

Q1. Which function is used to open a file? What are the different modes of opening a file? Explain each mode of file opening.

```
In [1]: # ans:-In Python, we use the open() method to open files.
# there are various modes of opening a file:-
# 1.Read :-Open a file for reading. (default)
# 2.Write:-Open a file for writing. Creates a new file if it does not exist or truncates the file if it exists.
# 3.Exculsive:-Open a file for exclusive creation. If the file already exists, the operation fails.
# 4.Append:-Open a file for appending at the end of the file without truncating it. Creates a new file if it does not exist.
# 5.Text:-Open in text mode. (default)
# 6.Binary:-Open in binary mode.
# 7.Update:-Open a file for updating (reading and writing)
```

Q2. Why close() function is used? Why is it important to close a file?

```
In [2]: # Ans:-The close() method closes an open file. You should always close your files, in some cases, due to buffering, changes made to a file may not s
```

Q3. Write a python program to create a text file. Write ‘I want to become a Data Scientist’ in that file. Then close the file. Open this file and read the content of the file

```
In [9]: vivek=open("file4.txt", 'w')
```

```
In [14]: vivek.write("i want to become a data scientist")
```

```
Out[14]: 33
```

```
In [15]: vivek.close()
```

```
In [16]: kumar=open("file4.txt", 'r')
```

```
In [17]: kumar.read()
```

```
Out[17]: 'i want to become a data scientisti want to become a data scientist'
```

Q4. Explain the following with python code: read(), readline() and readlines().

```
In [18]: # ans:-read():=The read() method returns the specified number of bytes from the file. Default is -1 which means the whole file.
#         readline():=The readline() method returns one line from the file. You can also specified how many bytes from the line to return, by using the
#         readlines():=The readlines() method returns a list containing each line in the file as a list item.
#         Use the hint parameter to limit the number of lines returned.
#         If the total number of bytes returned exceeds the specified number, no more lines are returned.
```

Q5. Explain why with statement is used with open(). What is the advantage of using with statement and open() together?

```
In [19]: # Ans:=The open() function to open a file. Unlike open() where you have to close the file with the close() method
#         , the with statement closes the file for you without you telling it to.
```

Q6. Explain the write() and writelines() functions. Give a suitable example.

```
In [20]: #ans:-write() function
#         The write() function will write the content in the file without adding any extra characters
```

```
In [26]: vivek=open("asus.txt", "w")
for i in range(3):
    name=input("enter your laptop brand name=")
    vivek.write(name)
    vivek.write("\n")

    vivek.close()
    print("data is written into the file")
```

```
enter your laptop brand name=asus
data is written into the file
enter your laptop brand name=hp
```

```
-----
ValueError                                Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_18332\64887489.py in <module>
      2 for i in range(3):
      3     name=input("enter your laptop brand name=")
----> 4     vivek.write(name)
      5     vivek.write("\n")
      6

ValueError: I/O operation on closed file.
```

```
In [27]: vivek.close()
```

```
In [ ]:
```