S21 Challenge: Timer Stopwatch

Description:

Create a Java 8 SE application in NetBeans using JavaFX that implements a stopwatch that has an analog and a digital display. The analog display is comprised of a dial with tick marks and a sweeping hand that displays the elapsed time in 1 second increments with a **smooth response** between seconds. The digital display consists of three parts. First, the digital time label shows the elapsed time in minutes, seconds, and hundredths of a second, mm:ss.SS, and displays the elapsed time in 1 centisecond increments (hundredth of a second). Second, the timer label shows the seconds left. Third, a record board shows the lap time and total time when the record button is pressed.

Before the stopwatch appears on the display, there should be a Text Input Dialog asking for a start time for the timer. The dialog should receive an integer number and your timer will start from the input number. **NOTE:** Make sure you have proper error checking for the input fields, it should be a positive integer.

The stopwatch has two buttons. When the stopwatch is first initialized, the two buttons will be “Record” and “Start”. When the stopwatch is running, the two buttons will be “Record” and “Stop”. When the stopwatch is stopped, or paused, the two buttons will be “Reset” and “Start”. The text on the two buttons should be displayed dynamically and change dynamically.

The timer label starts from the time you input in the dialog (e.g. 60) and displays the time left after the stopwatch started, for example, if 5 seconds has passed after the stopwatch has started, the timer label will show “Timer: 55.00s”. When the passed time arrives at 60 seconds, the timer label will show “Time's Up!” and “Record” button will stop working, when the button is clicked, and an alert will appear with the message “Time is up… No more records…”. In addition, the stopwatch will not be paused and will continue operating as normal no matter what happens with the timer, because the stopwatch will only be paused when you click “Stop” button.

The record board will be used to record the lap time and total time between different clicks. When the “Record” button is clicked, the calculated difference and total time will be recorded and displayed on the record board, where the newest record will always replace the oldest one. The record board consist of eight items (two labels and six text objects). The two labels will display “Lap Time” and “Total Time”, respectively, to show the header at the top of the board. If the stopwatch has not started yet, then the record button should not have any action.

NOTE: Labels should be used for text that will not frequently change whereas Text Objects should be use for text that will changed frequently. If you use a Label and change the text frequently, then points will be deducted. The first sentence on the Oracle Docs (our textbook) says a “A Label is a non-editable text control,” so make sure you use it as such.

https://docs.oracle.com/javase/8/javafx/api/javafx/scene/control/Label.html

S21 Challenge: Timer Stopwatch

Purpose:

This challenge develops skills in creating and manipulating JavaFX interfaces, generating and handling events, having dynamically generated layouts, and organizing code correctly.

Requirements:

Project Name: <Pawprint>TimerStopwatchS21

For the Project Name, follow the same naming scheme used in the first challenge. The Project Name is to be comprised of your pawprint with the first letter capitalized followed by TimerStopwatchS21. For example, if the pawprint is **abcxyz9** the project is to be named **Abcxyz9TimerStopwatchS21**.

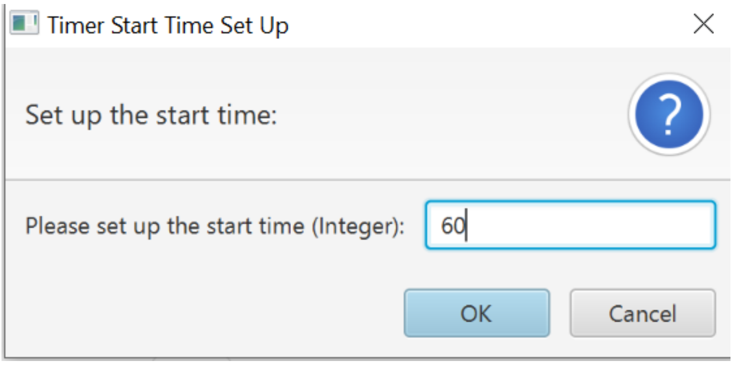
This challenge allows you to apply your creativity while meeting a set of requirements. Not every aspect of the application is described in the requirements. The requirements establish what functionality must exist, but it is up to you to apply your creativity in implementing the functionality and the user interface to support that functionality.

