**Objective:**

Dedicated and detail-oriented professional with an IT background seeking a Packaging Machine Operator position at PepsiCo. Committed to ensuring efficient production, maintaining quality standards, and contributing to a collaborative team environment.

**Professional Experience:**

**IT Support Specialist | XYZ Tech Solutions | 2018 – 2022**

Provided technical support to end-users, troubleshooting hardware and software issues.

* Collaborated with cross-functional teams to improve system performance and reliability.
* Developed strong problem-solving skills and attention to detail.

**Skills:Technical Skills:**

* + Familiarity with operating and troubleshooting machinery.
  + Quick learner when it comes to understanding new equipment.
  + Basic understanding of mechanical systems.
* **Communication:**
  + Effective communication skills for coordinating with team members and reporting issues promptly.
* **Quality Control:**
  + Attention to detail in ensuring products meet specifications.
* **Safety Awareness:**
  + Committed to following safety protocols and maintaining a safe work environment.

**Education:**

**Bachelor of Science in Computer Science | University of Technology | 2016 - 2020**

* Relevant coursework: Introduction to Engineering Systems, Basics of Mechanical Engineering.

**Certifications:**

* **Forklift Operator Certificate | Issuing Authority | 2021**
  + Demonstrates knowledge of equipment operation and safety procedures.

**Additional Information (Optional):**

* **Languages:** Proficient in English and Spanish.
* **Volunteer Work:** Participated in community service events, demonstrating teamwork and dedication.

**References:**

Available upon request.

Remember to customize this template further by adding any specific achievements or details related to your IT background that demonstrate transferable skills. Good luck with your application! 🌟

**## Maths Helper**

\_Please ensure you read and understand the entire document before undertaking this assignment. If you require

clarification on the specification within, please seek this as soon as possible.\_

**### Submission and Assessment**

The assignment comprises five tasks that each will require you to complete a series of steps to construct a Java program

that adheres to the described specification. Remember to analyse the problem as necessary and design a solution that

suits the deliverables. You need to determine the structures you will use. This is an application that will reuse

elements significantly and as such your development should consider this. Your program should also follow good code

writing practices, styles and conventions. This including aspects such as the appropriate use of indentation and

white-space, meaningful identifier names, suitable comments, minimal code duplication, privacy modifiers, etc.

At the conclusion of each task, you will be

asked to address several questions and report on the approach you have taken to complete the steps.

Your answers for each question should typically consist of one or two paragraphs.

Treat this as a professional activity, so in-line comments, documentation and application output should be presented appropriately for

the target audience. Furthermore, your report should be presented in a manner suitable for a professional audience.

You will be awarded marks for the level of completion of your program (including answers to the questions in each

section). This assignment will be marked out of a total of 100 marks as follows:

| TASK                |     | MARKS |

|:--------------------|----:|:-----:|

| Task 1              |   : |  15   |

| Task 2              |   : |  15   |

| Task 3              |   : |  10   |

| Task 4              |   : |  15   |

| Task 5              |   : |  10   |

| Task 6              |   : |  15   |

| Good Code Practices |   : |  20   |

Development should be consistent with the *\_IntelliJ\_* environment and Java version 17.

**### Submission**

Your program files must be submitted to the CodeGrade testing site before the due date listed on FLO.

CodeGrade will provide automatic test results, and you can resubmit your code any number of times before the due date.

You will also need to submit your report, as a single PDF document, to a separate submission link on Canvas by this same date

**### Academic Integrity**

A report on your submission will be generated by CodeGrade text-matching

software, which compares your work with a large body of material and generates a report of any similarities. Each

student should carefully ensure that the work they submit is their own.

**### Specification**

You have been approached by a local education company to provide a software solution to help primary school students with

their mathematics studies. The company requires an application that produces random arithmetic expressions tailored to

the level appropriate for the student.

This should be an application that is easy to use for students from Reception level to Year 7

(5 years old to 12 years old). It should be a text based application that presents an expression and allows the user to

type the answer.

The application will confirm if the supplied answer is correct or incorrect immediately and also keep track of the

overall percentage for the student. This overall percentage is reported at the end of the session with an appropriate

statement of feedback. The length of a session should be defined at the start of each session and can include 10, 20,

30, 40, or 50 questions.

The following issues should be considered at the start of each session, addressed by numbered menus:

- What is the level of the student?

