Q1. Text files store data as plain text, whereas binary files store data as binary code that can be interpreted by a computer. Text files are human-readable and can be edited with a text editor, whereas binary files cannot. Text files can be used to store data that can be easily interpreted by humans, such as configuration files or log files, while binary files are used to store data that needs to be interpreted by a computer, such as images or executable files.

Q2. Text files are a better option when you need to store and manipulate human-readable data, such as configuration files, HTML, or XML. Binary files are a better option when you need to store data that needs to be interpreted by a computer, such as images, audio files, or executable files.

Q3. One issue with using binary operations to read and write a Python integer directly to disc is that the byte order used by the computer that wrote the data may be different from the byte order used by the computer that reads the data. This can result in a byte-swapping error, which can cause the data to be read incorrectly.

Q4. The with keyword automatically handles closing the file after it's done being used, even if an exception is raised. This eliminates the need for manually closing the file, which can be error-prone and can lead to bugs.

Q5. When reading a line of text, Python includes the trailing newline character, if there is one. When writing a line of text, Python appends a newline character to the end of the line, unless you specify otherwise.

Q6. Random-access operations can be achieved using the seek() and tell() methods. The seek() method sets the file's current position, while the tell() method returns the current position.

Q7. The struct package is most commonly used when working with binary data, such as network packets or binary file formats. It allows you to convert between Python data types and C data types, which can be useful when working with binary data formats that use C data types.

Q8. Pickling is the best option when you need to serialize complex Python objects into a stream of bytes, and then deserialize them back into their original form. It can be used to save state between sessions, or to transmit data between processes or machines.

Q9. The shelve package is best used when you need to store Python objects in a dictionary-like format, and you need to be able to persistently store the data between sessions. It allows you to store and retrieve Python objects using keys, just like a dictionary.

Q10. A special restriction when using the shelve package is that the keys used to access the data must be strings. This is because the keys are used as filenames to store the data, and filenames must be strings in most file systems.