Q1. What is the purpose of the try statement?

A1. The try statement is used to enclose code that might raise an exception during execution. It allows the program to handle exceptions gracefully and continue running even if errors occur.

Q2. What are the two most popular try statement variations?

A2. The two most popular try statement variations are the try/except statement and the try/finally statement. The try/except statement allows the program to catch and handle specific exceptions that might be raised by the code in the try block. The try/finally statement allows the program to execute cleanup code regardless of whether an exception is raised or not.

Q3. What is the purpose of the raise statement?

A3. The raise statement is used to manually raise an exception in a program. It is typically used when the program encounters an error or unexpected condition that cannot be handled automatically, and needs to signal to the calling code that something has gone wrong.

Q4. What does the assert statement do, and what other statement is it like?

A4. The assert statement is used to check that a condition is true, and raise an AssertionError if it is not. It is used primarily for debugging and testing purposes, to ensure that assumptions made by the code are correct. The assert statement is similar to the if statement, but is intended for use in situations where a condition must be true in order for the program to continue executing.

Q5. What is the purpose of the with/as argument, and what other statement is it like?

A5. The with/as statement is used to create a context in which a resource is acquired, used, and automatically released when the context is exited, even if an exception is raised. It is used primarily for working with files, sockets, and other resources that need to be explicitly released when they are no longer needed. The with/as statement is similar to the try/finally statement, but provides a more concise and readable syntax for working with resources.