

# Python\_advance\_assignment\_18

Q1. Describe the differences between text and binary files in a single paragraph.

In [ ]:

Ans: The differences between Text Files **and** Binary Files are:

Text files are special subset of binary files that are used to store human readable characters **as** a rich text document **or** plain text document. Text files also store data **in** sequential **bytes** but bits **in** text file represents characters.

Binary files are those typical files that store data **in** the form of sequence of **bytes** grouped into eight bits **or** sometimes sixteen bits. These bits represent custom data **and** such files can store multiple types of data (images, audio, text, etc) under a single file.

Q2. What are some scenarios where using text files will be the better option? When would you like to use binary files instead of text files?

In [ ]:

Ans: Text files are less prone to get corrupted **as any** undesired change may just show up once the file **is** opened **and** then can easily be removed. Whereas Use binary files instead of text files **for** image data.

Q3. What are some of the issues with using binary operations to read and write a Python integer directly to disc?

In [ ]:

Ans: When we read **or** write a python integer using binary operations, Binary operations deal **with** raw data. One needs to identify how many **bytes** one would read **or** write.

Q4. Describe a benefit of using the with keyword instead of explicitly opening a file ?

In [ ]:

Ans: When a file **is** opened using the **with** keyword, **if** some exceptions occur after opening a file, **or** at the end of the file it automatically does the closing of the file. There by **not** leaving an file **in open** mode **and** there would no need to explicitly close a file.

Q5. Does Python have the trailing newline while reading a line of text? Does Python append a newline when you write a line of text?

In [ ]:

Ans: Yes, Python have the trailing newline **while** reading a line of text. When we write a newline has to be provided **in** python explicitly.

Q6. What file operations enable for random-access operation?

In [ ]:

Ans: The file operations enable **for** random-access operation are `seek()` **and** `tell()`

Q7. When do you think you'll use the struct package the most?

In [ ]:

Ans: The struct package **is** mostly used **while** converting a common python types into C language types.

Q8. When is pickling the best option?

In [ ]:

Ans: Pickling **is** best option **for** creating a new binary file using python.

Q9. When will it be best to use the shelve package?

In [ ]:

Ans: Shelve package **is** used to pickle data but treats the entire file **as** dictionary.

Q10. What is a special restriction when using the shelve package, as opposed to using other data dictionaries?

In [ ]:

Ans: Only string data **type** can be used **as** key **in** this special dictionary **object**, whereas **any** picklable Python **object** can be used **as** value.