Object-Oriented Programming Lab#10, Fall 23

Today's Topics

- Exception
- Thread

Problems/Assignments – Online Reservation System

Problem 1: Write a java program where a method will throw exception upon different conditions and handle those exception upon calling the method.

1. Define a class "**TestException**" and add two methods: the **main** method and a static method name **checkDataForStudent.** The **checkDataForStudent** will take 2 parameters: a float to represent the percentage of attendance and a boolean to represent whether a student has attended the final exam or not.
    a. Inside the **checkDataForStudent** method, implement the following logic.

    - If attendance is less than 70 (percent), throw an Exception with message saying "Attendance should be 70 or more to be eligible for the final exam.
    - If 2nd parameter (the boolean parameter), hasAttendedFinalExam, is false print "Failed"
    - If the above 2 check fail, print "Eligible for Final Exam"

    b. Inside the main method, take attendance percentage and final exam status for 10 students and call the **checkDataForStudent** for those students. If any exception is thrown, print the message and go for the next student.

Problem 2: Write a java program where a method will throw user defineed exceptions and handle those exception upon calling the method.

1. Create a user-defined exception **LowAttendanceException** will take a float type parameter **minAttendance** and set the exception message to ""Attendance percentage should be **minAttendance** or more to sit for the final exam"., here **minAttendance** is the parameter passed to the constructor.
2. Update the **checkDataForStudent** method of problem 1 as below.
    a. If attendance is less than 70, throw **LowAttendanceException** and pass 70 as **minAttendance**.
    b. If **hasAttendedFinalExam** is false, throw Exception saying "Failed due to absence in Final exam.".
3. Update the main method of problem 1. Handle both exceptions separately.

Problem 3: Write a multi-threaded java program where 2 threads will run parallelly. One thread will print even numbers between a range and other thread will print odd numbers between a range.

1. Create a class **Job** implementing the **Runnable** interface. Add a boolean attribute **isEven,** and 2 other attributes of int type**: min,** and **max (**here **min** and **max** are the minimum and maximum value of the range). Create a constructor and pass parameter for all 3 attributes and initialize accordingly. Override the **run** method. Inside the **run** method, print even numbers between min and max if **isEven** is **true**; print odd numbers if it is false.

2. Create a class "TestThread" and add main method. Inside the main method, create 2 threads: one for printing even numbers and other for odd numbers. Take the inputs for min and max for both threads separately. Now run the 2 threads by calling start method.