**Python advance assignment-6**

**Q1. Describe three applications for exception processing.**

* Exception handling can be used to handle unexpected errors or exceptional cases that may occur during the execution of a script, allowing the program to continue running instead of crashing.
* Exception handling can be used to validate user input and prevent invalid data from being processed.
* Exception handling can be used to handle network or file I/O errors, allowing the program to handle and recover from failures in these systems.

**Q2. What happens if you don't do something extra to treat an exception?**

If an exception is not handled, it will propagate up the call stack, potentially causing the program to crash or exit. This can leave the program in an unknown state and make it difficult to diagnose the cause of the error.

**Q3. What are your options for recovering from an exception in your script?**

There are several options for recovering from an exception in a script, including:

* Using a try-except block to catch the exception and handle it appropriately.
* Using a finally block to run cleanup code regardless of whether an exception occurred.
* Using an else block to run code that should only be executed if no exception occurred.
* Reraising the exception after handling it to allow it to be caught by a higher level in the call stack.

**Q4. Describe two methods for triggering exceptions in your script.**

* Using the **raise** statement to raise a specific exception.
* Using the **assert** statement to raise an **AssertionError** if a specified condition is not met.

**Q5. Identify two methods for specifying actions to be taken when an exception is caught.**

* Using a try-except block to catch the exception and handle it within the block.
* Using a catch block in a try-catch-finally block to catch the exception and handle it within the catch block.