ASSIGNMENT - 8

Q1. What are the two latest user-defined exception constraints in Python 3.X?

Ans: Two latest user-defined exception constraints in Python 3.x:

* \_\_cause\_\_ attribute: It provides a reference to another exception that caused the current exception.
* \_\_context\_\_ attribute: This points to the context related to the exception, enabling exceptions raised in an except block to have access to the original exception context.

Q2. How are class-based exceptions that have been raised matched to handlers?

Ans: When a class-based exception is raised, Python searches for an appropriate except block that matches the exception's class or one of its base classes. The matching process follows the method resolution order (MRO) to find the first matching except block.

Q3. Describe two methods for attaching context information to exception artefacts.

Ans: Methods for attaching context information to exception artifacts:

* Using the from keyword when raising a new exception allows attaching context from another exception. For example, raise NewException() from original\_exception.
* Customizing exception classes by defining attributes to store specific context information.

Q4. Describe two methods for specifying the text of an exception object&#39;s error message.

Ans: Methods for specifying the text of an exception object's error message:

* Overriding the \_\_str\_\_ method in a custom exception class allows specifying the error message when the exception is converted to a string using str(exception\_instance).
* Overriding the \_\_repr\_\_ method in a custom exception class helps specify a detailed and more informative representation of the exception when using repr(exception\_instance).

Q5. Why do you no longer use string-based exceptions?

Ans: Reasons for no longer using string-based exceptions:

* String-based exceptions lack structured information and context, making them less informative and harder to handle programmatically.
* Class-based exceptions provide better organization and allow for attaching additional data or context, aiding in debugging and error resolution.
* Using classes for exceptions provides a structured approach, facilitating better exception handling and allowing for more precise error identification and recovery.