Platform: Java 8 SE and JavaFX.
Project IDE: NetBeans

Create a stopwatch application that contains an analog and a digital display.

- Before the stopwatch appears on the display, there should be a Text Input Dialog asking for a start time for the timer. The dialog should receive an integer number and your timer will start from the input number. **NOTE:** Make sure you have proper error checking for the input fields, it should be a positive integer. See *Figure 1: Timer set up dialog* for an example.
- The analog display, minimally, contains a circular dial with 60 tick marks where each tick mark represents 1 second of elapsed time and a sweeping hand that increments to the next tick mark every elapsed second. See *Figure 2: Analog Interface* for an example. It takes 60 seconds for the hand to sweep one time around the dial. The digital display consists of three parts: a digital time label, the timer label, and the record board.
- The digital time text object, minimally, shows minutes, seconds, and hundredths of a second in a mm:ss.SS format. For each elapsed centisecond (hundredth of a second), the digital time label is to increment 1 centisecond (hundredth of a second). The digital time text object is to increment 1 second after the hundredths of a second is equal to 100.
- The timer text object, minimally, will start from a start second number passed from the Input Dialog and show the remaining time left as it is counting down.
- The record board, minimally, will be used to record the three different lap times and total times between different clicks. See *Figure 3: digital display* for an example.
- **NOTE:** “Minimally” means you can do more if you’d like, which is usually rewarded. You could display hours, milliseconds, or nanoseconds, etc.
- When the timer arrives at 0 seconds (i.e. means 0 seconds left), the timer label will show “Time’s Up!” and the “Record” button will stop working. When the record button is clicked, and an alert will appear with the message “Time is up… No more records…”. In addition, the

stopwatch will not be paused or stopped when the timer reaches 0 and will continue operating as normal no matter what happens with the timer, since the stopwatch will only be paused when you click “Stop” button. See *Figure 4: Alert window* for an example.



Timer Start Time Set Up

Set up the start time:

Please set up the start time (Integer): 60

Figure 1: Timer set up Example

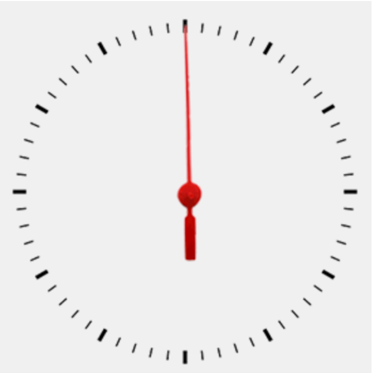
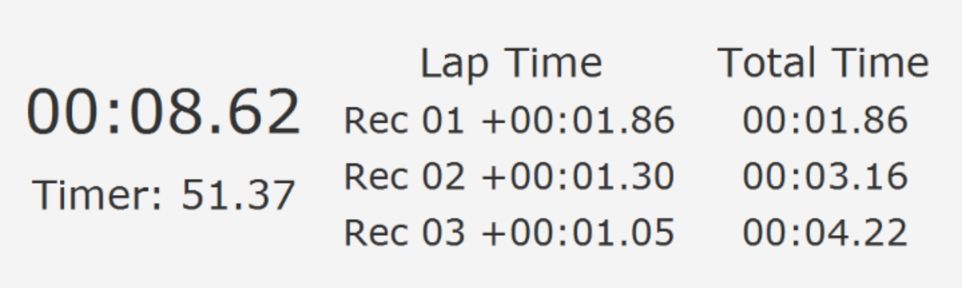
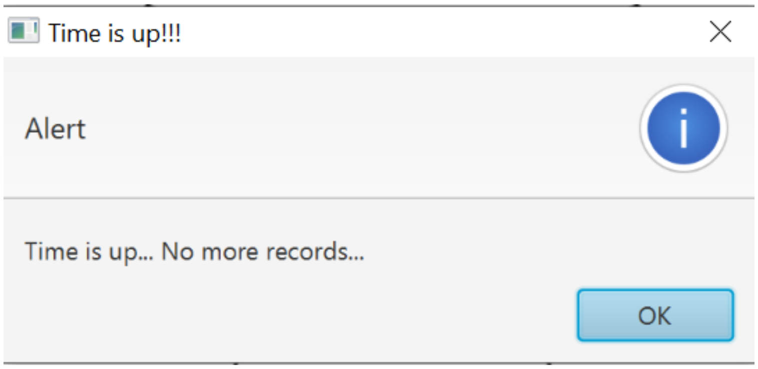


