Day.9

Introduction to Exception Handling
The try-catch blocks and flow of programs
The finally block
Throwing an exception
The throws clause
Rethrowing an exception
Checked and Unchecked exceptions
User defined exceptions

Q1) Make a Calculator program for add, subtract, multiply and divide. Take input from the user. Handle the appropriate exceptions.

Hint: Use InputMismatchException, Arithmetic Exception

Q2) A comma separated list containing pairs of topic name, time in hours (Java 14, JEE 10, JME 12) will be entered in the command line arguments. If a day consists of 8 hours, list out the topics that will be covered day-wise. Catch all the possible exceptions.

Q3) In Q1, modify the program such that the exceptions are not handled in the add, diff, mul and div methods. The exception handling should be delegated to the caller method.

Q4) Create a class called Employee that asks the user to input the name and the age of an employee. Raise a custom defined exception when the user enters an employee name that has already been entered and raise another exception if the age is negative or less than 18 or greater than 60. If there is any occurrence of InputMismatchException and NumberFormatException, throw those also as user defined exceptions.

Hint: use exception wrapping

Q5) Create a class MyCalculator which consists of a single method power(int,int). This method takes two integers,n and ,p as parameters and finds n power p. If either n or p is negative, then the method must throw an exception (IllegalArgumentException) which says "n and p should be non-negative".

- For the above program, first use try catch to catch generic exception
- Second, replace he generic exception to specific exception

Q6) Write a program to,

i. Create the situation when the below exceptions happen, and let the program throw these exceptions.

   1. ArithmeticException, ArrayIndexOutOfBoundsException, ClassCastException, IllegalArgumentException, IndexOutOfBoundsException, NullPointerException

ii. Handle these exception by try and catch methods in a chain of exception hierarchy.

iii. For all the exceptions get the message from the exception and print them

iv. For all the exceptions get the print the stack trace

Q7) Write a program to generate OutofMemory error, and see if any files are newly created by JVM for analysing this case.