Q1. What is the purpose of the try statement?

The purpose of the try statement is to define a block of code in which exceptions can occur. It allows you to catch and handle exceptions gracefully, preventing them from causing program crashes and providing a way to recover from errors.

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

The two most popular try statement variations are: a) The except clause, which follows the try block and specifies the code to be executed if a specific exception is raised. b) The finally clause, which is used to define cleanup actions that should be executed regardless of whether an exception was raised or not.

Q3. What is the purpose of the raise statement?

The purpose of the raise statement is to manually raise an exception in Python. It allows you to trigger an exception with a specific type and message, enabling custom error handling and control flow.

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

The assert statement checks if a given expression is True. If it's False, it raises an AssertionError exception with an optional error message. It is used for debugging and ensuring that assumptions in your code are valid. It is similar in concept to an if statement, but it's primarily used for debugging and self-documentation.

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

The purpose of the with/as argument, used in the context manager pattern, is to simplify resource management, like opening and closing files or database connections. It ensures that setup and teardown actions are properly handled. It is similar in concept to a try and finally block, but it provides a more structured and readable way to manage resources.