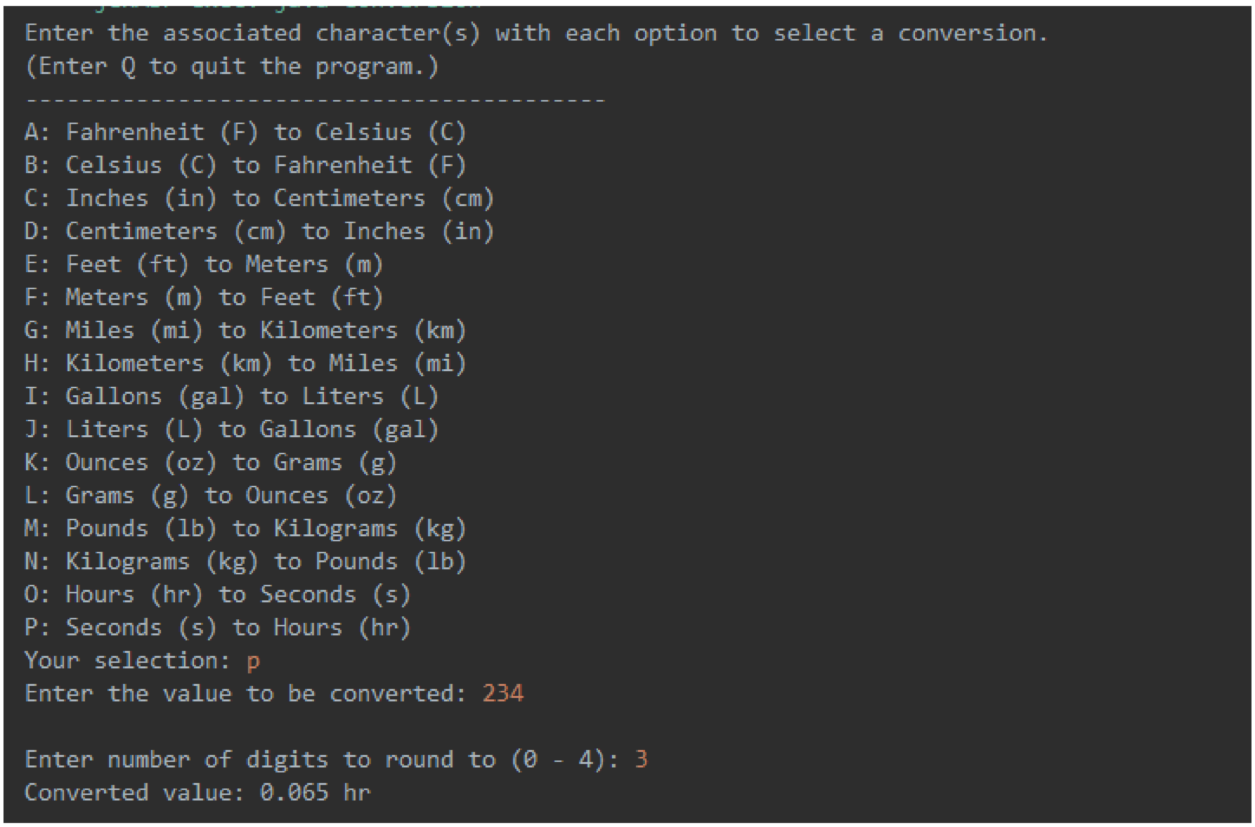
Assignment 3
SWE 437
Dr. Johnson
9/18/2023
Einin Blevins, Erica Therkorn, Christopher Chay

Where the TA can access our source files:

