

CSE 101 Programming Assignment 3

Page 1 of 5

Due:

Tuesday, 12-December-2023 by 23:59

Deliverables:

The following Java file 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.

- Assignment3_{StudentNumber}.java

Specifications:

Overview: You will continue the program this week to maintain the grades for a student in a course. Do not forget your headers with @author and @since information. This program will be expanded in future weeks, so be sure you understand the concepts covered in this program.

Requirements: Write a program that will simulate the user interface for a course. It will take arrays as parameters for the grading information. Then will ask the teacher to enter grade information, make grade changes, or display a grade.

To facilitate the execution of this program, you will write and modify (at minimum) the following methods:

1. menu(input, item)
 - a. A new method to display the options based on the contents of the array
 - b. Takes an array of strings and a Scanner object as parameters
 - c. Displays to the output each of the names in the array preceded by a consecutive number ending with "Q - to Quit"
 - d. Requests an input from the user
 - e. Returns an Integer for the user's choice
 - f. If the user requests "Q", returns -1

Code

```
public static void main(String[] args) {
    Scanner inp = new Scanner(System.in);
    String[] list = {"ITEM ONE", "item two"};
    int val = menu(inp, list);
    System.out.println("Returned value - " + val);
    val = menu(inp, list);
    System.out.println("Returned value - " + val);
}
```

Output

```
0 - ITEM ONE
1 - item two
Q - to Quit
2
Returned value - 2
0 - ITEM ONE
1 - item two
Q - to Quit
Q
Returned value - -1
```

2. calculateGrade(category, quantity, weight)
 - a. It will take three parameters
 - i. String array for names of the graded categories
 1. For each item in this array it will change the name to capitalize the first letter and lowercase other letters
 - ii. Integer array for number of each item
 - iii. Integer array for weight of each item
 - iv. If the sizes of the three arrays are not the same or if the values passed are not valid, should display an error message and exit.

- b. It will run according to the description above until the user types "Q" for the prompt.
 - c. NOTE: This method will use much of the main method from your previous assignment
 - d. Returns None
3. formatCategoryName(name)
- a. For each value in the name array, format name with the first letter uppercase; all other letters lowercase
 - b. Takes an array of Strings as a parameter
 - c. Changes the values in the array
 - d. Returns none
 - e. **Note:** You can call your previous formatCategoryName method from this method using method overloading

Code

```
public static void main(String[] args) {  
    String[] list = {"ITEM ONE", "item two"};  
    formatCategoryName(list);  
    System.out.println(list[0]);  
    System.out.println(list[1]);  
}
```

Output

```
Item one  
Item two
```

- 4. Methods from previous assignments that are useful to this assignment may be tested. There should be no change to the requirements for these methods.
- 5. Any other methods you feel helpful can be implemented, however, these will be the only methods tested.

Design:

When calculateGrade is called with valid quantity and weight values, your program should display the list of options to the teacher and ask what he/she would like to do. When the teacher decides to enter all grades, the system should ask for values for all grades. If the teacher asks to change a grade, the system should ask which grade to change, show the current grade and get the new value. If the teacher asks to display the student grade, it should calculate the student's grade based on the values entered so far.

The example on the next page was executed using the following code:

```
public static void main(String[] args) {  
    String[] category = {"QUIZ", "homework", "MidTerm exam", "FINAL Exam"};  
    int[] quantity = {4, 3, 1, 1};  
    int[] weight = {10, 20, 30, 40};  
    calculateGrade(category, quantity, weight);  
}
```

Note: The code above was a main method used to execute the calculateGrade program. You are not required to have a main method. You are encouraged to use a main method to test your calculateGrade program. If you include a main method, it will not be used for grading purposes.

<p>Welcome to our university grade system. Please enter a choice below: 0 - Enter all grades 1 - Display grade information 2 - Change a single grade Q - to Quit 0</p> <p>Please enter grade for Quiz 1 >> 60 Please enter grade for Quiz 2 >> 70 Please enter grade for Quiz 3 >> 80 Please enter grade for Quiz 4 >> 90 Please enter grade for Homework 1 >> 65 Please enter grade for Homework 2 >> 55 Please enter grade for Homework 3 >> 45 Please enter grade for Midterm exam >> 60 Please enter grade for Final exam >> 40</p> <p>Please enter a choice below: 0 - Enter all grades 1 - Display grade information 2 - Change a single grade Q - to Quit 1</p> <p>Category information: Quiz - 75.0 Homework - 55.0 Midterm exam - 60.0 Final exam - 40.0</p> <p>Overall Grade - 52.5 Grade Letter - DC GPA Points - 1.5 Status - conditionally passed</p> <p>Please enter a choice below: 0 - Enter all grades</p>	<p>1 - Display grade information 2 - Change a single grade Q - to Quit 2</p> <p>Please enter the category 0 - Quiz 1 - Homework 2 - Midterm exam 3 - Final exam Q - to Quit 0</p> <p>Please enter which Quiz you would like to change (1 - 4) >> 3 The current grade for Quiz 3 is 80.0 Please enter the new grade value >> 90</p> <p>Please enter a choice below: 0 - Enter all grades 1 - Display grade information 2 - Change a single grade Q - to Quit 2</p> <p>Please enter the category 0 - Quiz 1 - Homework 2 - Midterm exam 3 - Final exam Q - to Quit 3</p> <p>The current grade for Final exam is 40.0 Please enter the new grade value >> 60</p> <p>Please enter a choice below: 0 - Enter all grades 1 - Display grade information 2 - Change a single grade Q - to Quit</p>
	<p>1</p> <p>Category information: Quiz - 77.5 Homework - 55.0 Midterm exam - 60.0 Final exam - 60.0</p> <p>Overall Grade - 60.75 Grade Letter - CC GPA Points - 2.0 Status - passed</p> <p>Please enter a choice below: 0 - Enter all grades 1 - Display grade information 2 - Change a single grade Q - to Quit</p>

