Q1. What is the purpose of the try statement?

Ans : The try statement allows you to define a block of code to be tested for errors while it is being executed. The catch statement allows you to define a block of code to be executed, if an error occurs in the try block.

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

Ans : There are two other optional segments to a try block: else and finally . Both of these optional blocks will come after the try and the except . Also, there's nothing stopping you from using both else and finally in a single statement — but keep them in that order if you do.

Q3. What is the purpose of the raise statement?

Ans : The RAISE statement stops normal execution of a PL/SQL block or subprogram and transfers control to an exception handler. RAISE statements can raise predefined exceptions, such as ZERO\_DIVIDE or NO\_DATA\_FOUND , or user-defined exceptions whose names you decide.

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

Ans : The assert keyword is used when debugging code. The assert keyword lets you test if a condition in your code returns True, if not, the program will raise an AssertionError. You can write a message to be written if the code returns False, check the example below.

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

Ans : The with statement in Python helps you with resource management. It ensures no resources are accidentally left open. The with statement is a replacement for commonly used try/finally error-handling statements. A common example of using the with statement is opening a file.