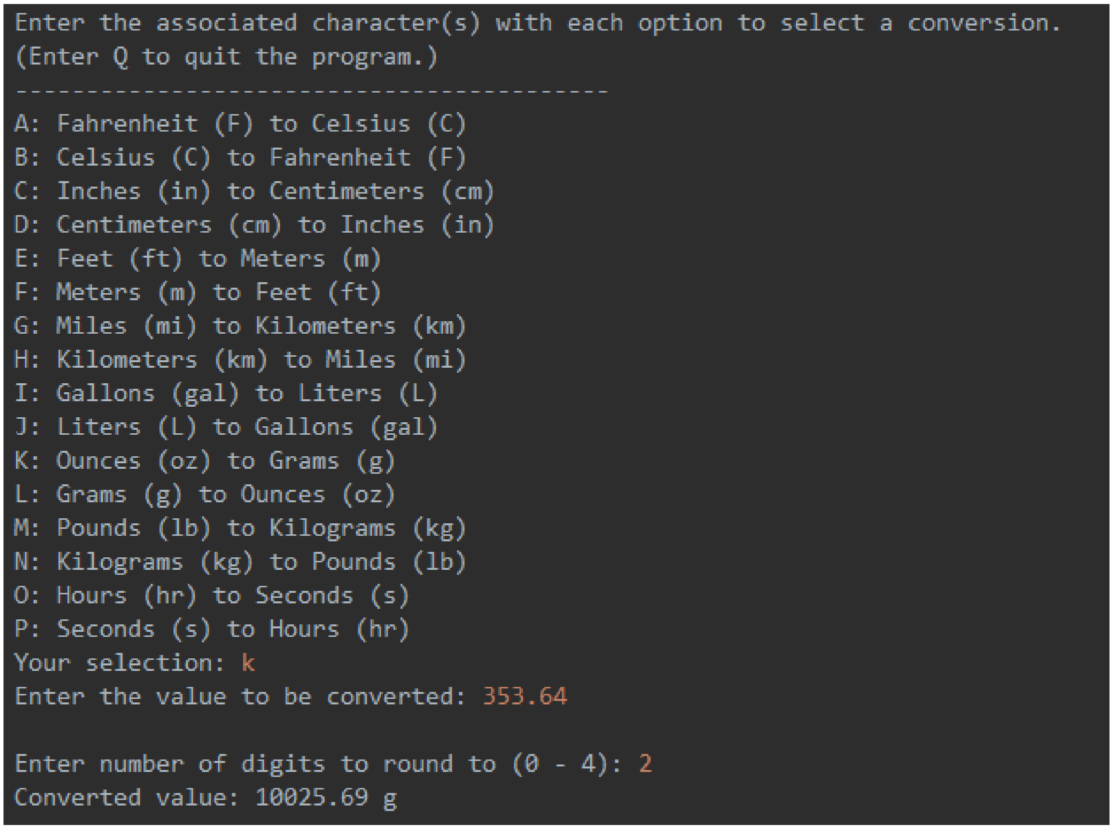
1. Navigate to Assignment3/ to find conversion.java
2. Public Github Repository Link: https://github.com/TheLightningMonarch/GMU_SWE437_PUBLIC/tree/main/Assignment3

