ASSIGNMENT II

Team Members:

Thomas Sherk
Vikas Gautam
Pengfei Guo

Problem 1: Converter

Analysis:

- The converter should be able to evaluate the values for the given input for Temperature, Height and Mass.
- The values for Temperature can be given in degree **Fahrenheit** (**°F**) or degree **Celsius** (**°C**). So we should be able to evaluate if any of its value is given.
- For Height conversion, we should be able to convert input form **Feet and Inches** to **meters** or vice versa.
- Also for the Mass conversion we should be able to perform the operation of converting **Pounds** to **Kilograms** or vice versa.
- The Unit Converter has a convert button that can perform different methods when different radio button is selected

Problems That Arises:

- When you take the value for one unit say Height, the user might give the input just as numbers like 32 for Feet or the user can give it as 32 feet 12 inches. So, we should be able to convert the value for given input for both the instances into meters. And vise versa.

- We need to make sure only one TextFIeld is inserted text. When one Textfiled is filled, the other one is not allowed to insert data.

- After the execution of conversion of one unit, we should be able to clear out the text Field area for another convert.

- We should be able to toggle from one conversion unit to another when the particular button for the unit is selected like when the user selects to convert the values for Mass then the prompts for the mass conversion should appear and ask the user to input the values for Pounds or Kilograms.

- The prompts for other units should be invisible when a particular unit is selected.

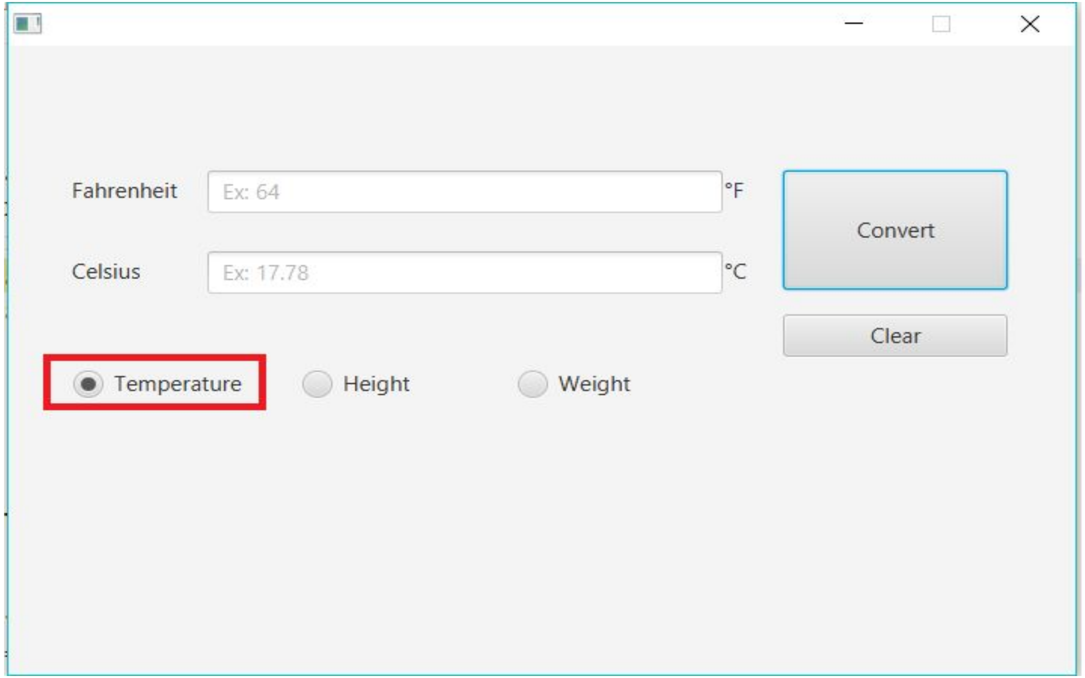
- The convert button can execute different code when different radio button is selected.

Solutions:

- Use if-else function to make sure only one Textfield is inserted.
- Use `&&` to determine which Textfiled is filled, then perform the corresponding conversion.
- Use ToggleGroup to ensure that only one radio button is select at a time, the `getSelectedToggle()` can provide the select radio button.
- The labels for the units of the corresponding radio button can be changed by listening to the status of the ToggleGroup.