Figure 2:
Analog Interface Example



```
             Lap Time            Total Time
00:08.62     Rec 01 +00:01.86    00:01.86
Timer: 51.37 Rec 02 +00:01.30    00:03.16
             Rec 03 +00:01.05    00:04.22
```

Figure 3:
Digital display Example



Time is up!!!

Alert

Time is up... No more records...

Figure 4: Alert Window Example

The stopwatch has **<u>two buttons</u>**. When the stopwatch is initialized, the two buttons will be “Record” and “Start”. When the stopwatch is in motion and running, the two buttons will be “Record” and “Stop”. If the user presses the “Stop” button and the stopwatch is no longer in motion or running, then the two buttons will be “Reset” and “Start”. If the user presses the reset button, the two buttons will return to “Record” and “Start”. Therefore, the buttons will change dynamically depending on the current state of the stopwatch.

The start button will start the stopwatch. The stop button will halt the stopwatch. The reset button resets the displays, elapsed time, places the displays in a halt state, clears the record board, and resets the time left in the timer. The record button will store the current time displayed at the moment the button was pressed as the total time. Then the record button will take the current time and subtract it from the previous lap time and store it as the lap time. Next the record button will display the current lap time and the total time on the correct corresponding text object on the board it the same format as the digital display, mm:ss.SS. Therefore, pressing the record button will result in the difference between the two button clicks or the difference between the current time and the last time of the record button was pressed. If you were able to press the record button, then immediately press the stop button, without letting any time lapse, the sum of all the individual lap times should equal the total time shown on the digital display.

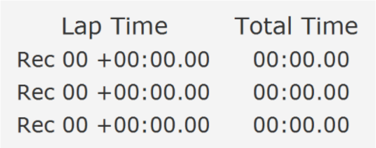
Starting the stopwatch causes it to increment from the last elapsed time value. If the stopwatch is at 0 and the start button is pressed, the stopwatch increments from 0. If the stopwatch is stopped at 00:07.11 and then the start button is selected, the stopwatch will increment from 00:07.11. The reset button is used to set the stopwatch to 0 and place it in a halted state. If the elapsed time of the stopwatch is 00:07.11 and the record button is pressed for the first time, then the current time of 00:07.11 will be displayed on the lap and total time text object. However, if the start button has not been pressed for the first time, meaning the stopwatch has not started yet, then the record button has no action and if pressed, nothing will happen.

The digital display consists of three parts, the digital time text object, a timer text object, and the record board. The timer text object will be initialized as the number the user entered in the dialog. If the start button is pressed and as time passes, the time left will be the start seconds minus the time elapsed, assuming the timer default value is 60 seconds. If the time passed is larger than 60 seconds, the label will always show “Time’s Up!”. When the reset button is pressed, the timer label will be reset to its initial state.

The record board consist of eight elements, two labels and six text objects. When the “Record” button is clicked for the first time, the time on the digital display will be displayed on the first lap time label and first total time label. If the “Record” button is clicked for the second time, the difference between the first click and the second click will be calculated (The difference between the current time and the last time of the record button was pressed), and then displayed in the second lap time. The total time text object will be calculated by totaling up all the different clicks and displayed in the second total time. If the “Record” button is clicked for the third time, you will repeat the procedure, take the difference of the current time and the last time of the record button was pressed, and display the result in the third lap time, while the total will be the addition of all the button clicks and displayed in the third total time text object. However, if the “Record” button is clicked for the fourth time, the first lap and total time label will be changed, and this process is repeated. Therefore, the newest record will always replace the oldest one. After the "Reset" button is clicked, the record board will be initialized back to "00:00.00" for all the text objects, the lap count, the lap times and total times will be set back to zero, and the record board should look the same as it was in the beginning.