Screenshots showing the working software



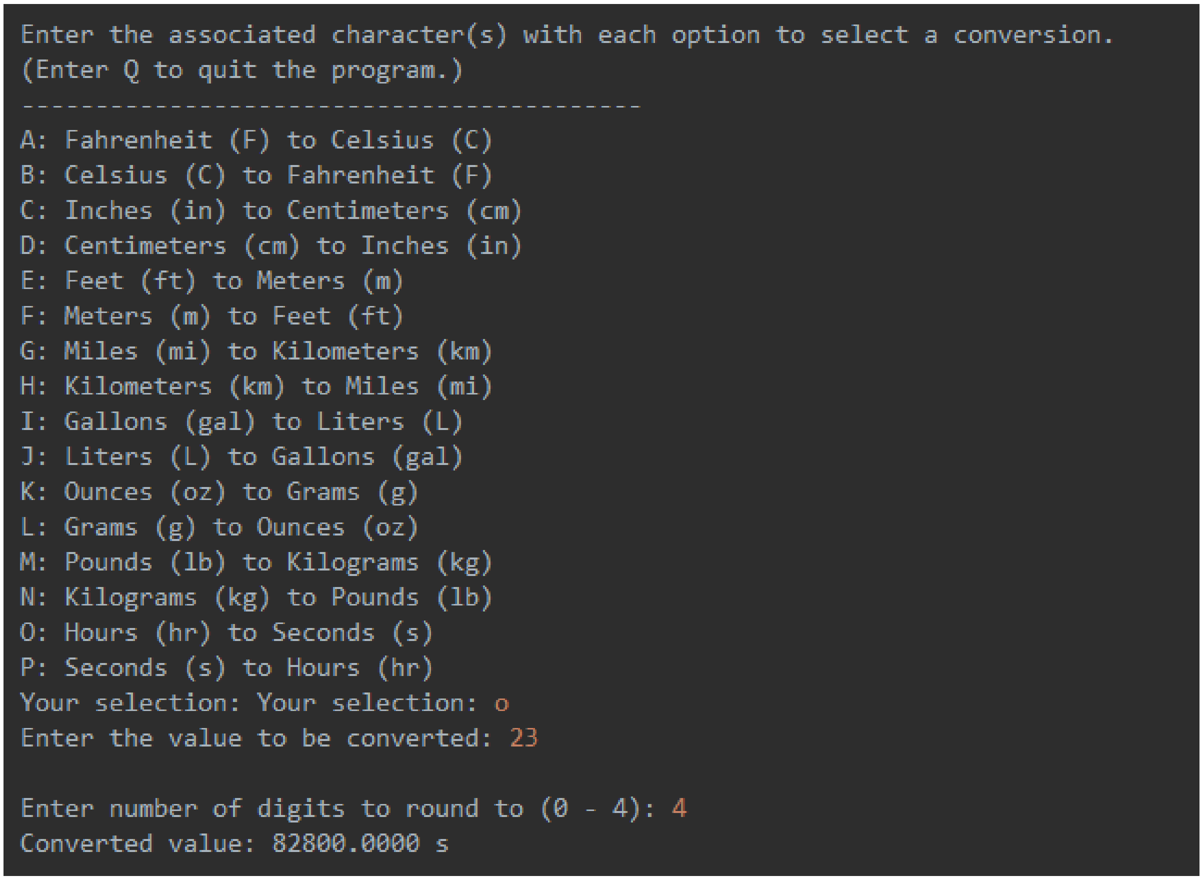
```
Enter the associated character(s) with each option to select a conversion.
(Enter Q to quit the program.)
------------------------------------------
A: Fahrenheit (F) to Celsius (C)
B: Celsius (C) to Fahrenheit (F)
C: Inches (in) to Centimeters (cm)
D: Centimeters (cm) to Inches (in)
E: Feet (ft) to Meters (m)
F: Meters (m) to Feet (ft)
G: Miles (mi) to Kilometers (km)
H: Kilometers (km) to Miles (mi)
I: Gallons (gal) to Liters (L)
J: Liters (L) to Gallons (gal)
K: Ounces (oz) to Grams (g)
L: Grams (g) to Ounces (oz)
M: Pounds (lb) to Kilograms (kg)
N: Kilograms (kg) to Pounds (lb)
O: Hours (hr) to Seconds (s)
P: Seconds (s) to Hours (hr)
Your selection: p
Enter the value to be converted: 234

Enter number of digits to round to (0 - 4): 3
Converted value: 0.065 hr
```



