RAVI,XII,A
5,ASHISH,XII,A

Help him complete the below code:

Incomplete Code

```
import _____                                                    #Statement-1
fh = open(_____, _____, newline='')                             #Statement-2
stuwriter = csv._____                                           #Statement-3
data = [ ]
header = ['ROLL_NO', 'NAME', 'CLASS', 'SECTION']
data.append(header)
for i in range(5):
        roll_no = int(input("Enter Roll Number : "))
        name = input("Enter Name : ")
        Class =  input("Enter Class : ")
        section =  input("Enter Section : ")
        rec = [ _____ ]                                         #Statement-4
        data.append(_____)                                      #Statement-5
stuwriter. _____ (data)                                         #Statement-6
fh.close()
```

37. Write the definition of a function Change Gender() in Python, which reads the contents of a text file "BIOPIC. TXT" and displays the content of the file with every occurrence of the word 'he replaced by 'she. For example, if the content of the file "BIOPIC. TXT" is as follows:
Last time he went to Agra, there was too much crowd, which he did not like. So this time he decided to visit some hill station.
The function should read the file content and display the output as follows:
Last time she went to Agra, there was too much crowd, which she did not like. So this time she decided to visit some hill station.

38. Ranjan Kumar of class 12 is writing a program to create a CSV file “user.csv” which will contain user name and password for some entries. He has written the following code. As a programmer, help him to successfully execute the given task.

```
import _____ # Line 1
def addCsvFile(UserName,PassWord): # to write / add data into the CSV file
   f=open(' user.csv','_____') # Line 2
   newFileWriter = csv.writer(f)
   newFileWriter.writerow([UserName,PassWord])
   f.close()
#csv file reading code
def readCsvFile(): # to read data from CSV file
   with open(' user.csv','r') as newFile:
      newFileReader = csv._____(newFile)# Line 3
      for row in newFileReader:
         print (row[0],row[1])
   newFile._____ # Line 4
addCsvFile(“Arjun”,”123@456”)
addCsvFile(“Arunima”,”aru@nima”)
addCsvFile(“Frieda”,”myname@FRD”)
readCsvFile() #Line 5
```

(a) Name the module he should import in Line 1.
(b) In which mode, Ranjan should open the file to add data into the file in line 2.
(c) Fill in the blank in Line 3 to read the data from a csv file.
(d) Fill in the blank in Line 4 to close the file.
(e) Write the output he will obtain while executing Line 5.

39. Which of the following modes is used for both writing and reading from a binary file?
a) wb+ b) w c) wb d) w+

40. The correct syntax of read() function from text files is:
a) file_object.read() b. file_object(read) c. read(file_object) d. file_object().read

41. What are the three modes in which a file can be opened in Python?
a) r, w, and a b. r+, w+, and a+ c. rb, wb, and ab d. All of the above

42. What mode should be used to open a file for reading only?
a). r b.) w c.) a d.) r+

43. What mode should be used to open a file for writing only?
a.) r b.) w c.) a d.) w+

44. What mode should be used to open a file for appending data?
a.) r b.) w c.) a d.) a+

45. Which mode is used for both reading and writing in a file?
a) . r+ b.) w+ c.) a+ d.) x+

46. Which method is used to read data from a file in Python?
a. read() b. write() c. seek() d. close()

47. Which method is used to write data to a file in Python?
a. read() b. write() c. seek() d. close()

48. Which method is used to move the file pointer to a specific position in a file in Python?
a. read() b. write() c. seek() d. close()

49. Which method is used to close a file in Python?
a. read() b. write() c. seek() d. close()

50. What is the default mode in which a file is opened in Python?
a. r b. w c. a d. None of the above

51. How do you read a specific number of bytes from a binary file in Python?
a. read() b. read_bytes() c. read(n) d. readlines(n)

52. What is the difference between a text file and a binary file?
a. Text files contain characters, while binary files contain arbitrary data.
b. Text files are human-readable, while binary files are not.
c. Text files are encoded using a character set, while binary files are not.
d. All of the above

53. What is the difference between the read() and readline() methods in Python?
a. The read() method reads all of the data from a file, while the readline() method reads only one line of data at a time.
b. The read() method returns a string, while the readline() method returns a list of strings.
c. The read() method does not move the file pointer, while the readline() method moves the file pointer to the next line.
d. All of the above

54. What is the difference between the write() and writelines() methods in Python?
a. The write() method writes a single string to a file, while the writelines() method writes a list of strings to a file.
b. The write() method does not append a newline character to the end of the string, while the writelines() method does.
c. The write() method moves the file pointer to the next position, while the writelines() method does not.
d. All of the above