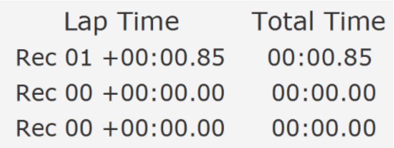
Here are some examples of the record board:

At the Beginning



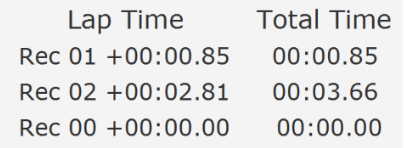
Lap Time	Total Time
Rec 00 +00:00.00	00:00.00
Rec 00 +00:00.00	00:00.00
Rec 00 +00:00.00	00:00.00

After First Click



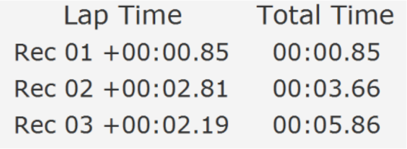
Lap Time	Total Time
Rec 01 +00:00.85	00:00.85
Rec 00 +00:00.00	00:00.00
Rec 00 +00:00.00	00:00.00

After Second Click



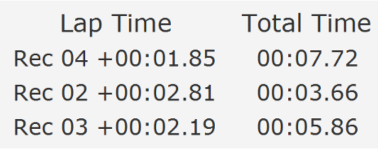
Lap Time	Total Time
Rec 01 +00:00.85	00:00.85
Rec 02 +00:02.81	00:03.66
Rec 00 +00:00.00	00:00.00

After Third Click



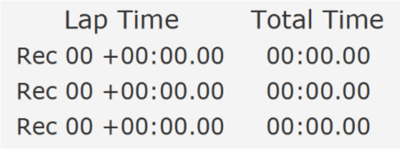
Lap Time	Total Time
Rec 01 +00:00.85	00:00.85
Rec 02 +00:02.81	00:03.66
Rec 03 +00:02.19	00:05.86

After Fourth Click



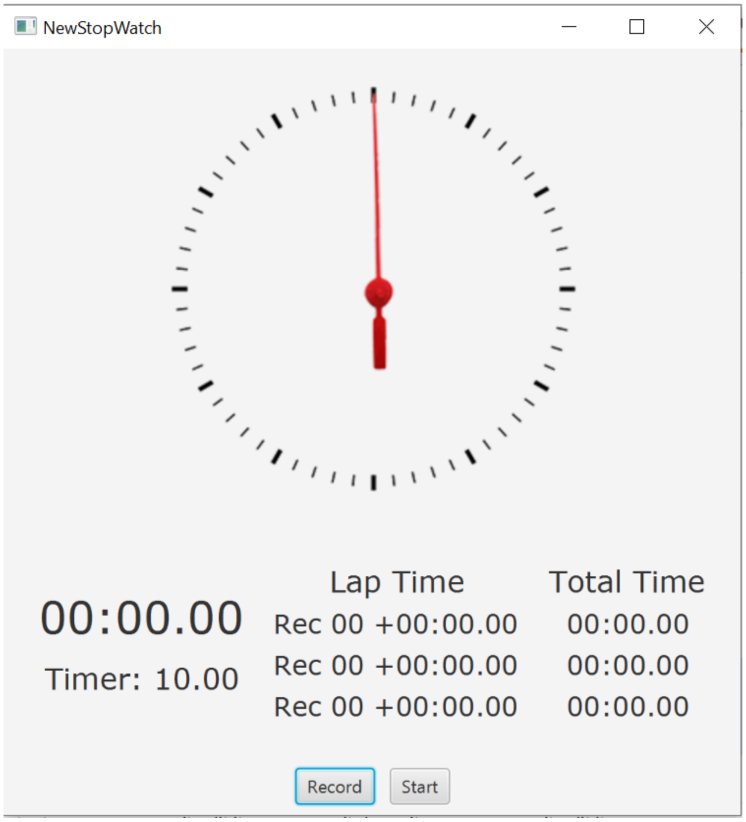
Lap Time	Total Time
Rec 04 +00:01.85	00:07.72
Rec 02 +00:02.81	00:03.66
Rec 03 +00:02.19	00:05.86

