CS1341 – Lab #3 Due Saturday 10/01/2022 at 6:00 am

Pre-Lab [10 POINTS]

Must be done prior to your lab session and brought to your lab class to be graded.

Write a Java program called Discount which does the following:

In a while loop,

1. Prompt the user to enter the price of an item and the discount rate
2. Calculate the discount
3. Print the discount amount
4. Ask the user if they want to calculate another discount (if you haven't covered String comparisons in class yet, use 1 for yes, 2 for no – see both options in the sample output below)

Bring the working Discount.java and Discount.class files to your lab session for pre-lab credit.

A sample output is provided below where user input is highlighted in yellow.

```
Enter the price of the item: $299.99
Enter the discount rate: 0.10
The discount is: $30.00
Do you want to calculate another discount? (Enter y for Yes): y
//OR Do you want to calculate another discount? (Enter 1 for yes 2 for no): 1
Enter the price of the item: $69.99
Enter the discount rate: 0.2
The discount is: $14.00
Do you want to calculate another discount? (Enter y for Yes): n
//OR Do you want to calculate another discount? (Enter 1 for yes 2 for no): 2
Good-bye.
```

GRADING: Comment your program to explain your steps. Each program should compile without errors and should run to produce outputs described. The following points will be discounted if the related element is missing or incorrect:

- Output formatting [2 points each]
- Proper names for classes, variables, and methods [1 point each]
- No Comments [5 points]
- Program doesn't compile [5 points for each minor error up to 5 errors provided that after fixing the errors the program compiles. If the program does not compiler after the 5 errors are fixed, partial credit will be given not to exceed 50 points]
- Source code (java file) missing [15 points each]
- Both java files and class files missing [90 points]
- Missing method where required [5 points each]
- Missing loop where a loop is required [5 points each]
- Missing if-else where required [5 points each]

Plagiarism or collaboration with anyone other than your professor, lab instructor, or CS help desk personnel will result in no credit for the assignment and possible honor code violation. Plagiarism is inclusion of any line of code that was created by another person, regardless of the source.

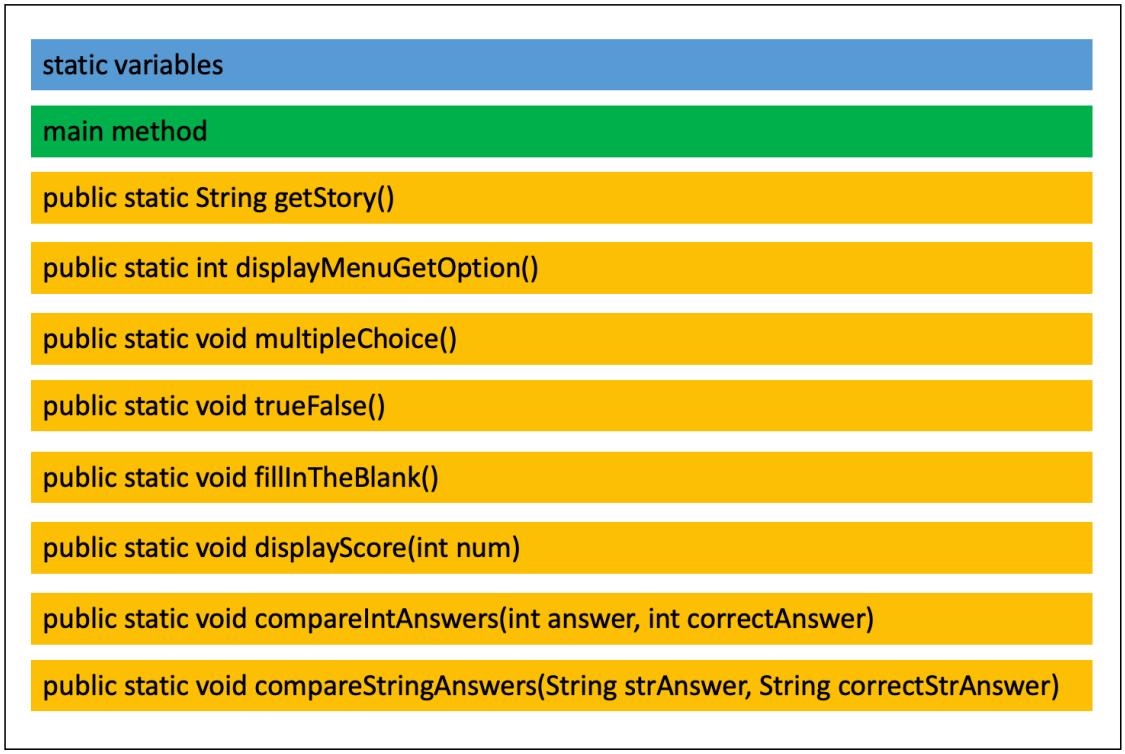
Complete the code for each of the lab problems so they compile and run successfully. Submit the java and class files via Canvas (as a single zip-file). Include a comment block at the top of each Java file that includes your name, student id number, and "**Lab 3-Fall 2022**".