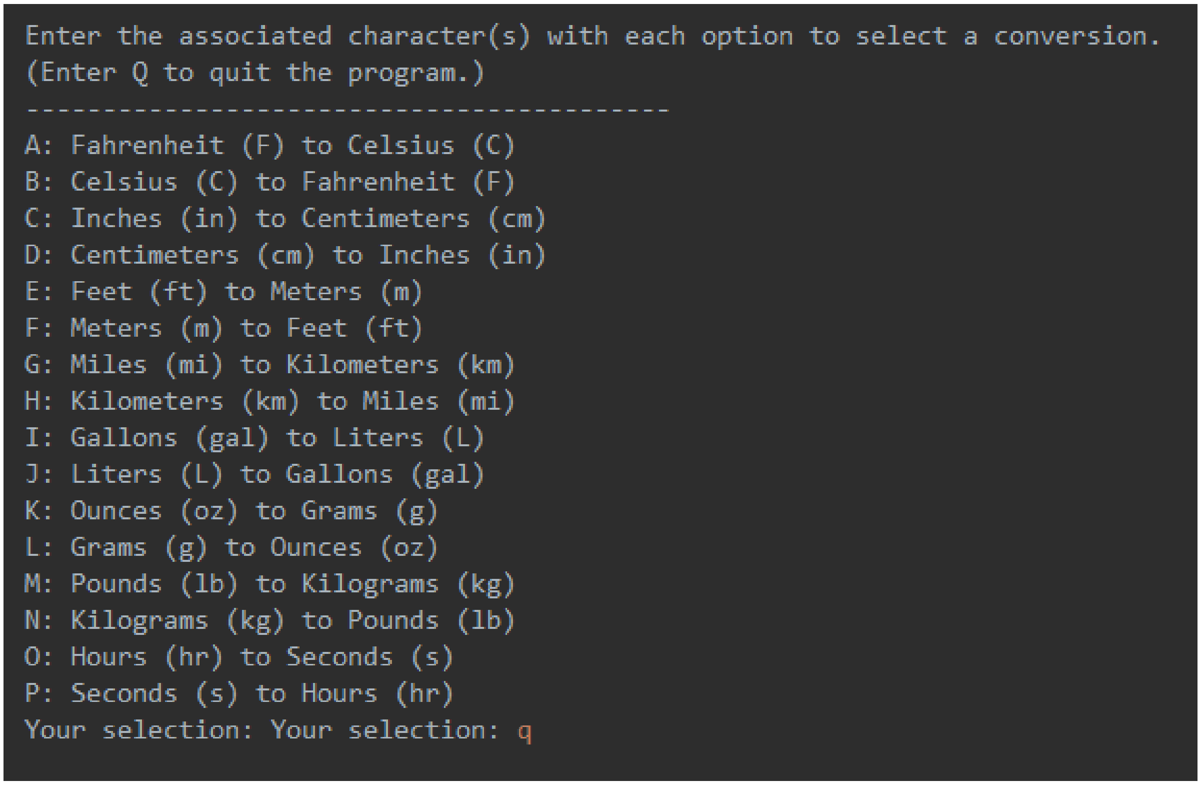
```
Enter the associated character(s) with each option to select a conversion.
(Enter Q to quit the program.)
------------------------------------------
A: Fahrenheit (F) to Celsius (C)
B: Celsius (C) to Fahrenheit (F)
C: Inches (in) to Centimeters (cm)
D: Centimeters (cm) to Inches (in)
E: Feet (ft) to Meters (m)
F: Meters (m) to Feet (ft)
G: Miles (mi) to Kilometers (km)
H: Kilometers (km) to Miles (mi)
I: Gallons (gal) to Liters (L)
J: Liters (L) to Gallons (gal)
K: Ounces (oz) to Grams (g)
L: Grams (g) to Ounces (oz)
M: Pounds (lb) to Kilograms (kg)
N: Kilograms (kg) to Pounds (lb)
O: Hours (hr) to Seconds (s)
P: Seconds (s) to Hours (hr)
Your selection: k
Enter the value to be converted: 353.64

Enter number of digits to round to (0 - 4): 2
Converted value: 10025.69 g
```



```
Enter the associated character(s) with each option to select a conversion.
(Enter Q to quit the program.)
------------------------------------------
A: Fahrenheit (F) to Celsius (C)
B: Celsius (C) to Fahrenheit (F)
C: Inches (in) to Centimeters (cm)
D: Centimeters (cm) to Inches (in)
E: Feet (ft) to Meters (m)
F: Meters (m) to Feet (ft)
G: Miles (mi) to Kilometers (km)
H: Kilometers (km) to Miles (mi)
I: Gallons (gal) to Liters (L)
J: Liters (L) to Gallons (gal)
K: Ounces (oz) to Grams (g)
L: Grams (g) to Ounces (oz)
M: Pounds (lb) to Kilograms (kg)
N: Kilograms (kg) to Pounds (lb)
O: Hours (hr) to Seconds (s)
P: Seconds (s) to Hours (hr)
Your selection: Your selection: o
Enter the value to be converted: 23

Enter number of digits to round to (0 - 4): 4
Converted value: 82800.0000 s
```



```
Enter the associated character(s) with each option to select a conversion.
(Enter Q to quit the program.)
------------------------------------------
A: Fahrenheit (F) to Celsius (C)
B: Celsius (C) to Fahrenheit (F)
C: Inches (in) to Centimeters (cm)
D: Centimeters (cm) to Inches (in)
E: Feet (ft) to Meters (m)
F: Meters (m) to Feet (ft)
G: Miles (mi) to Kilometers (km)
H: Kilometers (km) to Miles (mi)
I: Gallons (gal) to Liters (L)
J: Liters (L) to Gallons (gal)
K: Ounces (oz) to Grams (g)
L: Grams (g) to Ounces (oz)
M: Pounds (lb) to Kilograms (kg)
N: Kilograms (kg) to Pounds (lb)
O: Hours (hr) to Seconds (s)
P: Seconds (s) to Hours (hr)
Your selection: Your selection: q
```