After Reset Button



Lap Time	Total Time
Rec 00 +00:00.00	00:00.00
Rec 00 +00:00.00	00:00.00
Rec 00 +00:00.00	00:00.00

Here are some examples of the timer stopwatch, your design does not need to be exactly the same, it can be more fashionable and have a nicer UI design. In addition, if there are some differences between the examples and the requirements page, then make sure you follow the requirements page:

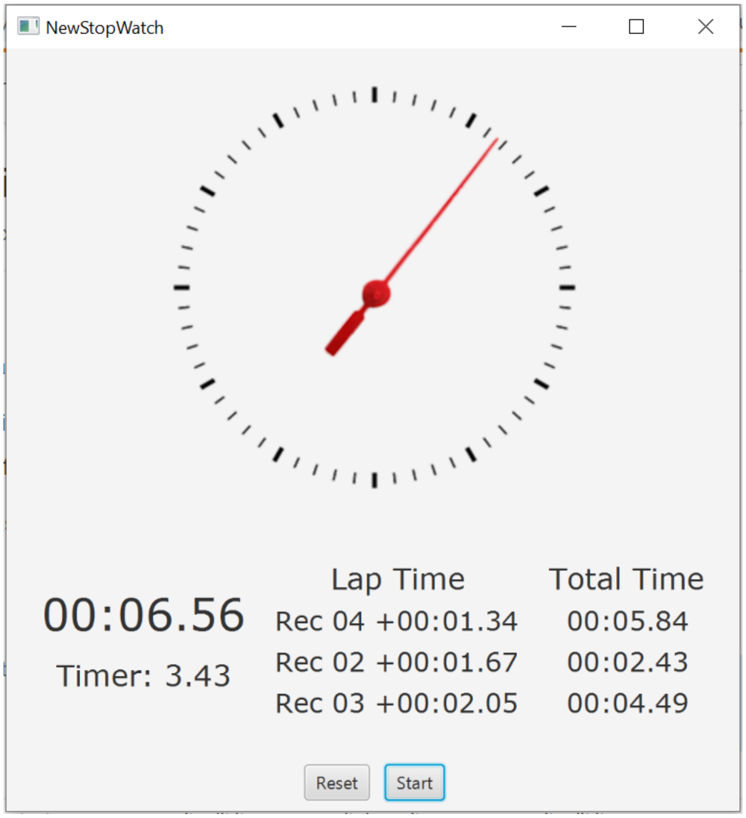


00:00.00

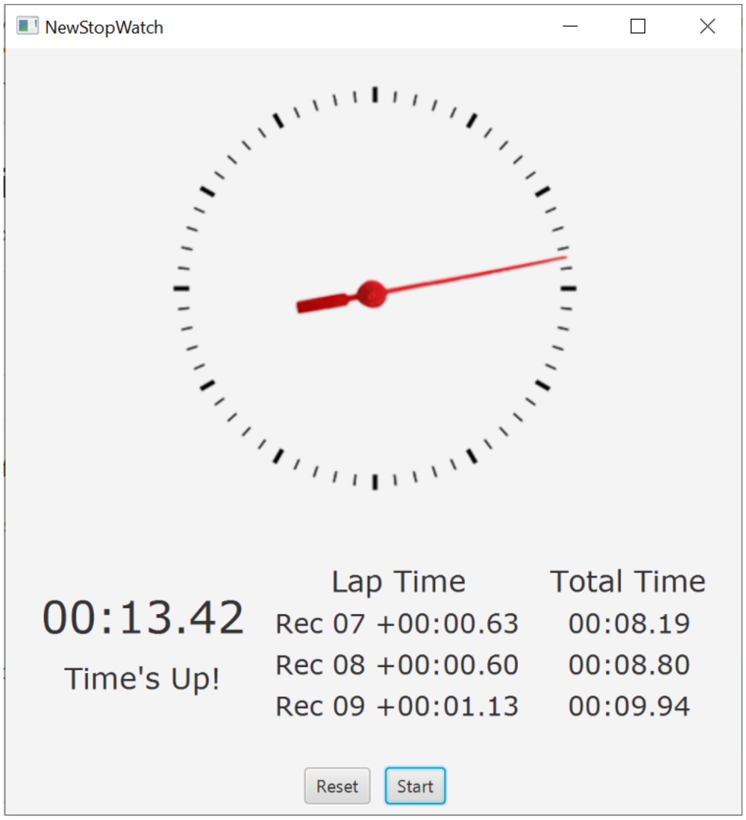
Timer: 10.00

Lap Time	Total Time
Rec 00 +00:00.00	00:00.00
Rec 00 +00:00.00	00:00.00
Rec 00 +00:00.00	00:00.00

S21 Challenge: Timer Stopwatch



```
                 Lap Time           Total Time
00:06.56   Rec 04 +00:01.34      00:05.84
           Rec 02 +00:01.67      00:02.43
Timer: 3.43 Rec 03 +00:02.05     00:04.49
```



```
                 Lap Time           Total Time
00:13.42   Rec 07 +00:00.63      00:08.19
           Rec 08 +00:00.60      00:08.80
Time's Up! Rec 09 +00:01.13      00:09.94
```

S21 Challenge: Timer Stopwatch

Two image files are provided with this challenge that you can use: clockface.png and hand.png. You are not required to use the provided images. If you want to use alternatives, that is okay. The clockface.png image is the dial of the stopwatch. The hand.png image is the sweeping hand of the stopwatch.

How you layout the user interface is up to you. You get to decide the locations of the analog display, digital display, and the two dynamically changing buttons. Whatever you choose should be a well-organized, thoughtful, aesthetically pleasing, and useable interface or an interface that logically makes sense. The layout should look like you made intentional choices and are in control of their placement. This means the layout should not be a disorganized mess that is a result of not knowing how to implement the user interface, layout, and/or code in a meaningful way.

You may expand the functionality beyond the basic requirements provided above if you choose. Usually expanding beyond the basic requirements gets rewarded. You could do the following for example:

- Add an additional sweeping analog hand that displays the minutes elapsed
- Add an additional sweeping analog hand that displays the hours elapsed
