Python Exceptions+IO Practicals

1) WAP (Write a program) to test raising these errors without exceptional handling:
    a) NameError (when you use a variable name that is not accessible)
    b) ValueError (when the value contains invalid data, for ex: int('abc')
    c) TypeError (when the type expected is incorrect, for ex: '2' + 2)
    d) IndexError (when you access an index that does not exist in a list or string)
    e) IOError (when io operations fail)

    Read the traceback and verify if you understand it.

2) Modify the above program to try..except and check whether the program will crash. Have a generic except Exception as e & print str(e). Add a specific catch for an error and check if that is executed. Put the generic except first and then the specific one and check which one gets executed.
3) Add else: & finally: blocks and verify if your understanding of when they are executed.
4) Try writing code that results in an error in else: & finally: and check its outcome.
5) Embed a nested try..except in a try..except. Catch ValueError in outer try & TypeError in inner try. Now have a statement fail first with TypeError and check which except: block gets executed. Change the statement to raise ValueError and check which except:block will get executed.
6) Create functions f1 & f2. In f1, invoke f2(). Have try..except in both functions. Catch specific errors TypeError & ValueError for ex in the functions. Add finally: blocks to both as well. In try, use return to return a value in both of them. Now verify the control flow working and double check your understanding of execution of finally: blocks.
7) Create a file named actors.txt. Add a name of an actor per line in the file.
    a) WAP to list all the actor names from the file and print it to the monitor.
    b) WAP to sort the actor names and print it to monitor.
    c) WAP to display only the unique names in the file and print them to the monitor.

8) WAP to build the following menu based app:

    Press 1 to add an Actor
    Press 2 to list the Actors
    Press 3 to delete an Actor
    Press 4 to exit

    When user chooses the option, reuse the code (build as functions) built in 7th problem to implement the given functionality.

9) Handle exceptions correctly in both 7th & 8th problems.

10) Read the documentation of List & Dictionary to verify which methods throw errors. WAP to test catching those errors to ensure the program doesn't crash.