Working

If the Temperature button is selected:



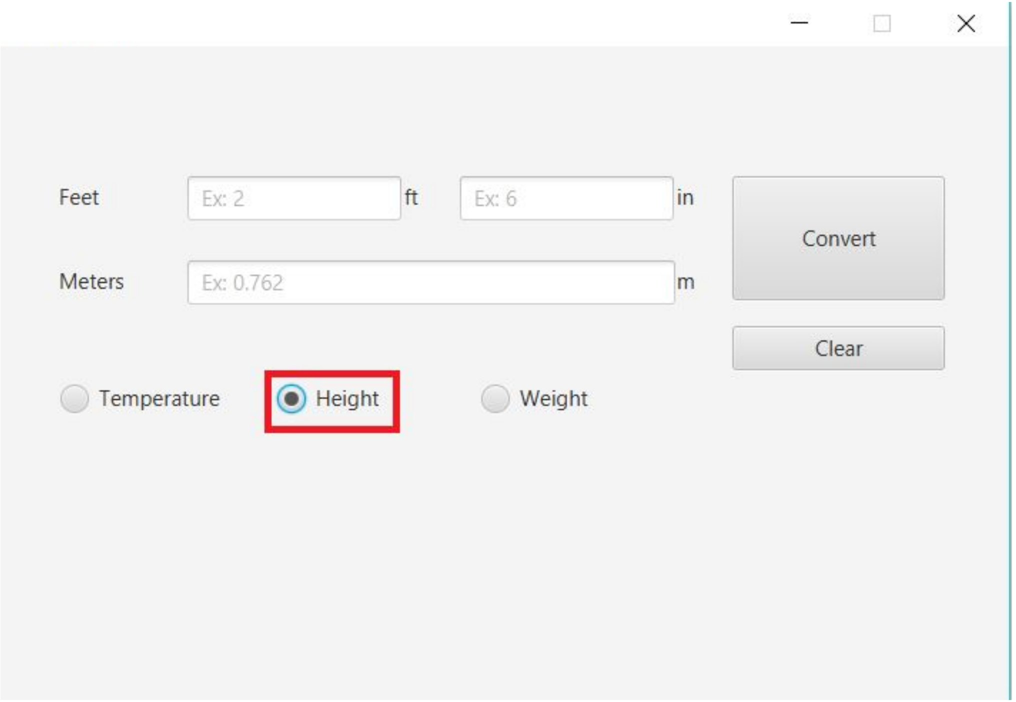
Fahrenheit Ex: 64 °F

Celsius Ex: 17.78 °C

● Temperature ○ Height ○ Weight

Working

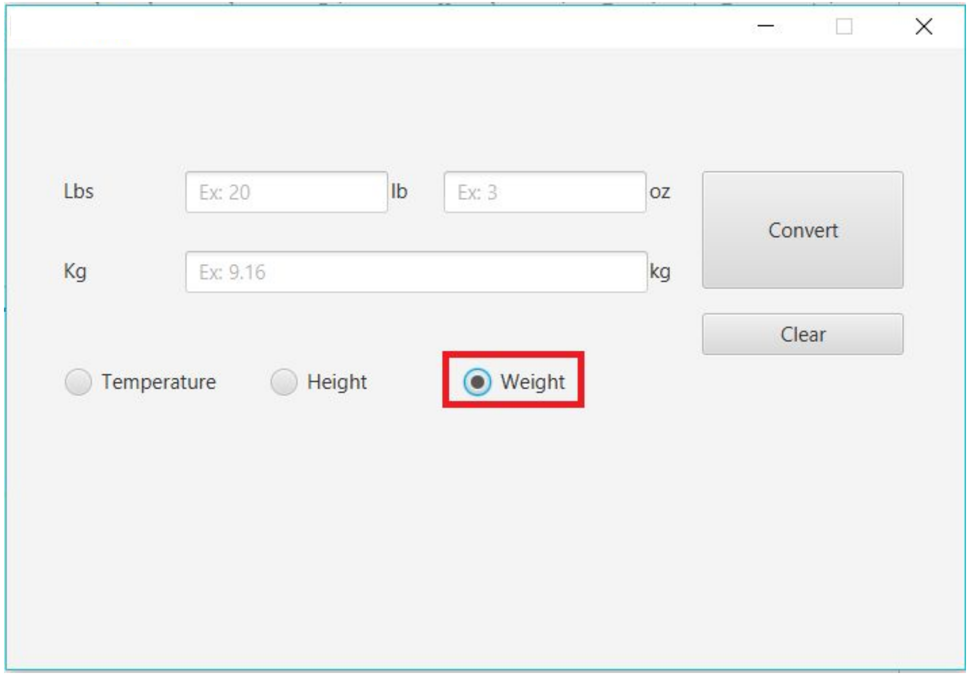
If the Height button is selected:



○ Temperature ● Height ○ Weight

Working

If the Height button is selected:



