**Q1. What is the purpose of the try statement?**

Ans- The ‘**try**’ statement is used in Python to handle exceptions or errors that may occur during program execution. It allows the programmer to write code that may potentially cause an exception, and then define how to handle that exception if it occurs. Additionally, the ‘try’ statement can handle multiple types of exceptions by including multiple ‘except’ blocks, and it can also raise exceptions using the ‘raise’ statement.

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

Ans- The two most popular variations of the ‘**try**’ statement in Python are:

1. **‘try-except**’ - This variation allows the programmer to specify one or more exceptions that may be raised by the code inside the ‘**try’** block. If any of the specified exceptions occur during execution, the interpreter will jump to the corresponding ‘**except’** block to handle the exception.
2. **‘try-finally’ -** This variation allows the programmer to specify a ‘**finally**’ block of code that will always be executed, regardless of whether an exception was raised or not. This is useful for performing cleanup operations, such as closing files or network connections, after a block of code has executed.

**Q3. What is the purpose of the raise statement?**

Ans- The ‘**raise**’ statement in Python is used to manually raise an exception in a program. It allows the programmer to specify a custom exception message and type, and to interrupt program execution if a critical error occurs.

Also, the ‘**raise’** statement can be used to re-raise an exception that was caught by a ‘**try-except’** block. This can be useful for debugging or logging purposes, as it allows the original exception to be raised again and handled by a higher-level exception handler.

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

Ans- The ‘**assert’** statement in Python is used to test whether a given condition is true, and raise an exception if it is not. It is often used to perform debugging checks, ensuring that certain assumptions or conditions hold true at specific points in the program.

So, the **assert** statement in Python is a useful tool for debugging and testing, allowing the programmer to check that certain conditions hold true at specific points in the program, and raising an exception if they do not.

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

Ans- The ‘**with/as**’ statement in Python is used to manage resources, such as files or network connections, that need to be cleaned up after use. It ensures that the resource is properly closed or released, even if an exception is raised during execution. The ‘**with/as**’ statement is often used as a safer and more convenient alternative to using ‘**try-finally’** blocks.