55. What is the difference between the seek() and tell() methods in Python?
a. The seek() method moves the file pointer to a specific position in a file, while the tell() method returns the current position of the file pointer.
b. The seek() method takes an offset and a mode as arguments, while the tell() method does not.
c. The seek() method can move the file pointer to a position before the beginning of the file, while the tell() method cannot.
d. All of the above

56. What is the purpose of file handling in Python?
a. To store data permanently
b. To perform mathematical operations on files
c. To create graphical user interfaces
d. To handle network connections

57. What happens if you try to open a file in 'w' mode that already exists?
a. It overwrites the existing file.
b. It raises a FileNotFoundError.
c. It appends data to the existing file.
d. It creates a backup of the existing file.

58. What is the purpose of the 'with' statement in file handling?
a. It is used for creating new files.
b. It automatically closes the file when done.

c. It is used for reading binary files.
d. It is used for appending data to files.

59. What is the purpose of the 'tell()' method in file handling?
a. It tells you the size of the file.
b. It returns the current cursor position.
c. It tells you the number of lines in the file.
d. It returns the file's modification time.

60. What is the purpose of the 'os' module in file handling?
a. It provides functions for file operations.
b. It is used for performing mathematical operations.
c. It handles network connections.
d. It creates graphical user interfaces.

61. Which method is used to delete a file in Python?
a. os.delete() b. os.remove() c. file.delete() d. file.remove()

62. Which method is used to rename a file in Python?
a. os.delete() b. os.remove() c. file.delete() d. file.remove()

63. What happens if you try to open a non-existent file in Python using 'r' mode?
a. It raises a FileNotFoundError
b. It creates an empty file
c. It raises an IOError
d. It opens a file in read mode

64. How can you check if a file exists in Python before trying to open it?
a. Using the os module
b. Using the file.exists() method
c. Using the file.exist() method
d. Using the file.open() method

65. What is the difference between 'rb' and 'r' when opening a file in Python?
a. 'rb' opens the file in binary mode, 'r' in text mode
b. 'rb' reads the file backward, 'r' reads it forward
c. 'rb' is for reading, 'r' is for writing
d. 'rb' and 'r' are the same

66. Which character is used as the path separator in Python?
a. / b. \ c. : d. ,

67. What does CSV stand for?
a. Comma-Separated Values
b. Computer System Variables
c. Centralized Storage Values
d. Common String Values

68. Which module in Python is used to read and write CSV files?
    a. os
    b. csv
    c. fileio
    d. csvio

69. How can you read a CSV file in Python using the `csv` module?
    a. csv.read(file)
    b. csv.reader(file)
    c. csv.load(file)
    d. csv.load_csv(file)

70. What function is used to write data to a CSV file using the `csv` module in Python?
    a. csv.write()
    b. csv.writer()
    c. csv.write_csv()
    d. csv.write_to_file()

71. In a CSV file, how are values typically separated?
    a. By commas (`,`)
    b. By spaces (``)
    c. By semicolons (`;`)
    d. By tabs (`\t`)

72. How can you write a dictionary to a CSV file using the `csv` module in Python?
    a. csv.write_dict()
    b. csv.writerows()
    c. csv.writer_dict()
    d. csv.DictWriter()

73. Which of the following statements is true about reading a CSV file with the `csv` module in Python?
    a. Each row is returned as a list of strings
    b. Each row is returned as a dictionary
    c. Each column is returned as a separate value
    d. CSV files cannot be read using the `csv` module

74. Raghav is trying to write a tuple tup1 = (1,2,3,4,5) on a binary file test.bin. Consider the following code written by him.

```
import pickle
tup1 = (1,2,3,4,5)
myfile = open("test.bin",'wb')
pickle._______ #Statement 1 myfile.close()
```

    Write the missing code in Statement 1

75. A binary file employee.dat has following data

Empno	empname	Salary
101	Anuj	50000
102	Arijita	40000
103	Hanika	30000
104	Firoz	60000
105	Vijaylakshmi	40000

```
def display(eno):
        f=open("employee.dat","rb")
        totSum=0
        try:
            while True:
                R=pickle.load(f)
                if R[0]==eno:
                    _____ #Line1
                totSum=totSum+R[2]
         except:
            f.close()
            print(totSum)
```

When the above mentioned function, display (103) is executed, the output displayed is 190000. Write appropriate jump statement from the following to obtain the above output.