    - **\*\*Reception\*\***: only addition

    - **\*\*Years 1 - 2\*\***: addition and subtraction

    - **\*\*Years 3 - 4\*\***: addition, subtraction, multiplication and division

    - **\*\*Years 5 - 6\*\***: addition, subtraction, multiplication and division

    - **\*\*Year 7\*\***: addition, subtraction, multiplication, division and modulo (division with remainder – presented as <samp>xry</samp>, where <samp>x</samp> is the result of the division and <samp>

      y</samp> is the remainder, ie <samp>10 % 3 = 3r1</samp>).

- What is the length of the session?

    - 10 questions

    - 20 questions

    - 30 questions

    - 40 questions

    - 50 questions

The year level that the user supplies will determine what number ranges are used.

| Range          | Year Level   |

|----------------|--------------|

| (0 → 9)        | At Reception |

| (0 → 9)        | Years 1 - 2  |

| (0 → 99)       | Years 3 - 4  |

| (-999 → 999)   | Years 5 - 6  |

| (-9999 → 9999) | At year 7    |

Example output could be as follows (user input in <mark>**\*\*bold highlighted\*\***</mark> text)

<pre>

+------------------------------------------------------------------------+

|                      Welcome to the Maths Helper.                      |

|       Use this application to test your knowledge of mathematics.      |

|      This program is intended for children from reception to year 7    |

+------------------------------------------------------------------------+

What is your year level? Choose an option from the list below:

+: addition, -: subtraction, \*: multiplication, /: division, %: division with remainder

[0] Reception [+]

[1] Year 1 [+, -]

[2] Year 2 [+, -]

[3] Year 3 [+, -, \*, /]

[4] Year 4 [+, -, \*, /]

[5] Year 5 [+, -, \*, /]

[6] Year 6 [+, -, \*, /]

[7] Year 7 [+, -, \*, /, %]

<mark><b>3</b></mark>

How many questions would you like to attempt? Choose an option from the list below:

[1] 10 questions

[2] 20 questions

[3] 30 questions

[4] 40 questions

[5] 50 questions

<mark><b>4</b></mark>

You are a Year 3 student and want to do 40 questions. Is this correct (Y/N)?: <mark><b>Y</b></mark>

Let's begin ... (press 'Q' at any time to quit)

</pre>

If the student indicates that the information entered was incorrect then they should be prompted to re-enter their

details. *\_Both menus should be displayed again until the user enters the correct options\_*.

A standard session will display a random expression (random numbers and random operator)

and wait for a user response to the

expression. Once an answer is provided, the program will assess the correctness of the input. A string indicating the correctness of the

student's answer will be displayed and then the next expression displayed. This process will continue

until all questions in the session have been answered.

Example output could be as follows (user input in <mark>**\*\*bold highlighted\*\***</mark> text):

<pre>

64 + 12 = <mark><b>76</b></mark>

Correct! Well Done!

Your current percentage is 100%

10 + 97 = <mark><b>107</b></mark>

Correct! Well Done!

Your current percentage is 100%

5 \* 39 = <mark><b>159</b></mark>

Bad luck that was incorrect. The correct answer was 195.

Your current percentage is 66.67%

22 / 58 = <mark><b>0</b></mark>

Correct! Well Done!

Your current percentage is 75%

...

Your total percentage was 72%.

Well done. That was a good effort.

Did you want to start a new Session or Quit (S/Q)? <mark><b>S</b></mark>

+------------------------------------------------------------------------+

|                      Welcome to the Maths Helper.                      |

|       Use this application to test your knowledge of mathematics.      |

|      This program is intended for children from reception to year 7    |

+------------------------------------------------------------------------+

What is your year level? Choose an option from the list below:

+: addition, -: subtraction, \*: multiplication, /: division, %: division with remainder

[0] Reception [+]

[1] Year 1 [+, -]

[2] Year 2 [+, -]

...

You are a Reception student and want to do 10 questions. Is this correct (Y/N)?: <mark><b>Y</b></mark>

Let's begin ... (press 'Q' at any time to quit)

9 + 2 = <mark><b>Q</b></mark>

</pre>

By entering the character '**\*\*H\*\***' the user is able to ask for a hint to be supplied. This hint should display the last

digit in the answer preceded by a '**\*\*-\*\***' for each extra digit in the answer. Any subsequent requests for a hint will

reveal the preceding digit. A student can continue to request help for the expression until all digits are revealed. If

this happens then this question should be graded as incorrect.