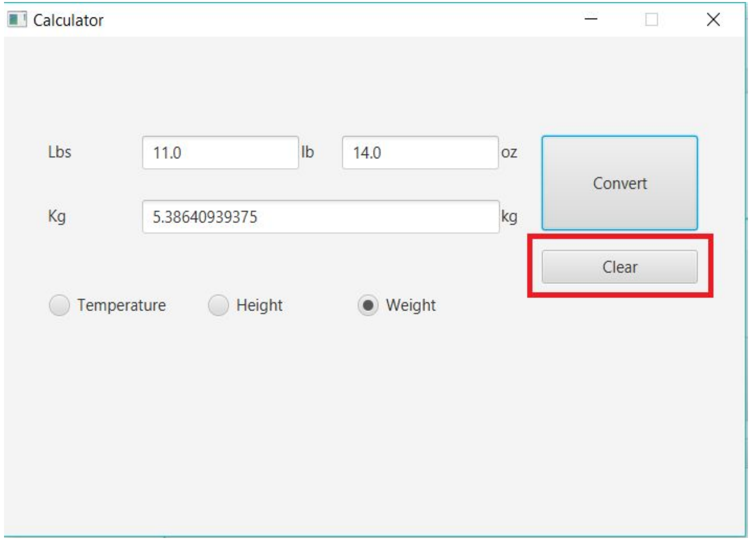
Lbs Ex: 20 lb Ex: 3 oz

Kg Ex: 9.16 kg

○ Temperature ○ Height ● Weight

Working

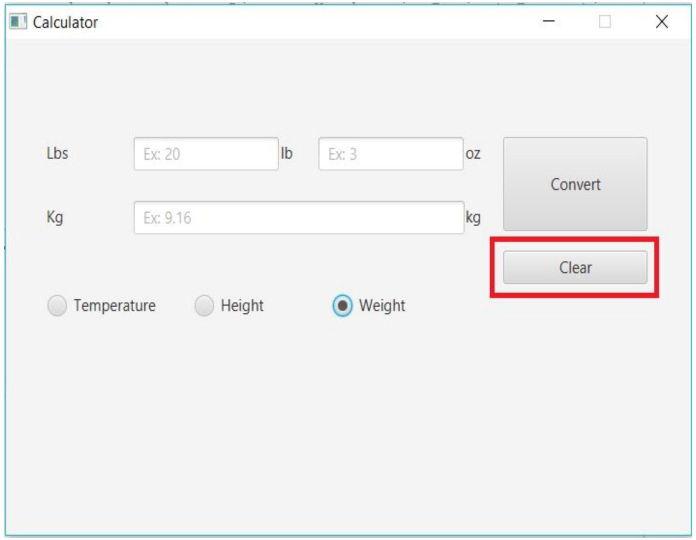
If the Clear button is selected:



Lbs 11.0 lb 14.0 oz

Kg 5.38640939375 kg

○ Temperature ○ Height ● Weight



Lbs Ex: 20 lb Ex: 3 oz

Kg Ex: 9.16 kg

○ Temperature ○ Height ● Weight

Problem 2: Calculator

Analysis:

- The calculator should be able to do the basic calculations, such as plus, minus, multiply, and divide.
- The calculator should also handle big number operations as well as operations that include decimals.
- The calculator should display the results of a calculation between two numbers when the equal sign has been pressed or a second calculation has been initiated with the pressing of one of the operators.
- It should also behave as first-in-first-execute
- Each calculation process should be recorded in a txt file that is cleared and rewritten whenever the program is opened.
- The calculator should be able to convert a positive number to a negative and a negative number to a positive with the +/- sign.
- The calculator should be able to add a decimal to the textfield if there is none.
- The calculator should be able to convert a number to a percentage with the % sign.
- The calculator should be able to clear itself with the clear button.

Issues & Solutions: (1)

Modifying the Textfield, Storing Numbers, and Performing the Operation

1. The basic premise of our calculator revolves around displaying a String in the textfield which is modified based upon which button is pressed.
2. Numbers are added to the end of the string with the push of the numerical buttons. Operators store the string as a double in one of two variables. The operator itself is stored as a string.
3. Each button initializes, adds on, or performs operations depending on what other variables have been initialized through if statements.
4. Example 1: first number has not been initialized. Pressing a number will add a number to the end of the string in the textfield. Pressing ‘plus’ will store whatever is in textfield as a double in the first variable. ‘Plus’ is stored as the operand. When a second number is pressed it will show up as a new number in the textfield. Pressing minus will perform the operation, store the sum in the first variable, store ‘minus’ as the operator, and clear the second variable.

Issues and Solutions (2)

Storing, Writing, and Rewriting Calculations in txt file.

- In the main function, we initialize a file, calculations, with the document “calculations.txt”.
- The main function checks to see if the file exists and, if not, creates a new file.
- Through the use of a PrintWriter and BufferedWriter, the function clears the file.
- If statements in the number button actions and equal button action call a method called writeCalculation that appends the file with each calculation storing the first number + operand + second number + = + output on each line.

Issues & Solutions (3)

Decimals, Negatives, Percentages, and Clear

- The decimal button action is simple. It checks to see if there is a decimal point in the textfield string. If there is not one, it adds it to the end of the string.
- The +/- button uses a . contains method to check if there is a negative sign. If there is one, the button returns the textfield string as a substring minus the first character. If there is not one, the button adds a negative sign to the beginning of the string.
- The percentage sign takes the first number initialized and multiplies it by the second number over 100 and stores that as the second number. If the second number has not been initialized, the % button stores the first number as the second number and then performs the same operation.
- The clear button sets the first and second number as Double.NaN and sets the operand as null.