LAB Problem [90 points]

Create a java program called HungryCaterpillar that's going to help an elementary school student practice their comprehension skills.

Here's the general structure of the program which includes the methods' headers showing the return type and parameters for each method.

HungryCaterpillar.java



First, declare the score variable and initialize it to 0. Add the Scanner declaration as well. Here's how to do this:

static int score = 0;

static Scanner input = new Scanner(System.in);

Important Note: A variable declared within a class but outside any method is called a *class member variable* or ***field***, in contrast to a local variable defined inside a method. A field's scope extends from the class's opening brace to the class's closing brace and reaches into methods regardless of where the field is declared within the class.

We now want the following methods included:

main() – We need a main() method, but don't want it to do all the work.

In the main() method, create an int variable called numberOfQuestions and assign it the value 6. Then, call the getStory() method, which returns a String. Print this String. Then in a while loop, perform the following steps:

- Create an int variable called choice, which will hold the value returned by the displayMenuGetOption() method
- If the choice is 1, then call the multipleChoice() method
- If the choice is 2, then call the trueFalse() method
- If the choice is 3, then call the fillInTheBlank() method
- If the choice is 4, then call the displayScore() method passing it the numberOfQuestions variable as an argument
- If the choice is 5, then print Good-bye and break out of the loop
- Otherwise, print Try again to continue looping

getStory() – This method returns the String which represent the story. You can utilize this code for the getStory() method.

```
public static String getStory() {
        String copyright = "\nCopyright 1969, 1987 by Eric Carle LLC.\n\n";
        String titleAuthor = "\nTHE VERY HUNGRY CATERPILLAR by Eric Carle\n\n";
        String story = "\"In the light of the moon a little egg lay on a leaf."
                + "\nOne Sunday morning the warm sun came up and -pop!- out of the egg came a tiny
and very hungry caterpillar."
                + "\nHe stared to look for some food."
                + "\nOn Monday he ate through one apple. But he was still hungry."
                + "\nOn Tuesday he ate through two pears, but he was still hungry."
                + "\nOn Wednesday he ate through three plums, but he was still hungry."
                + "\nOn Thursday he ate through four strawberries, but he was still hungry."
                + "\nOn Friday he ate through five oranges, but he was still hungry."
                + "\nOn Saturday he ate through one piece of chocolate cake, one ice-cream cone,
one pickle, one slice of Swiss cheese, "
                + "one slice of salami, \none lollipop, one piece of cherry pie, one sausage, one
cupcake, and one slice of watermelon."
                + "\nThat night he had a stomachache!"
                + "\nThe next day was Sunday again. The caterpillar ate through one nice green
leaf, and after that he felt much better."
                + "\nNow he wasn't hungry anymore - and he wasn't a little caterpillar anymore."
                + "\nHe was a big, fat caterpillar."
                + "\nHe built a small house, called a cocoon, around himself. He stayed inside for
more than two weeks. "
                + "Then he nibble a hole in the cocoon, pushed his way out and...he was a beautiful
butterfly!\"";
        String theEnd = "\n\nThe End.\n";
        return copyright + titleAuthor + story + theEnd;
}
```

displayMenuGetOption() – This method displays the menu to the user, which shows them the available options to choose from. The method returns the user's choice. Examine the sample output to determine what the menu looks like and what the different choices are that you must print to the user.

multipleChoice() – This method displays the multiple-choice questions to the user one question at a time. For each question, the method gets the user's answer and compares the user's answer to the correct answer by calling a comparison method (see compareIntAnswers() method). Examine the sample output to determine what the questions and the choices are that you must print to the user.

trueFalse() – This method displays the true/false questions to the user one at a time. For each question, the method gets the user's answer and compares the user's answer to the correct answer by calling a comparison method (see compareIntAnswers() method). Examine the sample output to determine what the questions are that you must print to the user.