Example output could be as follows (user input in <mark>**\*\*bold highlighted\*\***</mark> text):

<pre>

84 \* 7 = <mark><b>H</b></mark>

84 \* 7 = --8: <mark><b>H</b></mark>

84 \* 7 = -88: <mark><b>588</b></mark>

Correct! Well Done!

Your current percentage is 100.00%

104 \* 58 = <mark><b>H</b></mark>

104 \* 58 = ---2: <mark><b>H</b></mark>

104 \* 58 = --32: <mark><b>H</b></mark>

104 \* 58 = -032: <mark><b>H</b></mark>

Bad luck that was incorrect. The correct answer was 6032.

Your current percentage is 66.67%

</pre>

**## Tasks**

While the steps described below should be completed sequentially, you should also pause during your development to test

recent changes and fix problems as they arise. Don't leave testing until you complete Task 6.

**### Task 1 – Initial Presentation**

Take some time to read through the existing code and the comments.

The program's main method is located inside the `MathsHelperDriver` class.

Running the provided program without any modifications or additions will result in the following output

<pre>

+------------------------------------------------------------------------+

|                      Welcome to the Maths Helper.                      |

|       Use this application to test your knowledge of mathematics.      |

|      This program is intended for children from reception to year 7    |

+------------------------------------------------------------------------+

What is your year level? Choose an option from the list below:

+: addition, -: subtraction, \*: multiplication, /: division, %: division with remainder

[0] Reception [+]

[1] Year 1 [+, -]

[2] Year 2 [+, -]

[3] Year 3 [+, -, \*, /]

[4] Year 4 [+, -, \*, /]

[5] Year 5 [+, -, \*, /]

[6] Year 6 [+, -, \*, /]

[7] Year 7 [+, -, \*, /, %]

How many questions would you like to attempt? Choose an option from the list below:

[1] 10 questions

[2] 20 questions

[3] 30 questions

[4] 40 questions

[5] 50 questions

</pre>

You will **\*\*not\*\*** need to change or add any code in the `MathsHelperDriver` class, however you will be adding code to the `MathsHelper`

class. You should see several methods defined int the <code>MathsHelper</code> class:

- **\*\*displayWelcome()\*\***

- **\*\*displayYearMenu()\*\***

- **\*\*displayQuestionMenu()\*\***

As well as these already complete methods, there are also methods that require further definition by

you.

- **\*\*letsPlay()\*\***

- **\*\*confirmSessionDetails(int year, int questions)\*\***

Your first task is to display the content to the user, capture their input and verify with the user that the information they

have provided is correct.

**### \*\*letsPlay() - Part 1\*\***

You will need to update the <samp>**\*\*letsPlay()\*\***</samp> method to capture the user's input after the year and question menus.

A <samp>**\*\*Scanner\*\***</samp> object <samp>**\*\*scan\*\***</samp> has already been defined for your needs.

Do not create any additional <samp>**\*\*Scanner\*\***</samp> objects.

An integer instance variable <samp>**\*\*yearLevel\*\***</samp> should be used to store the user's

response to the year menu and another integer instance variable <samp>**\*\*numQuestions\*\***</samp> used to store the user response to the

question menu.

Create getter and setter methods for accessing and modifying both of these instance variables.

After making these changes, example output could be as follows (user input in <mark>**\*\*bold highlighted\*\***</mark> text):

<pre>

+------------------------------------------------------------------------+

|                      Welcome to the Maths Helper.                      |

|       Use this application to test your knowledge of mathematics.      |

|      This program is intended for children from reception to year 7    |

+------------------------------------------------------------------------+

What is your year level? Choose an option from the list below:

+: addition, -: subtraction, \*: multiplication, /: division, %: division with remainder

[0] Reception [+]

[1] Year 1 [+, -]

[2] Year 2 [+, -]

[3] Year 3 [+, -, \*, /]

[4] Year 4 [+, -, \*, /]

[5] Year 5 [+, -, \*, /]

[6] Year 6 [+, -, \*, /]

[7] Year 7 [+, -, \*, /, %]

<mark><b>3</b></mark>

How many questions would you like to attempt? Choose an option from the list below:

[1] 10 questions

[2] 20 questions

[3] 30 questions

[4] 40 questions

[5] 50 questions

<mark><b>4</b></mark>

</pre>

Add a call to the <samp>**\*\*confirmSessionDetails()\*\***</samp> method at the bottom of the <samp>**\*\*letsPlay()\*\***</samp> method, where the two instance variables <samp>**\*\*yearLevel\*\***</samp> and <samp>**\*\*numQuestions\*\***</samp> are passed in as arguments.

