CPSC 1181 - Lab 4 [40 marks]

Objectives:

- Create subclasses
- Use visibility modifiers to maintain encapsulation during inheritance
- Override methods from super classes in subclasses

Instructions:

- Complete all of the exercises and submit your zip file prior to the due date.

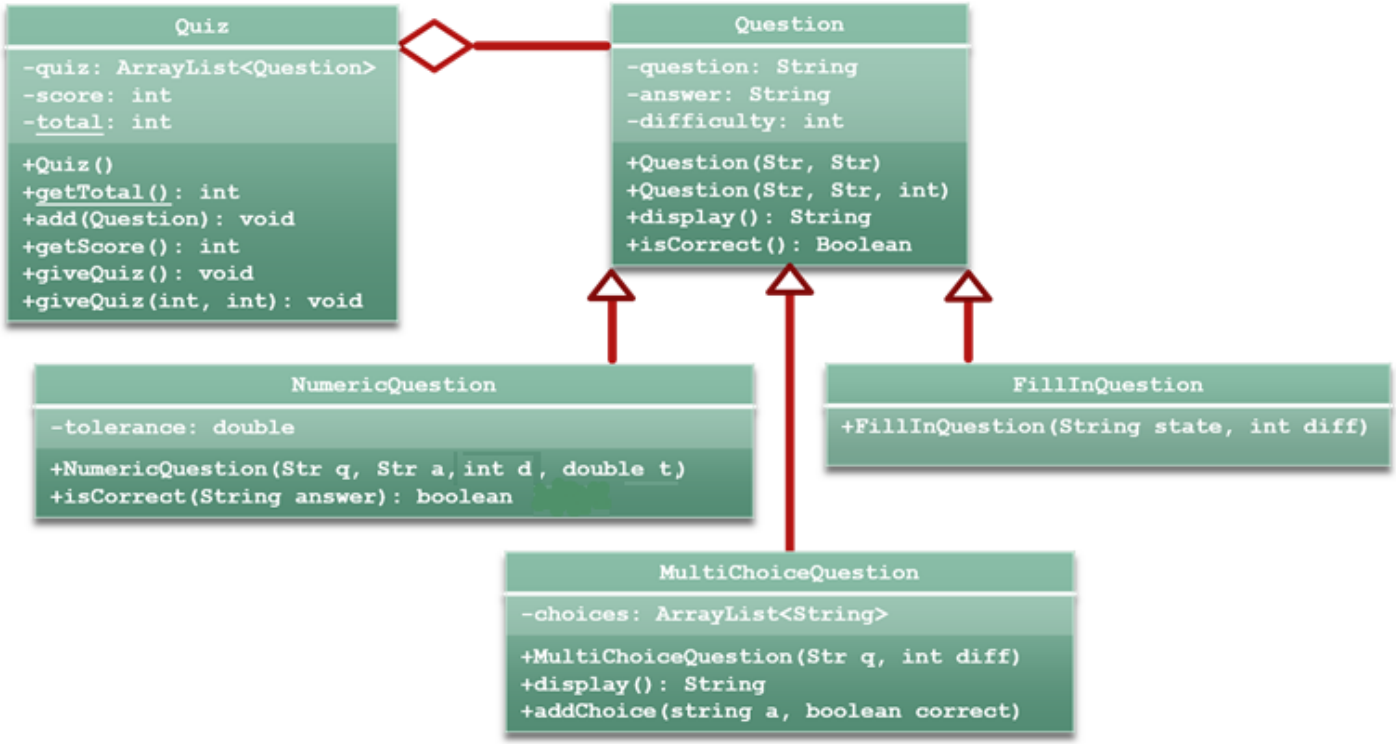
Submission:

- Zip up your NumericQuestion.java, MultiChoiceQuestion.java, FillInQuestion.java, Question.java, Quiz.java, and QuizTime.java and submit them to D2L.
- Unzipped submissions or submissions containing .class files will be automatically given ZERO

Exercise 1

In this lab you are going to be implementing the inheritance hierarchy shown below. You've already done Quiz and Question.

Modify your Question class to have a String display() method returns a string containing the question and difficulty of the current question. Modify Quiz to use this display method when displaying a question to the user. **You should not modify Question or Quiz in any other way.**



Exercise 2

Create a class called FillInQuestion that inherits from Question. This kind of question is created using a single string that contains the answer, surrounded by _ _. A sample question like this would be "The color _white_ is the most common color found on flags." The question should be displayed to the user as "The color _____ is the most common color found on flags."

Modify one of your quiz objects in QuizTime.java to contain some FillInQuestions.

CPSC 1181 - Lab 4 [40 marks]

Exercise 3
Create a class called NumericQuestion that is a child of Question. In this class, if the correct answer and the given answer differ by no more than some value, accept the response as correct. Refer to the UML diagram for more information about this class.
Modify one of your quiz objects in QuizTime.java to contain some NumericQuestions.

Exercise 4
Create a subclass of Question called MultiChoiceQuestion. This class should allow multiple correct choices as answers for a question.
MultiChoiceQuestion should contain a method that accepts and String and a boolean and adds a choice to the ArrayList of possible answers for the question.
This class should override the display method to show the user all of the possible answers for a question and give the user instructions on how to answer the question. See the UML diagram above and the sample output below for more information.
Modify one of your quiz objects in QuizTime.java to contain some MultiChoiceQuestions.

```
Violets are ____ (Difficulty: 1)
purple
Correct!
17.2 - 15.1 (Difficulty: 1)
2.1
Correct!
What are correct colors of ants? (Difficulty: 1)
1. black
2. green
3. orange
4. red
 Enter all correct choices. For example, if you think 1 and 2 are correct enter 12
14
Correct!
```

Marking Rubric:

Style, Convention, Documentation [5 marks]

Modifications to Question.java [2 marks]
Modifications to Quiz.java [1 marks]

NumericQuestion.java[8 marks]
+1 instance data
+2 constructor
+4 overrriding isCorrect correctly
+1 correctly parsing Strings to doubles

MultiChoice.java [10 marks]
+1 instance data
+2 constructor
+5 addChoice
+2 overriding display correctly
FillInQuestion.java [7 marks]
+1 inheriting
+6 constructor
QuizTime.java[7 marks]
+2 add FillInQs to QuizTime
+2 adding NumericQs to QuizTime
+3 adding MultiChoiceQs to QuizTime