Q1. What is the purpose of the try statement?

Ans: The try block lets you test a block of code for errors. The except block lets you handle the error. The else block lets you execute code when there is no error

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

Ans:

A.

try:

# Some Code

except:

# Executed if error in the try block

B.

try:

# Some Code

except:

# Executed if error in the

# try block

else:

# execute if no exception

Q3. What is the purpose of the raise statement?

Ans: **Python raise Keyword** is used to raise exceptions or errors. The raise keyword raises an error and stops the control flow of the program. It is used to bring up the current exception in an exception handler so that it can be handled further up the call stack.

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

Ans: Assertion is the boolean expression that checks if the statement is True or False. If the statement is true then it does nothing and continues the execution, but if the statement is False then it stops the execution of the program and throws an error.

**assert** keyword helps in achieving this task. This statement takes as input a boolean condition, which when returns true doesn’t do anything and continues the normal flow of execution, but if it is computed to be false, then it raises an AssertionError along with the optional message provided.

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.