76. Suppose content of 'Myfile.txt' is:
Honesty is the best policy.
What will be the output of the following code?

```
myfile = open("Myfile.txt")
x = myfile.read()
print(len(x))
myfile.close()
```

77. Suppose content of 'Myfile.txt' is
Culture is the widening of the mind and of the spirit.
What will be the output of the following code?

```
myfile = open("Myfile.txt")
x = myfile.read()
y = x.count('the')
print(y)
myfile.close()
```

78. Suppose content of 'Myfile.txt' is

Humpty Dumpty sat on a wall
Humpty Dumpty had a great fall

All the king's horses and all the king's men
Couldn't put Humpty together again

What will be the output of the following code?

```
myfile = open("Myfile.txt")
 record = myfile.read().split()
 print(len(record))
 myfile.close()
```

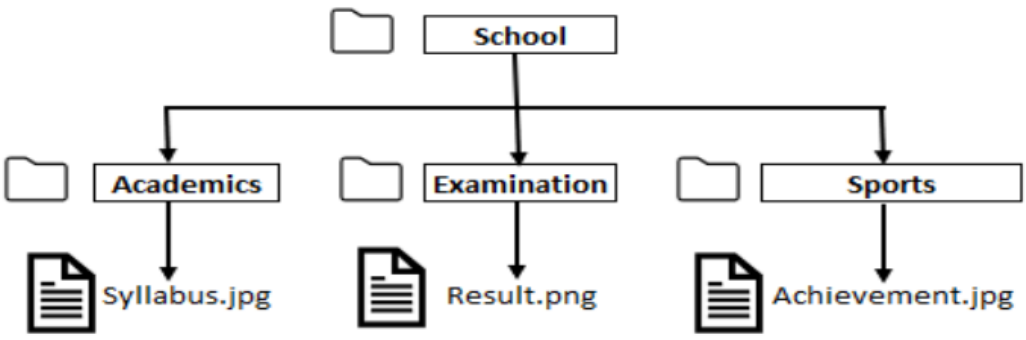
79. Suppose content of 'Myfile.txt' is

Ek Bharat Shreshtha Bharat

What will be the output of the following code?

```
myfile = open("Myfile.txt")
vlist = list("aeiouAEIOU")
vc=0
x = myfile.read()
for y in x:
    if(y in vlist):
        vc+=1
print(vc)
myfile.close()
```

80. Consider the following directory structure.



Suppose root directory (School) and present working directory are the same. What will be the absolute path of the file Syllabus.jpg?

a) School/syllabus.jpg
b) School/Academics/syllabus.jpg
c) School/Academics/../syllabus.jpg
d) School/Examination/syllabus.jpg

81. A binary file "STUDENT.DAT" has structure (admission_number, Name, Percentage). Write a function countrec() in Python that would read contents of the file "STUDENT.DAT" and display the details of those students whose percentage is above 75. Also display number of students scoring above 75%

82. A binary file "Book.dat" has structure [BookNo, Book_Name, Author, Price]. i. Write a user defined function CreateFile() to input data for a record and add to Book.dat . ii. Write

a function CountRec(Author) in Python which accepts the Author name as parameter and count and return number of books by the given Author are stored in the binary file "Book.dat"

83. Write a function in Python that counts the number of "Me" or "My" words present in a text file "STORY.TXT". If the "STORY.TXT" contents are as follows:
My first book was
Me and My Family.
It gave me chance to be
Known to the world.

The output of the function should be: Count of Me/My in file: 4

84. Write a function AMCount() in Python, which should read each character of a text file STORY.TXT, should count and display the occurance of alphabets A and M (including small cases a and m too).
Example: If the file content is as follows:

Updated information As simplified by official websites.
The EUCount() function should display the output as:
A or a:4
M or m :2

85. Write a function in python to count the number of lines in a text file 'STORY.TXT' which is starting with an alphabet 'A' .

86. Write a method/function DISPLAYWORDS() in python to read lines from a text file STORY.TXT, and display those words, which are less than 4 characters.

87. Write the definition of a function Count_Line() in Python, which should read each line of a text file "SHIVAJI.TXT" and count total number of lines present in text file. For example, if the content of the file "SHIVAJI.TXT" is as follows:
Shivaji was born in the family of Bhonsle.
He was devoted to his mother Jijabai.
India at that time was under Muslim rule.
The function should read the file content and display the output as follows:

Total number of lines: 3

88. A binary file "PLANTS.dat" has structure (ID, NAME, PRICE).

Write the definition of a function WRITEREC () in Python, to input data for records from the user and write them to the file PLANTS.dat.
Write the definition of a function SHOWHIGH () in Python, which reads the records of PLANTS. dat and displays those records for which the PRICE is more than 500.

