Rochester Institute of Technology
National Technical Institute for the Deaf
Information and Computing Studies Department

Name: Edward Riley

NACA.161 Programming Fundamentals II
In-class Exercise #22 – Methods and Exceptions

Objective

Using Methods in the class with the main method
Using Exceptions

Write a program to read in data and catch exceptions

1) Write a program named CalcPay.java to calculate the pay of an employee. Write your entire program in the main. Prompt the user to enter the number of hours as a double and the hourly rate as a double. The pay is the hourly rate times the number of hours. Assume there is no overtime pay.
2) You **must** handle all possible exceptions from the input of the hours and hourly rate.
   NOTE: You **must** have a try-catch block for the hours and another try-catch block for the hourly rate. Catch any specific exceptions that might occur and the generic exception as the "default" case. For each exception, print a message that indicates the type of error using the methods in the Exception class. The "default" exception can print a message that an unexpected exception occurred.
3) For the number of hours, if the user's input causes an exception, print an error message and allow the user to re-enter the number of hours. The user must enter a valid numeric value to end the loop. The value of numeric number may not make logical sense. For example, if the user enters a value of -10.5 as the number of hours, the program will consider this input as valid, although is not a logical value for the number of hours.
4) Do the same for the hourly rate.
5) After both inputs have been entered, calculate and print the pay.
6) Put as few statements as possible inside the try blocks.
7) Run the program until it works as required.
8) Ask your instructor to check your program and sign below.

Instructor's Signature: _____

Creating a method

9) Write a method named getDouble that will read in a valid double number. The method has one parameter which is the Scanner object. The method returns a double value. The method must contain a try-catch block. If the user enters an invalid value allow the user to re-enter the number. Since the prompt is not passed into the method, just print a "?" each time input is requested.
10) Modify your main method to use the **getDouble** method to read the double value for the hours and the hourly rate.
11) Ask your instructor to check your program and sign below.

Instructor's Signature: _____

Modifying your main method

12) Modify your main method to validate the input values. The number of hours must be greater than or equal to 0 and less than or equal to 60.0. The hourly rate must be greater than or equal to 5.15 and less than or equal to 25.00. Do not change the **getDouble** method.

Creating another method

13) Create another method named **getInteger** that will read in a valid integer number. . The method has one parameter which is the Scanner object. The method returns an integer value. The method must contain a try-catch block.

Modifying your main method

14) Modify your main method to read in the number of weeks that the employee will be paid. The pay is now the number of weeks times the hours per week times the hourly rate.

When you complete all of the steps successfully and answer all of the questions, contact your instructor to check if your application(s) executes correctly and to review your code. We will initial the line below.

_____**Successful execution of code**

If you do not finish the program during the class period, contact your instructor to check to review your code and initial below.

_____**Code not completed during lab time**

You may then submit your work at the start of next class. You may not use the work period of the next class to complete this assignment. If you do not have a signature, then you cannot receive any points for this assignment.