fillInTheBlank() – This method displays the fill in the blank questions to the user one at a time. For each question, the method gets the user's answer and compares the user's answer to the correct answer by calling a comparison method (see compareStringAnswers() method). Examine the sample output to determine what the questions are that you must print to the user.

compareIntAnswers() – This method will receive the user's answer and the correct answer to the question as integer inputs, specifically for multiple-choice and true/false questions. The method will then compare the user's answer to the correct answer. If the user's answer is correct, the method will display a message to the user confirming that this is the case, and will update the score. Recall that score is a static variable that was mentioned at the beginning of the instructions. Otherwise, the method will inform the user that their answer is incorrect.

compareStringAnswers() – This method will receive the user's answer and the correct answer to the question as String inputs for fill in the blank questions. The method will then compare the user's answer to the correct answer using String comparison. If the user's answer is correct, the method will display a message to the user confirming that this is the case and will update the score. Recall that score is a static variable that was mentioned at the beginning of the instructions. Otherwise, the method will inform the user that their answer is incorrect.

displayScore() – This method displays the user's score. In addition, if the score is:

- Greater than or equal to 5, the method prints "Excellent!"
- Equals to 3 or 4, the method prints "Very good!"
- Otherwise, the method prints "Try taking the test again."

A sample output is provided below where user input is highlighted in yellow.

```
Copyright 1969, 1987 by Eric Carle LLC.


THE VERY HUNGRY CATERPILLAR by Eric Carle

"In the light of the moon a little egg lay on a leaf.
One Sunday morning the warm sun came up and -pop!- out of the egg came a tiny and very hungry
caterpillar.
He stared to look for some food.
On Monday he ate through one apple. But he was still hungry.
On Tuesday he ate through two pears, but he was still hungry.
On Wednesday he ate through three plums, but he was still hungry.
On Thursday he ate through four strawberries, but he was still hungry.
On Friday he ate through five oranges, but he was still hungry.
On Saturday he ate through one piece of chocolate cake, one ice-cream cone, one pickle, one slice of
Swiss cheese, one slice of salami,
one lollipop, one piece of cherry pie, one sausage, one cupcake, and one slice of watermelon.
That night he had a stomachache!
The next day was Sunday again. The caterpillar ate through one nice green leaf, and after that he
felt much better.
Now he wasn't hungry anymore - and he wasn't a little caterpillar anymore.
He was a big, fat caterpillar.
He built a small house, called a cocoon, around himself. He stayed inside for more than two weeks.
Then he nibble a hole in the cocoon, pushed his way out and...he was a beautiful butterfly!"

The End.

Choose the type of questions (5 to exit):
1) Multiple Choice
2) True/False
3) Fill in the blank
4) Display score
```

```
5) Exit
1

Answer the following Multiple Choice Questions. Each correct answer is worth 1 point.
-------------------------------------------------------------------------------------
What did the caterpillar eat first?
1) a green leaf
2) a lollipop
3) an apple
4) a plum
1
That's incorrect!

What did the caterpillar become?
1) a beautiful dragonfly
2) a beautiful fly
3) a beautiful ladybug
4) a beautiful butterfly
4
That's correct!

Choose the type of questions (5 to exit):
1) Multiple Choice
2) True/False
3) Fill in the blank
4) Display score
5) Exit
2

Answer the following True/False Questions. Each correct answer is worth 1 point.
-----------------------------------------------------------------------------
A tiny and very hungry caterpillar came out of the little egg.
0 = false; 1 = true
1
That's correct!

A strawberry made the caterpillar feel much better.
0 = false; 1 = true
1
That's incorrect!

Choose the type of questions (5 to exit):
1) Multiple Choice
2) True/False
3) Fill in the blank
4) Display score
5) Exit
3

Answer the following Fill in the Blank Questions. Each correct answer is worth 1 point.
-------------------------------------------------------------------------------------
The caterpillar built a _______ around himself.
coccon
That's incorrect!

The caterpillar ate through _______ pears.
2
That's incorrect!

Choose the type of questions (5 to exit):
1) Multiple Choice
2) True/False
3) Fill in the blank
4) Display score
```

```
5) Exit
4

You scored 2 out of 6.

Try taking the test again.

Choose the type of questions (5 to exit):
1) Multiple Choice
2) True/False
3) Fill in the blank
4) Display score
5) Exit
9

Try again!

Choose the type of questions (5 to exit):
1) Multiple Choice
2) True/False
3) Fill in the blank
4) Display score
5) Exit
5

Good-bye.
```