Finally, entering Q should exit the system and any invalid value should display an error message and continue:

<p>Please enter a choice below: 0 - Enter all grades 1 - Display grade information 2 - Change a single grade Q - to Quit 4 Invalid choice.</p>	<p>Please enter a choice below: 0 - Enter all grades 1 - Display grade information 2 - Change a single grade Q - to Quit -1 Invalid choice.</p>
<p>Please enter a choice below: 0 - Enter all grades 1 - Display grade information 2 - Change a single grade Q - to Quit q Thank you for using our system. Have a nice day.</p>	

If a teacher enters an invalid choice for changing grades, display an error message and return to the main menu:

<pre>Please enter a choice below: 0 - Enter all grades 1 - Display grade information 2 - Change a single grade Q - to Quit 2 Please enter the category 0 - Quiz 1 - Homework 2 - Midterm exam 3 - Final exam Q - to Quit 5 Invalid choice. Please enter a choice below: 0 - Enter all grades 1 - Display grade information</pre>	<pre>Please enter a choice below: 0 - Enter all grades 1 - Display grade information 2 - Change a single grade Q - to Quit 2 Please enter the category 0 - Quiz 1 - Homework 2 - Midterm exam 3 - Final exam Q - to Quit 1 Please enter which Homework you would like to change (1 - 3) >> 4 Invalid choice. Please enter a choice below: 0 - Enter all grades 1 - Display grade information</pre>
---	---

If we pass invalid values in the arrays to the method, the system should display an error message and exit:

<pre>public static void main(String[] args) { String[] category = {"QUIZ", "homework", "MidTe int[] quantity = {4, 3, 1, -1}; int[] weight = {10, 20, 30, 40}; calculateGrade(category, quantity, weight); }</pre> <p>Assignment3_123456789.java</p> <p>mpile Messages jGRASP Messages Run I/O Interactions</p> <p>nd lear ln</p> <pre>-----jGRASP exec: java -ea Assignment3_12 ERROR: Invalid quantity entered -----jGRASP: operation complete.</pre>	<pre>public static void main(String[] args) { String[] category = {"QUIZ", "homework int[] quantity = {4, 3, 1, 1}; int[] weight = {10, -20, 30, 40}; calculateGrade(category, quantity, wei }</pre> <p>signment3_123456789.java</p> <p>mpile Messages jGRASP Messages Run I/O In</p> <p>d ar</p> <pre>-----jGRASP exec: java -ea Assig ERROR: Invalid weight entered -----jGRASP: operation complete.</pre>
<pre>public static void main(String[] args) { String[] category = {"QUIZ", "homework int[] quantity = {4, 3, 1, 1}; int[] weight = {10, 20, 80, 40}; calculateGrade(category, quantity, wei }</pre> <p>ssignment3_123456789.java</p> <p>mpile Messages jGRASP Messages Run I/O In</p> <p>nd ear ln</p> <pre>-----jGRASP exec: java -ea Assig ERROR: Invalid weight entered -----jGRASP: operation complete.</pre>	<pre>public static void main(String[] args) { String[] category = {"QUIZ", "homework int[] quantity = {4, 3, 1, 1}; int[] weight = {10, 20, 20, 40}; calculateGrade(category, quantity, wei }</pre> <p>signment3_123456789.java</p> <p>mpile Messages jGRASP Messages Run I/O In</p> <p>d ar</p> <pre>-----jGRASP exec: java -ea Assig ERROR: Invalid weight entered -----jGRASP: operation complete.</pre>

If the three arrays have different lengths, the system should display an error message and exit.

```
public static void main(String[] args) {
    String[] category = {"QUIZ", "homework", "MidTerm
    int[] quantity = {4, 3, 1};
    int[] weight = {10, 20, 30, 40};
    calculateGrade(category, quantity, weight);
}
```

signment3_123456789.java

mpile Messages jGRASP Messages Run I/O Interactions

d

```
-----jGRASP exec: java -ea Assignment3_123
ERROR: Array lengths are not all the same
```

Code: Create variables and arrays for the values entered by the user and assign using a single Scanner object. As a user enters valid values update the values of the variables and arrays accordingly.

NOTE: The user will not enter category, quantity, and weight values. Those will be passed to the method using arrays.

Test: You are responsible for testing your program. It is important to not rely solely on the examples presented in this Project description.

Grading:

MS Teams Submission: If anything is ambiguous, it is your responsibility to ask questions. It is also your responsibility to complete this assignment in a timely manner. E-mails with questions regarding this assignment will likely not be answered in time if received after 17:00 on the due date of the assignment. You can submit multiple times, however, we will only grade the last version that you submitted. Be sure to click the “Turn In” button.

Filename: You must name your java file according to the description above. If your file is not named in this way, your submission for this assignment will not be accepted.