List of standard exceptions

Exception	Description
ArithmeticError	Base class for all errors that occur for numeric calculation.
AssertionError	Raised in case of failure of the assert statement.
AttributeError	Raised in case of failure of attribute reference or assignment.
EnvironmentError	Base class for all exceptions that occur outside the Python environment.
EOFError	Raised when there is no input from input() function and the end of file is reached.
Exception	Base class for all exceptions.
FloatingPointError	Raised when a floating-point calculation fails.
ImportError	Raised when an import statement fails.
IndentationError	Raised when indentation is not specified properly.
IndexError	Raised when an index is not found in a sequence.
IOError	Raised when an input/output operation fails, such as the *print* statement or the *open()* function when trying to open a file that does not exist.
KeyboardInterrupt	Raised when the user interrupts program execution, usually by pressing Ctrl + C.
KeyError	Raised when the specified key is not found in the dictionary.
LookupError	Base class for all lookup errors.
NameError	Raised when an identifier is not found in the local or global namespace.
NotImplementedError	Raised when an abstract method that needs to be implemented in an inherited class is not implemented.
OSError	Raised for operating system related errors.
OverflowError	Raised when a calculation exceeds the maximum limit for a numeric type.
RuntimeError	Raised when a generated error does not fall into any category.
StandardError	Base class for all built-in exceptions except *StopIteration* and *SystemExit*.
StopIteration	Raised when the *next()* method of an iterator does not point to any object.
SyntaxError	Raised when there is an error in Python syntax.
SystemError	Raised when the interpreter finds an internal error, but the situation does not look so serious to cause it to abandon all hope.
SystemExit	Raised when the Python interpreter is quit by using the *sys.exit()* function. If not handled in the code, it causes the interpreter to exit.

TypeError	Raised when an operation or function is attempted that is invalid for the specified data type.
UnboundLocalError	Raised when trying to access a local variable in a function or method but no value has been assigned to it.
ValueError	Raised when the built-in function for a data type has the valid type of arguments, but the arguments have invalid values specified.
ZeroDivisonError	Raised when a division or modulo division by zero takes place for all numeric types.