Q1. Describe three applications for exception processing.

A single try statement can have multiple except statements.

You can also provide a generic except clause, which handles any exception.

After the except clause(s), you can include an else-clause.

The else-block is a good place for code that does not need the try: block's protection

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

An exception object is created when a Python script raises an exception. If the script explicitly doesn't handle the exception, the program will be forced to terminate abruptly.

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

Any function can be assigned to a script as a recovery function. Once set, if there is an error during execution, the recovery function will be called before the script stops.

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

As a Python developer you can choose to throw an exception if a condition occurs. To throw (or raise) an exception, use the raise keyword.

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

If the type of exception doesn't match any of the except blocks, it will remain unhandled and the program will terminate. The rest of the statements after the except block will continue to be executed, regardless if the exception is encountered or not.