Langara
THE COLLEGE OF HIGHER LEARNING.

Department of Computing Science & Information Systems
CPSC 1181
Lab#9
November 5, 2019

Objectives:

Study GUI, Event handling, and Exception

Preparation:

Study chapters layout management, Event Handling, and Exception Handling

Due date:

Due Date: 11:00 PM on Monday November 11, 2019

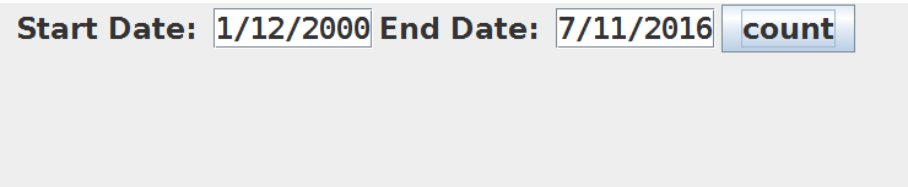
Where to upload:

zip folder to yourStudentID.zip and upload it to Lab9 in D2L.

Part 1: Tutorail (due date: 5:30 PM)

Part2: GUI Application

Write a Java Graphical User Interface (GUI) application program to obtain from a user two strings that should represent two valid dates. The dates are verified both for their correct format and for their validity in a calendar. Your program should have a GUI with two labels, and a button. Here is an example:



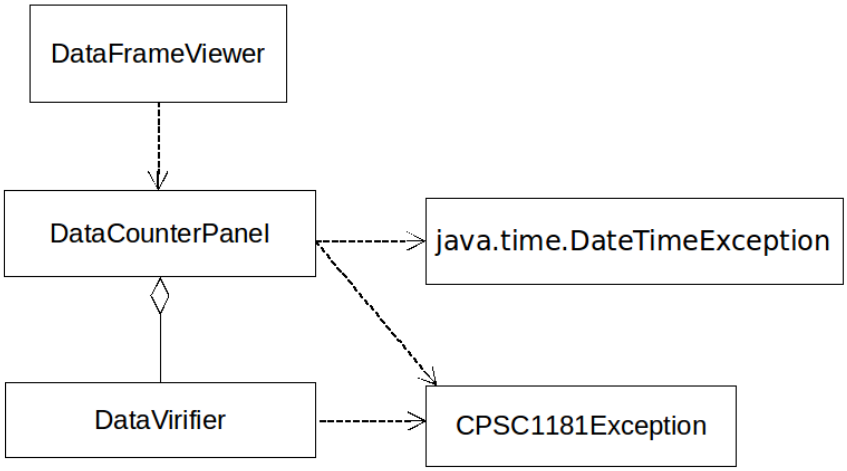
The default ending date must be today's date, i.e. the day when the program is being run by your marker. If the dates entered are correct, a popup window gives the number of days (months, years) between the dates; however, if there is an error in the input, a detailed error about the date is given in a popup window.

Possible errors to detect are missing parts of the date, no date given, an impossible date, a negative date, a date with characters and so on. Your program should handle errors gracefully and it should report the errors with an informative message.

The date should be in the format: DD/MM/YYYY, where D, M, and Y are all digits.

Allowed, as well, are D/M/YYYY and D/MM/YYYY and DD/M/YYYY. Note that in this assignment you may **not** assume that the input is numeric. In addition, DD should be less than or equal to 28, 30 or 31 depending on the month, and MM should be less than or equal to 12. The range of years allowed is 1000 to 3000.

The rough UML diagram of the implementation shown below:



DataFrameViewer starts the program.

DataCounterPanel handles all exceptions.

DataVerifier checks for the format of the input dates for validation and throws exceptions.

CPSC1181Exception is a user defined unchecked exception.

java.time.DateTimeException is thrown by Java bulit-in methods. Your program should handle this type of exception too.

Download code.zip file and run it from command line: java DateFrameViewer

Your program should provide the same messages by throwing and catching exceptions.

Notes:

You must Throw exceptions when errors are detected and implement the appropriate error handlers. Do not just use if statements.

set the font of theTextFields to Monospaced, Bold, and size = 24

set the font of the labels and button to sansSerifs, Bold, and size = 24

Study the following classes:

java.time.LocalDate
java.time.LocalDateTime
java.time.temporal.ChronoUnit
java.time.Period

Use the **JoptionPane.showMessageDialog** method for the popup windows including the result that counts the number of days (or months or years)

Use the classes and the subclasses (and interfaces) in the package **java.time** e.g. **java.time.LocalDate** and **java.time.temporal.ChronoUnit.**

Do not use the Java depreceted classes such as java.util.Calendar nor java.util.Date nor java.util.Gregorian nor any time classes from JDK 5, 6, 7 (i.e. you can only run your code with JDK 1.8).

You can use “between” method of ChronoUnit class to find total days between two instance of LocalDate instances :

```
long days =ChronoUnit.DAYS.between(startLocalDate, endLocalDate);
```

Download JavaTimeSample.java, study, compile, and run it as a demo.

The exepctions that you probably are going to deal with are NumberFormatException, DateTimeException, and your own exception CPSC1181Exceptrion classes depends on your implementation. Refer to Java API for the methods you are using from Java Library and study the exceptions they may throw.

What to upload

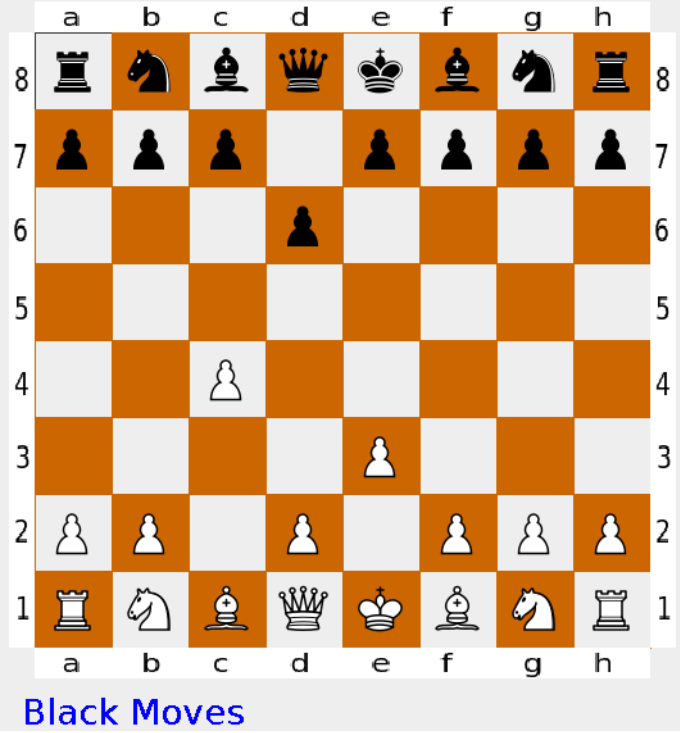
Create one page UML diagram of your design, and save it either as PDF or as word 2003 (JPEG Inserted into word document).

Zip your source files and UML diagram into *your StudentID.zip* file and then upload it to lab9 in D2L Dropbox.

TOTAL MARK: 50

Bonus: [5 marks]

Download border images of the the chess program and save them in image directory of the program. Use the side.png and top.png and add them to your program.



Notes:

1. You need to stretch the images to fit sides of the board.
2. Do not use layout management tools to add border edges of the chess program. Just use them as image and draw them around your board.
3. Modify Def class and define images in the class.
4. You can use following methods to read and draw images

```
img = ImageIO.read(new File(imageFileName));
g2.drawImage( … )
```