CSE 101 Programming Assignment 1

Due:

Wednesday, 19-October-2022 by 23:59

Deliverables:

The following Java files should be submitted to MS Teams by the due date and time specified above. Submissions received after the deadline will be subject to the late policy described in the syllabus.

- LifeGoals_{StudentNumber}.java
- BreadStore_{StudentNumber}.java

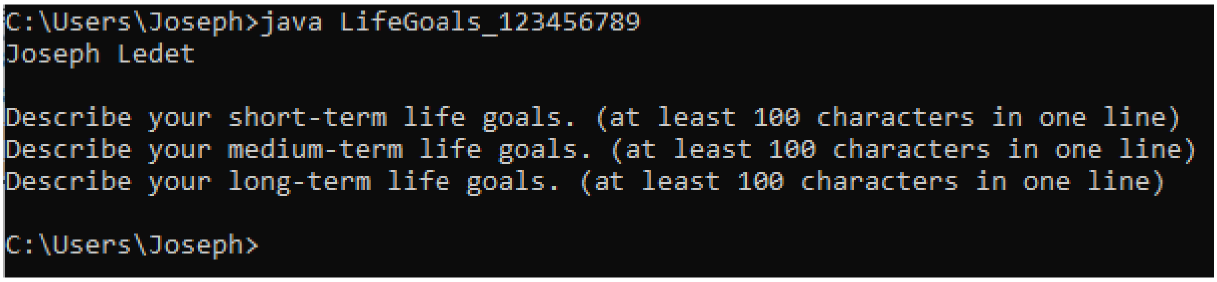
Specifications:

Overview: You will write two programs for this assignment. One will display your short-term, medium-term, and long-term goals to the standard output. The other will keep track of inventory in a store. Do not forget your program headers with Author and Date information.

- **LifeGoals**

Requirements: Write a program that prints your name, your short-term, medium-term, and long-term goals.

Design: Your program should consist of Java instructions to display the following information:



```
C:\Users\Joseph>java LifeGoals_123456789
Joseph Ledet

Describe your short-term life goals. (at least 100 characters in one line)
Describe your medium-term life goals. (at least 100 characters in one line)
Describe your long-term life goals. (at least 100 characters in one line)

C:\Users\Joseph>
```

Describe your life goals (if you have never thought about them, this is a good chance to think about them carefully). The actual output for each line must be at least 100 characters. But remember that the lines of code can NOT exceed 80 characters.

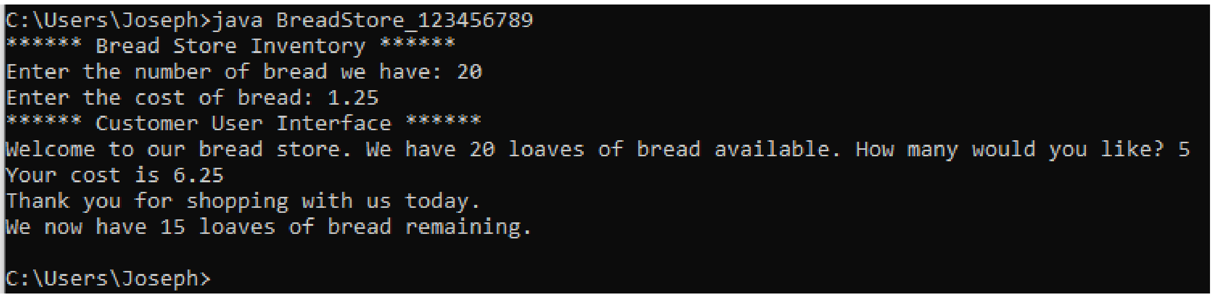
Code and Test: The expected output for the program will vary from student to student, but it is important to follow the output pattern described above, formatting requirements, and minimum character requirements. If you aren't sure how many characters are in your output, you can copy your output into Microsoft Word or a Google doc and use the Word Count feature under the Review ribbon.

- **BreadStore**

Requirements: Write a program that gets information about a product in a store from the owner, then gets the number requested from a customer. It will then display the cost to the customer and how many products are remaining. The following guidelines should be followed:

- Store product
    - Amount must be a whole number 0 or greater
        - If the owner tries to enter a negative number, the system should display an error message and end (NOTE: Do not use System.exit)
    - Cost must be a decimal number 0 or greater
        - If the owner tries to enter a negative number, the system should display an error message and end (NOTE: Do not use System.exit)
- Customer request
    - Amount must be a whole number 1 or greater
        - If the customer tries to enter 0 or a negative number, an Error message must display and no changes to the number of products
    - Amount must not be greater than the number remaining
        - If the customer tries to enter a value greater than the number remaining, an Error message must display and no changes to the number of products
    - The customer should be shown the total cost (number requested X cost)
    - The system should then display the new remaining count of the product

