Q1. Describe three applications for exception processing.

Answer: Exception handling is the process of responding to unwanted or unexpected events when a computer program runs. Exception handling deals with these events to avoid the program or system crashing, and without this process, exceptions would disrupt the normal operation of a program. There are mainly three kinds of distinguishable errors in Python: syntax errors, exceptions and logical errors.

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

Answer:

When an exception occurred, if you don't handle it, the program terminates abruptly and the code past the line that caused the exception will not get executed.

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

Answer;

n Python, we catch exceptions and handle them using try and except code blocks. The try clause contains the code that can raise an exception, while the except clause contains the code lines that handle the exception.

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

Exception Cause of Error EOFError Raised when the input() function hits end-of-file condition. FloatingPointError Raised when a floating point operation fails. GeneratorExit Raise when a generator's close() method is called. ImportError Raised when the imported module is not found.

Q5. Identify two methods for specifying actions to be executed at termination time, regardless of whether or not an exception exists.

Answer:

KeyError Raised when a key is not found in a dictionary. KeyboardInterrupt Raised when the user hits the interrupt key (Ctrl+C or Delete). MemoryError Raised when an operation runs out of memory. NameError Raised when a variable is not found in local or global scope.

Double-click (or enter) to edit