Documentation log

- toBeConverted = getInputMeasurement(); redundancy is moved outside of switch cases
- Moved and replaced num1 = (Float.valueOf(FAsStr).floatValue()); The line where we convert each string to a float inside each conversion function is redundant. So we will convert from string to float once before passing the value to the conversion function, and remove that line from each unit conversion function.
- Made a global array of the options letters (A-N). Moving it from a local variable in the menu function to make it global so that more functions can access it.
    - We want to check the user’s menu selection for validity and handle bad inputs as early as possible
    - Doing this allows us to ask for conversion value input only one time instead of for every switch case.
- Rounded values only at the after the conversion instead of rounding the input of the conversion and the output; this preserves mathematical accuracy and reduces redundancy
    - Also moved rounding to its own separate function to be called after the conversion function, to avoid repetitive code within conversion functions
    - The rounding function sets the digits past the decimal point to user input (any value from 0 to 4).

Maintainability Assessment

The original programmers made it difficult to change the conversion functions because of duplicated functionality. They chose to do everything in the convert functions, whereas it would have been more maintainable if there was a single line of code (the actual conversion) in each function, instead of having the conversion alongside rounding and parsing for floats. Essentially, putting too much functionality in one function made it cumbersome to change each of them.

The original programmers made it easy to change the conversion functions by keeping the logic, format, pattern, and style for each function the same. In one respect, this made modifying the existing functions easy because we knew exactly where to look for the part to be modified. In another respect, having the same format for each conversion function helped as a pattern/ guide for making the new conversion methods (hours and seconds). Additionally, although we are not the original programmers, we did work on the last version of this project, so our previous contributions to the frontend also helped us easily add in new functionality to convert between hours and seconds. All it took was adding variables to a few string/char arrays, creating new cases for our switch statement, and creating new methods to match the previously existing ones. We planned for this type of situation while programming for HW2, so it was easier on us for this assignment.

If we were to do the assignment again, we would start using Git earlier in the process. Because we didn’t set it up at the very beginning, we were programming synchronously (one person types while the other waits, and then the file is sent to the other person). This left gaps of time where one person was not doing much to contribute. Version control would have allowed us to work at the same time and merge our different contributions to separate locations in the program. The use of commit messages and highlighted version differences would aid our maintainability too, since we would be able to quickly identify what was changed and why.

Collaboration Summary

Partner names:

1. Einin Blevins
2. Erica Therkorn
3. Christopher Chay

Contributions:

Einin

[Was responsible for fixing the redundancies in the switch statement that calls each conversion function, writing the new hours to seconds conversion methods, adding points to the documentation log, taking screenshots showing the working software, and writing major contributions to the maintainability assessment.]

Erica

[Was responsible for fixing redundancies in the unit conversion functions, made the new rounding function, contributing changes to the documentation log, major contributions to maintainability assessment]

Christopher

[Set up the GitHub repository and handled final commits, prepared files for submission and submitted, typed up where the TA can find our files]