Q1. What is the purpose of the try statement?

Try statement is used to trigger exception. If code in try block causes some run time error, then try will throw an exception. Exception is caught by except block.

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

IOError – It occurs on errors like a file fails to open. ImportError – If a python module can’t be loaded or located. KeyboardInterrupt : when an unrequired key is pressed by the user ValueError : when built-in function receives a wrong argument EOFError : if End-Of-File is hit without reading any data

How try() works?

First, try clause is executed i.e. the code between try and except clause. If there is no exception, then only try clause will run, except clause is finished. If any exception occured, try clause will be skipped and except clause will run. If any exception occurs, but the except clause within the code doesn’t handle it, it is passed on to the outer try statements. If the exception left unhandled, then the execution stops. A try statement can have more than one except clause

Q3. What is the purpose of the raise statement?

raise statement is used to raise an exception. You can define what kind of error to raise.

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

The assert statement lets you test if a condition in your code returns True, if not, the program will raise an AssertionError. It is like raise statement, where we can create our custom exceptions.

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

with statement in Python is used in exception handling to make the code cleaner and much more readable. It simplifies the management of common resources like file streams. The with statement itself ensures proper acquisition and release of resources. This allows common try…except…finally usage patterns to be encapsulated for convenient reuse.

when "as" is used after "with", it will reassign the the returned object to a new identifier.