

CST8132 Object Oriented Programming

Exercise 6 – Exceptions

Due in the lab period of week 11

Steps:

- Create a new project in Eclipse
- Add three classes:
 1. FullNameException
 2. Person
 3. PersonTester

Class Details:

1. Class FullNameException
 - This should inherit from class java.lang.Exception
 - It should contain 4 overloaded constructors:
 - No Parameter: use the super class constructor to set the message to “Full name not correct format”
 - String Parameter: use the super class constructor to set the message passed in via the parameter
 - String and Throwable parameters: use the super class constructor to set the message and throwable passed in via the parameters
 - Throwable Parameter: use the super class constructor to set the throwable that was passed via the parameter
2. Class Person
 - One field: fullName as String
 - A getter and setter for fullName
 - A constructor that sets fullName to “unknown”
 - A **private method** called validateData that **takes one parameter a String**
 - Inside the validateData method throw a FullNameException with an appropriate message for the following rules:
 - The String cannot be null
 - The String cannot have a length of zero
 - The String cannot consist of only white space
 - Tip: think about using trim() and length() on your String object
 - The String cannot have a length exceeding 20 characters
 - The String cannot contain a comma
 - Tip: there might be a method within String you can use, check the API
 - “might” meaning at least two methods that can be used**
 - You will need to add throws FullNameException to two of your methods.
3. PersonTester
 - Use the code provided to test your other classes.

**** Bonus of 2 marks** if you figure out two methods of class String that can locate a comma, and demonstrate them in your program.

Person Tester:

```
public class PersonTester {  
  
    public static void main(String[] args){  
  
        try{  
            Person p = new Person(); // name set to "Unknown" by default  
            // as you test each call, which triggers an exception,  
            // comment it out and re-run the program to test the  
            // next case.  
            p.setFullName(null); // null  
            p.setFullName(""); // empty String  
            p.setFullName(" "); // white space  
            p.setFullName("12345678901234567890tuna"); // length > 20  
            p.setFullName("fish, tuna"); // contains comma  
        }  
        catch(FullNameException ex){  
            System.out.println(ex.getMessage());  
        }  
    }  
}
```

Grading Guide (Total Score 10):

FullNameException coded correctly?	3
Person class has requested field, methods and constructor?	3
Person class members are marked with throws FullNameException where appropriate?	2
Program produces correct output	2
Bonus?	2
Expected Total	10

Sample Program Output

Test run 1: null

Full name cannot be empty or null

Test run 2: empty String

Full name cannot be empty or null

Test run 3: whitespace only

Full name cannot be empty or null

Test run 4: length > 20

Full name cannot exceed 50 characters

Test run 5: containing comma

Full name cannot contain a comma