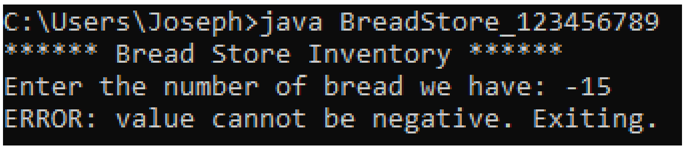
Design: Your program should consist of Java instructions to display the following information (NOTE: bread was used as the product however, there is no requirement for what product you use):



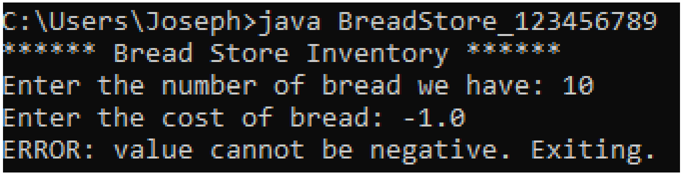
```
C:\Users\Joseph>java BreadStore_123456789
****** Bread Store Inventory ******
Enter the number of bread we have: 20
Enter the cost of bread: 1.25
****** Customer User Interface ******
Welcome to our bread store. We have 20 loaves of bread available. How many would you like? 5
Your cost is 6.25
Thank you for shopping with us today.
We now have 15 loaves of bread remaining.

C:\Users\Joseph>
```

Your program should verify that the owner enters non-negative values:

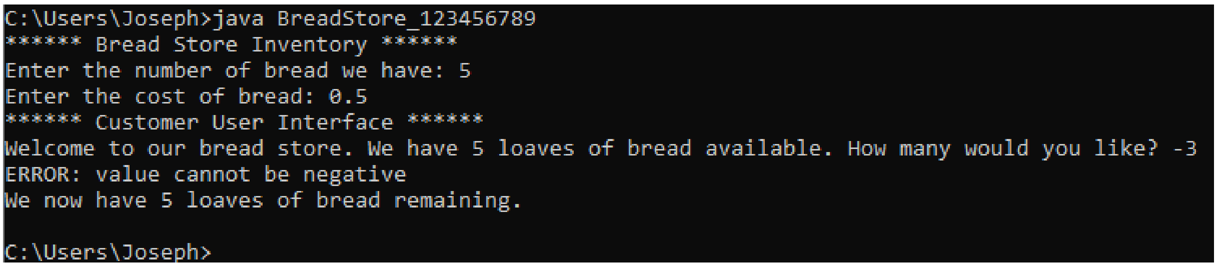


```
C:\Users\Joseph>java BreadStore_123456789
****** Bread Store Inventory ******
Enter the number of bread we have: -15
ERROR: value cannot be negative. Exiting.
```



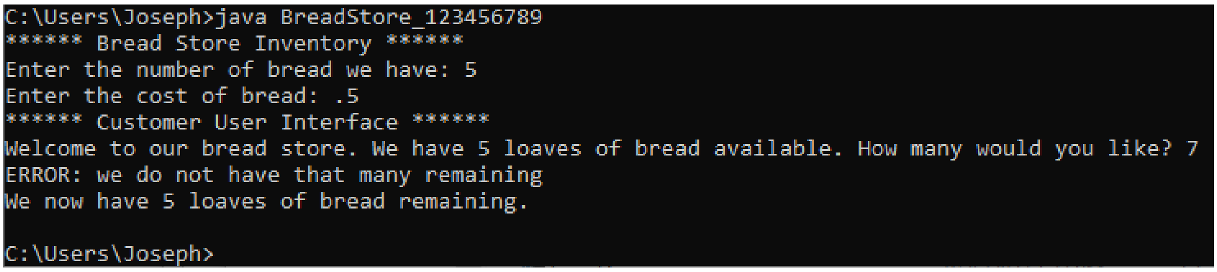
```
C:\Users\Joseph>java BreadStore_123456789
****** Bread Store Inventory ******
Enter the number of bread we have: 10
Enter the cost of bread: -1.0
ERROR: value cannot be negative. Exiting.
```

Your program should finally check to make sure the customer enters a positive number and one that is not greater than we have left:



```
C:\Users\Joseph>java BreadStore_123456789
****** Bread Store Inventory ******
Enter the number of bread we have: 5
Enter the cost of bread: 0.5
****** Customer User Interface ******
Welcome to our bread store. We have 5 loaves of bread available. How many would you like? -3
ERROR: value cannot be negative
We now have 5 loaves of bread remaining.

C:\Users\Joseph>
```



```
C:\Users\Joseph>java BreadStore_123456789
****** Bread Store Inventory ******
Enter the number of bread we have: 5
Enter the cost of bread: .5
****** Customer User Interface ******
Welcome to our bread store. We have 5 loaves of bread available. How many would you like? 7
ERROR: we do not have that many remaining
We now have 5 loaves of bread remaining.

C:\Users\Joseph>
```

Code and Test: Create variables to store the values entered by the owner and customer and assign it using a `Scanner` object.

Grading:

MS Teams Submission: You can submit multiple times, however, we will only grade the last version that you submitted.

NOTE: If you use System.exit() in your code, you will automatically **receive 0 points** for this assignment.