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

Ans1- Text files and binary files differ in how they store and represent data. Text files store data as plain text, typically using characters from a character encoding like UTF-8, and may include readable text, line breaks, and special characters. Binary files, on the other hand, store data as a sequence of binary values, which can represent any type of information, including text, images, or executable code.

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?

Ans2- Text files are better suited for scenarios where human readability and easy editing are important, such as configuration files, source code, or structured data like CSV files. Binary files are preferred when you need to preserve data integrity, handle complex data structures, or store non-textual data like images, audio, or binary formats specific to certain applications

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

Ans3- Reading and writing Python integers directly to disk using binary operations can lead to issues related to endianness (byte order) and platform compatibility. Different systems may have different byte orders, causing problems when reading binary data written on a system with a different endianness.

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

Ans4- Using the with keyword in Python when opening a file provides automatic resource management. It ensures that the file is properly closed after you finish working with it, even if an exception is raised during the process. This simplifies code and reduces the risk of resource leaks compared to explicitly opening and closing files.

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?

Ans5- Python includes the trailing newline character ('\n') when reading a line of text, but it doesn't append a newline automatically when writing a line. To add a newline when writing, you need to include it explicitly in the string you write to the file.

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

Ans6- File operations like seek() and tell() enable random-access operations on files. seek() allows you to move the file pointer to a specific position in the file, and tell() retrieves the current position of the file pointer.

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

Ans7- The struct package in Python is most commonly used for working with binary data when you need to pack and unpack structured data into/from binary formats. It is especially useful when dealing with binary file formats like those used in network protocols, serialization, or reading/writing data to hardware devices.

Q8. When is pickling the best option?

Ans8- Pickling is a good option when you need to serialize Python objects into a binary format for storage or transmission and later restore them to their original state. It is suitable for preserving complex data structures and custom Python classes.

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

Ans9- The shelve package is best used when you need to store and retrieve Python objects in a dictionary-like data structure with persistent storage. It is suitable for managing large datasets and complex data structures, offering a convenient way to save and retrieve Python objects, including custom classes, lists, and dictionaries, with automatic serialization.

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

Ans10- One special restriction when using the shelve package, as opposed to other data dictionaries, is that the keys used to access the data must be strings. In traditional dictionaries, keys can be of various types, including integers or tuples, but in a shelve database, keys must be strings. This limitation is due to the underlying database system used by shelve.