Training Exercise - 3

Exception Handling, Modules, File Management and Python Modules / Standard Libraries

1. Generate Atleast 5 different Errors.

2. Handle all the 5 different Erros using try-except.

3. Handle an error with try-except-else.

4. Handle an error with try-except-else-finally.

5. Use raise for generating User Defined Exception for minimum length of a list should be 5.

6. Create a file 'mod.py' with a class with multiple methods and few member variables. Also create an individual methods outside the class as well. Create another file 'test.py' and without executing the 'mod.py' get it executed using the 'test.py' file.

7. Using the above mentioned files 'test.py' and 'mod.py'. In 'test.py' create an object of the class defined in the 'mod.py'. Call the methods using the object. Make sure only that class is accessible and not the individual method to execute.

8. Using the files 'test.py' and 'mod.py' call the individual method which is defined outside the class in 'mod.py' and call in it test.py. Make sure only that method from the file 'mod.py' is accessible in 'test.py'.

9. Using the files execute both class methods and individual method outside the class.

10. Create a script/program to open a file to write a string. Write a string in a file 'test_file.txt'.

11. Create a script/program to open a file 'test_file.txt' to read a string. Read the whole string content from the file and print it.

12. Create a script/program to open a file 'test_file.txt' to read the content line by line and print it.

13. Create a script/program to open a file 'test_file.txt' to append a string at the end of the existing string in a file.

14. Create a script/program to write and read binary data in a file 'test_file.data'.

15. Using pickle dump no of variables with different data types in a file 'my_variables.data'.

16. Create another script/program and read the dumped variables in the file 'my_variables.data'.

17. Print the current date using datetime and date libraries.

18. Convert a datetime to a string.

19. Get the difference between two dates in days.

20. Calculate your age in years, months and days.

21. Get the date which is 1 week from today's date.

22. Get the date which is 1 year from today's date.

23. Get the date which is 1 month from today's date.

24. Get the 1st day of the current month from today's date.

25. Get the 1st month of the current year from today's date.

26. Get the dates of current month starting from Monday to Sunday in a list.

27. Get the first date and last date of the current month.

28. Get me the 1st and last date of the current month in the format as following. '14th June 2016 Tuesday 10:00:00 AM'

29. Get me random number from 1 to 100.

30.Get me a random combination of 4 different numbers between 1 to 100.

31.You have a sorted list from 1 to 10 you have to unsort it.

32.Execute a shell script command from python code.

33.Create a regular expression to check a valid URL.

34. Create a class Student and add member variables with False values. The variables are as listed below. Marks will have a default value blank list.
    1. Name
    2. Reg No
    3. Roll No
    4. Standard
    5. Admission Year
    6. Marks
    7. Result

    Add a constructor that will assign Name, Reg No, Roll No, Standard, Admission year. In the constructor add validation for Name to be only Alphabetic, Reg No to be alphanumeric, Roll No, Standard nad Admission year to be numeric. All abobve values will be accepted as string only. If the passed parameters are not string raise and Error to the user.

    Add a method that will accept a dictionary for marks containing subject as key and marks as values. It will add the dictionary to the list of marks. Marks list will have multiple elements and each element will be a dictionary only. Here there should be a validation to acccept the marks which are less than or equal to 100 only. If the obtained marks are less than 40 the result willl be fail otherwise pass. In the dictionary the result can be added.

    Add another method that will generate the result. This method will check if there is any line in the marks having fail as result the result will be Fail and it will print the complete result as following.

    If any of the result is Fail the Percentage will not be shown and only -- will be printed instead.

    Define a function to calculate the grade. For Example if the result is fail for any subject it will be F grade. If the Percentage is 95+ it will be O+. If the percentage is 90+ and less than 95 then it will be O. If the percentage is 85+ and less than 90 it will be A+. If the Percentage is 80+ and less than 85 it will be A. If the Percentage is 75+ and less than 80 it will be a B+. If the percentgae is 70+ and less than 75 it will be B. If the Percentage is 60+ and less than 70 it will be C. If the percentage is 50+ and

less than 60 it will be D. If the percetage is 40+ and less than 50 it will be E. The Grade must be displayed on the result which is shown below.

****

Name : <Student Name>
Roll No : <Roll No> Standard: <Standard>

****

Subject	Total Marks	Passing Marks	Obtained Marks	Result
<Sub 1>	100	40	<obtained marks>	<result>
<Sub 1>	100	40	<obtained marks>	<result>
<Sub 1>	100	40	<obtained marks>	<result>

****

TOTAL <total> <total> <total>

Result: PASS / FAIL Percentage: <percentage>

Grade : <Obtained Grade>