**### \*\*confirmSessionDetails(int year, int questions)\*\***

The purpose of this method is to provide a confirmation message for the user after they have selected their year level and number of questions.

Note that the <samp>**\*\*confirmSessionDetails()\*\***</samp> method is a <samp>**\*\*boolean\*\***</samp> type method; meaning the

method is required to return a **\*\*true\*\*** or **\*\*false\*\*** value.

Within the

<samp>\*\*confirmSessionDetails()\*\*</samp> method you need to present the following information to the user:

<samp>You are a \_\_\_\_ student and want to do \_\_\_\_ questions. Is this correct (Y/N)?:

where the values for the *\_\_*\_\_ spaces need to be determined based on user input.

At the moment you are capturing two integers from the user and passing them to <samp>**\*\*confirmSessionDetails()\*\***</samp>.

Complete the following steps to fill in the necessary information for the prompt above:

- For the first *\_\_*\_\_, you need to analyse the **\*\*year\*\*** parameter and if it is equal to 0 then this should be 'Reception', otherwise

  it should be 'Year **\*\*year\*\***'.

- For the second *\_\_*\_\_, you need to analyse the **\*\*questions\*\*** parameter as a response to the question menu (i.e. 1 for 10 questions, 2 for 20 questions, 3 for 30 questions, etc.)

For example, calling the method using the statement <samp>**\*\*confirmSessionDetails(3,4)\*\***</samp> should print the following:

<samp>You are a Year 3 student and want to do 40 questions. Is this correct (Y/N)?:

Calling the method using the statement <samp>**\*\*confirmSessionDetails(0,2)\*\***</samp> should print the following:

<samp>You are a Reception student and want to do 20 questions. Is this correct (Y/N)?:

Finally, the <samp>**\*\*confirmSessionDetails()\*\***</samp> method needs to capture the user's input response to the above question.

If they supply either a '<samp>**\*\*Y\*\***</samp>' or '<samp>**\*\*y\*\***</samp>' then the method should return true, otherwise return false.

**### \*\*letsPlay() - Part 2\*\***

Now that we have completed the <samp>**\*\*confirmSessionDetails()\*\***</samp> method, we can call this from the <samp>**\*\*letsPlay()\*\***</samp>

method to verify that the user is happy with their selected input.

If the returned value from <samp>**\*\*confirmSessionDetails()\*\***</samp> is false, then our program needs to repeat the presentation

of both menus to the user and capture a fresh set of inputs for

<samp>\*\*yearLevel\*\*</samp> and <samp>\*\*numQuestions\*\*</samp>.

By following the above process you are locking the user into a loop until they provide a response indicating that they

have supplied the correct information (in this case either '<samp>\*\*Y\*\*</samp>' or '<samp>\*\*y\*\*</samp>').

Once the user has indicated that the information is correct, print the message <samp>Let's begin ... (press 'Q' at any time to quit)

At this point you should be able to present all the welcome details to the user, have them make selections from each

menu, and then confirm the information provided is correct.

Example output could be as follows (user input in <mark>**\*\*bold highlighted\*\***</mark> text):

<pre>

+------------------------------------------------------------------------+

|                      Welcome to the Maths Helper.                      |

|       Use this application to test your knowledge of mathematics.      |

|      This program is intended for children from reception to year 7    |

+------------------------------------------------------------------------+

What is your year level? Choose an option from the list below:

+: addition, -: subtraction, \*: multiplication, /: division, %: division with remainder

[0] Reception [+]

[1] Year 1 [+, -]

[2] Year 2 [+, -]

[3] Year 3 [+, -, \*, /]

[4] Year 4 [+, -, \*, /]

[5] Year 5 [+, -, \*, /]

[6] Year 6 [+, -, \*, /]

[7] Year 7 [+, -, \*, /, %]

<mark><b>0</b></mark>

How many questions would you like to attempt? Choose an option from the list below:

[1] 10 questions

[2] 20 questions

[3] 30 questions

[4] 40 questions

[5] 50 questions

<mark><b>2</b></mark>

You are a Reception student and want to do 20 questions. Is this correct (Y/N)?: N

What is your year level? Choose an option from the list below:

+: addition, -: subtraction, \*: multiplication, /: division, %: division with remainder

[0] Reception [+]

[1] Year 1 [+, -]

[2] Year 2 [+, -]

[3] Year 3 [+, -, \*, /]

[4] Year 4 [+, -, \*, /]

[5] Year 5 [+, -, \*, /]

