Assignment 8

# Q1

The two latest user-defined exception constraints in Python 3.X are:

The \_\_cause\_\_ attribute: This attribute stores the exception that caused the current exception to be raised.

The \_\_context\_\_ attribute: This attribute stores a dictionary of context information about the exception.

# Q2

Class-based exceptions that have been raised are matched to handlers based on their class hierarchy. When an exception is raised, Python starts searching for an appropriate exception handler by examining the inheritance chain of the exception class. It looks for an except block that matches the exception class or one of its base classes. If a match is found, the corresponding except block is executed to handle the exception. If no matching handler is found, the exception is propagated up the call stack until it is caught or the program terminates.

# Q3

Adding custom attributes to the exception object: we can define custom attributes in our exception class to store additional context information related to the exception. These attributes can be accessed within the exception handler to provide more detailed information about the error condition.

Using the args attribute of the exception object: The args attribute of the exception object is a tuple that can be used to store additional arguments or values associated with the exception. By convention, the first item in the args tuple is often used to store the error message.

# Q4

There are two methods for specifying the text of an exception object's error message:

The \_\_str\_\_ method: This method is called when the exception object is converted to a string.

The \_\_repr\_\_ method: This method is called when the exception object is printed.

The \_\_str\_\_ method is the preferred method for specifying the text of an exception object's error message, as it is more user-friendly. The \_\_repr\_\_ method is only used for debugging purposes.

# Q5

String-based exceptions are no longer used in Python because they are not as flexible or as powerful as class-based exceptions. Class-based exceptions allow us to store more information about the exception, and they allow us to define custom exception handlers.