89. Write the definition of a Python function named LongLines ( ) which reads the contents

of a text file named 'LINES.TXT' and displays those lines from the file which have at least 10 words in it. For example, if the content of 'LINES.TXT' is as follows:

Once upon a time, there was a woodcutter
He lived in a little house in a beautiful, green wood.
One day, he was merrily chopping some wood.
He saw a little girl skipping through the woods, whistling happily.
The girl was followed by a big gray wolf.

Then the function should display output as :

He lived in a little house in a beautiful, green wood.
He saw a little girl skipping through the woods, whistling happily.

90. Write a function count Words (in Python to count the words ending with a digit in a text file "Details.txt".
Example:
If the file content is as follows:

On seat2 VIP1 will sit and
On seat1 VVIP2 will be sitting

Output will be:

Number of words ending with a digit are 4

91. A binary file "PATIENTS.dat" has structure (PID, NAME, DISEASE).

Write the definition of a function countrec () in Python that would read contents of the file "PATIENTS.dat" and display the details of those patients who have the DISEASE as 'COVID-19'. The function should also display the total number of such patients whose DISEASE is 'COVID-19'.

92. Devender Singh, the Principal in GSV school, is developing a school teachers data using the csv module in Python. He has partially developed the code as follows leaving out statements about which he is not very confident. The code also contains error in certain statements. Help him in completing the code to read the code the desired CSV File named "Teachers.csv"
#CSV File Content
ENO, NAME, SUBJECT
E1, R.S. Hooda, Maths
E2, Sumit Beniwal, Chemistry
E3, Neetu Grover, Shorthand
E4, Jitender Meena, Social Science
E5, Samta Devi, Sanskrit
E6, Preeti Saini, Domestic Science
E7, Mayank, Computer Science

E8, Sanjiv Kala, Hindi

```
#incomplete Code With Errors
Import CSV                                          #Statement-1
With open(_____'_____'newline=' ') as File:         #Statement-2
    ER=csv.                                         #Statement-3
    for R in range(ER):                             #Statement-4
        If _____== 'Chemistry':                     #Statement-5
            print(____'_____)                       #Statement-6
```

Q1. Devender Singh gets an Error for the module name used in Statement 1. What should he write in place of CSV to import the correct module?

a) File b) csv c) csv d) pickle

Q2. Identify the missing code for blank spaces in the line marked as Statement-2 to open the mentioned file.

a.) "Teacher.csv","r" b) "Teacher.csv","w" c) "Teacher.csv","rb" d)"Teacher.csv","wb"

Q3 Choose the function name (with parameter) that should be used in the line marked as Statement-3 .

a) reader(File) b) readrows(File) c) writer(File) d)"writerows(File)

Q4. Devender Singh gets an Error in Statement-4. What should he write to correct the statement?

a) For R in ER: b) while R in range(ER): c) for R = ER: d) while R = = ER:

Q5. Identify the suitable code for blank space in statement-5 to match every row's 3rd property with "Maths".

a) ER[3] b) ER[2] c) R[2] d) R[3]

Q6. Identify the suitable code for blank space in Statement-6 to display every Employee's Name and corresponding Department?

a) ER[1],R[2] b) R[1], ER[2] c) R[1],R[2] d) ER[1],ER[2]

93. Shreyas is a programmer, who has recently been given a task to write a user defined function named write_bin() to create a binary file called Cust_file.dat containing customer information customer number (c_no), name (c_name), quantity (qty), price (price) and amount (amt) of each customer.

The function accepts customer number, name, quantity and price. Thereafter, it displays the message Quantity less than 10..... Cannot SAVE', if quantity entered is less than 10. Otherwise the function calculates amount as price quantity and then writes the record in the form of a list into the binary file.

```
import pickle
def write_bin():
  bin_file=_______ #Statement 1
  while True:
    c_no=int(input("enter customer number"))
    c_name=input("enter customer name")
    qty=int(input("enter qty"))
    price=int(input("enter price"))
    if ______  #Statement 2
      print("Quantity less than 10..Cannot SAVE")
    else:
      amt=price * qty
      c_detail=[c_no,c_name,qty,price,amt]
      ______ #Statement 3
      ans=input("Do you wish to enter more records y/n")
      if ans.lower()=='n':
        ________ #Statement 4
    ____________ #Statement 5
  _____________ #Statement 6
```

i) Write the correct statement to open a file 'Cust_file.dat' for writing the data of the customer.

(ii) Which statement should Shreyas fill in Statement 2 to check whether quantity is less than 10.

(iii) Which statement should Shreyas fill in Statement 3 to write data to the binary file and in Statement 4 to stop further processing if the user does not wish to enter more records.

(iv) What should Shreyas fill in Statement 5 to close the binary file named Cust_file.dat and in Statement 6 to call a function to write data in binary file?