- Add an additional smaller hand, that is underneath the seconds hand, that displays hundredths of a second on the analog stopwatch
- Add hours to the digital display
- Add milliseconds to the digital display
- Add nanoseconds to the digital display
- Add additional functionality to the stopwatch (Could add additional buttons, etc.)
- Change the colors of the buttons (For example, when the “Start” button is showing the button’s color could be green and when the “Stop” button is showing the button’s color could be red)
- Add additional record labels
- Calculating more information for the record board or displaying more information than the minimum requirements listed here

For inspiration: https://en.wikipedia.org/wiki/Stopwatch

S21 Challenge: Timer Stopwatch

Things to Submit on Canvas:

1. You will use NetBeans to export a ZIP file and submit the ZIP on Canvas.
    - Run your application and make sure everything works as expected
    - Make sure you include all files associated with your challenge, if you use images, make sure you use relative paths and include the images in your project directory
        - Use a similar naming convention as the project name for ALL assets for your projects
        - For example if you use your own dial image or hand image then name it
            - <Pawprint>TimerStopwatchS21DailImage.png
            - or <Pawprint> TimerStopwatchS2HandImage.jpg … etc.
    - Export the project directory located in your NetBeans project folder
    - Make sure to add the zip extension, otherwise it will not save properly
    - Then submit that zip file to Canvas
2. Submit screenshots of your application running for proof with the system clock and pawprint. Screenshots can be submitted as extra files on the Canvas assignment submission (preferred method) along with your zip file (All in one submission)
    - The system clock must include the current **date and time** in order to be valid
    - You must take screenshots of the output from your code
        - Show the application running and the functionality you built
        - Similar to the ones shown above as examples
    - You must also take screenshots of **ALL of the code**
    - The more screenshots the better