[6] Year 6 [+, -, \*, /]

[7] Year 7 [+, -, \*, /, %]

<mark><b>1</b></mark>

How many questions would you like to attempt? Choose an option from the list below:

[1] 10 questions

[2] 20 questions

[3] 30 questions

[4] 40 questions

[5] 50 questions

<mark><b>5</b></mark>

You are a Year 1 student and want to do 50 questions. Is this correct (Y/N)?: Y

Let's begin ... (press 'Q' at any time to quit)

</pre>

---

**#### REPORT TASK**

Answer the following questions:

1. What issues (if any) did you have to fix in order to compile and run the provided code successfully?

2. What would happen in your program if the user provided invalid menu item values?

3. What approach could you employ to ensure that the user provides only valid menu item values?

---

**### Task 2 – Question Generator**

Your second task is to write the `QuestionGenerator` class that will create questions with the desired properties.

This class should define the following integer (int) instance variables:

- <samp>\*\*min\*\*</samp>

- <samp>\*\*max\*\*</samp>

The class should also define the following character (char) array instance variable:

- <samp>\*\*operations\*\*</samp>

Create getter and setter methods for accessing and modifying these instance variables.

You will need to implement the following three public static methods inside the `QuestionGenerator` class:

**### \*\*findMin(int year)\*\***

This method should return the minimum number for a value in our expressions, based on the year level of the user.

Looking at the specifications provided above you can see that all students from reception to year 4 have a minimum value of 0,

while students in year 5 and 6 have a minimum value of -999, and year 7 users have a minimum value of -9999. With this

knowledge you could create a collection of if statements or use a switch statement to get the right value for the year

level.

For example, calling this method using the statement:

<pre>

System.out.println(QuestionGenerator.findMin(3));

</pre>

Should give the following output:

<pre>

0

</pre>

**### \*\*findMax(int year)\*\***

This method should return the maximum number for a value in our expressions, based on the year level of the user.

This can be determined in a similar manner to the <samp>**\*\*findMin()\*\***</samp> method, by consulting the specifications above.

For example, calling this method using the statement:

<pre>

System.out.println(QuestionGenerator.findMax(5));

</pre>

Should give the following output:

<pre>

999

</pre>

**### \*\*findOperations(int year)\*\***

This method should return char[] array of possible operations for our expressions, based on the year level of the user.

This can be determined in a similar manner to the <samp>**\*\*findMin()\*\***</samp> and <samp>**\*\*findMax()\*\***</samp> methods, by consulting the specifications above.

For example, calling this method using the statement:

<pre>

System.out.println(QuestionGenerator.findOperations(6));

</pre>

Should give the following output:

<pre>

+-\*/

</pre>

**### \*\*QuestionGenerator(int year)\*\***

The constructor for this class should have a single parameter representing the user's year level.

Using calls to your newly created methods, initialise the min, max and operations instance variables of any new **\*\*QuestionGenerator\*\*** object based on this parameter.

After this has been completed, you should be able to create QuestionGenerator objects for a specific year level.

For example, calling the following statements:

<pre>

QuestionGenerator quiz1 = new QuestionGenerator(3);

QuestionGenerator quiz2 = new QuestionGenerator(7);

System.out.println(quiz1.getMin());

System.out.println(quiz1.getMax());

System.out.println(quiz1.getOperations());

System.out.println(quiz2.getMin());

System.out.println(quiz2.getMax());

System.out.println(quiz2.getOperations());

</pre>

Should give the following output:

<pre>

0

99

+-\*/

-9999

9999

+-\*/%

</pre>

---

**#### REPORT TASK**

Answer the following questions:

1. What would you need to change about your code so far if you wanted to add another year level to your program (e.g., year level 8).

2. If I wanted to create a QuestionGenerator object with a specific set of values for min, max and operations (rather than calculating them based on the year level of the student), how could you update your code to support this?

---

**### Task 3 – Random Questions**

Your third task is to write the `Question` class that represents a single question.

This class should define the following string (String) instance variables:

- <samp>\*\*question\*\*</samp>

- <samp>\*\*answer\*\*</samp>

Create getter and setter methods for accessing and modifying these instance variables.

**### \*\*Question(int min, int max, char[] operations)\*\***

The constructor for this class should have three parameters representing the min and max values for the expression used in the question, as well as a char array of possible operations.

Use these parameters to initialise the question and answer instance variables for this question.

