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

Each line of text is terminated with a special character, which is the new line character (‘\n’).

While binary files are where there is no termination of line and data is stored only after the files are converted in binary.

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?

Text files are usually preferred as there is minor chances of data getting corrupted.

Binary files are used when the data includes (audio, video and text) in a single file.

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

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

It is guaranteed to close the file no matter how the nested block exits.

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?

Yes, but does not append automatically.

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

The file operations enable for random-access operation are : clearcache(), lazycache(), getline(file,x), updatecache(), getlines()

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

The struct package is used to convert data types in cases of strings and numbers and vice versa.

Q8. When is pickling the best option?

A process used when you are required to store a python object into a database/file or to transport data over the network.

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

Accessed by keys its best to use shelve when arbitrary python objects are needed to be pickled using a dictionary like API.

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

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