The **\*\*question\*\*** instance variable should contain a string representing a random question of the form:

<samp>"\_\*\*value1\*\*\_ \_\*\*operator\*\*\_ \_\*\*value2\*\*\_ = "

Where *\_***\*\*value1\*\****\_* and *\_***\*\*value2\*\****\_* are random integers between min and max (inclusive), and *\_***\*\*operator\*\****\_* is a random operation from the operations array

For some operations, if *\_***\*\*value2\*\****\_* equals zero this can cause an exception to occur. For this reason, you should add an additional check to make sure that *\_***\*\*value2\*\****\_* cannot equal zero for any expression.

The **\*\*answer\*\*** instance variable should contain a string representing the answer to this question.

For divide (**\*\*/\*\***) expressions, the answer should be rounded to the nearest integer.

For modulo (**\*\*%\*\***) expressions, the answer should be given in the form **\*\*x\*\***r**\*\*y\*\***, where **\*\*x\*\*** is the result of the division and **\*\*y\*\*** is the remainder (ie. 10 % 3 = 3r1).

After this has been completed, you should be able to create Question objects.

For example, calling the following statements:

<pre>

Question question1 = new Question(0, 100, new char[]{'+','-'});

Question question2 = new Question(-30, -5, new char[]{'\*','/'});

Question question3 = new Question(200, 1000, new char[]{'%'});

System.out.println(question1.getQuestion());

System.out.println(question1.getAnswer());

System.out.println(question2.getQuestion());

System.out.println(question2.getAnswer());

System.out.println(question3.getQuestion());

System.out.println(question3.getAnswer());

</pre>

Could give the following output (depending on the random values selected):

<pre>

88 + 26 =

114

-26 / -17 =

2

828 % 691 =

1r137

</pre>

**### \*\*generateQuestions(int num)\*\***

Add a new instance variable to the `QuestionGenerator` class called **\*\*questions\*\*** that stores an ArrayList of Question objects.

This variable should be initialised as an empty ArrayList.

Create getter and setter methods for accessing and modifying this instance variable.

Add a void type method to the `QuestionsGenerator` class called <samp>**\*\*generateQuestions()\*\***</samp> that has a single integer parameter **\*\*num\*\***.

This method should populate the **\*\*questions\*\*** ArrayList with **\*\*num\*\*** new Question objects.

These questions should be created in accordance with the **\*\*min\*\***, **\*\*max\*\***, and **\*\*operations\*\*** instance variables of the `QuestionGenerator` class.

For example, calling the following statements:

<pre>

QuestionGenerator quiz1 = new QuestionGenerator(0);

quiz1.generateQuestions(5);

</pre>

Should populate the **\*\*questions\*\*** instance variable with 5 Question objects suitable for a Reception level student.

---

**#### REPORT TASK**

Answer the following questions:

1. How could you modify your program to always produce the same random questions every time the program is run, rather than different questions each time?

2. The company has notified you that questions should now be output in a slightly different format that includes a

   prefix identifying the question number in the format <samp>Q\_n\_.</samp> where <samp>*\_n\_*</samp>

   is the question number. For example, an addition question that is the first question in the set should appear as:

   <samp>Q1. 6 + 3 =</samp>

   What would you need to change in the code to accomplish this?

---

**### Task 4 – Asking Questions**

**### \*\*generateQuestions(QuestionGenerator quiz)\*\***

Add the following QuestionGenerator instance variable to the `MathsHelper` class:

- <samp>\*\*quiz\*\*</samp>

Create getter and setter methods for accessing and modifying this instance variable.

At the end of your <samp>**\*\*letsPlay()\*\***</samp> method in the `MathsHelper` class, create a QuestionGenerator object for the user's year level and assign it to the **\*\*quiz\*\*** instance variable.

Call the <samp>**\*\*generateQuestions()\*\***</samp> method on this instance variable to generate the required number of questions.

**### \*\*askQuestions()\*\***

Define a new public void type method to the MathsHelper class called <samp>**\*\*askQuestions()\*\***</samp> that takes no parameters.

This method should present each of the questions stored in the **\*\*questions\*\*** ArrayList for the **\*\*quiz\*\*** instance variable (i.e., every Question object in quiz.getQuestions()).

If the user's response to each question matches the stored answer for the question, you should print the message:

<samp>Correct! Well Done!

If the user's response does not match the stored answer for the question, you should print the message (with **\*\*correctAnswer\*\*** replaced by the correct answer to the question):

<samp>Bad luck that was incorrect. The correct answer was \*\*correctAnswer\*\*.

Regardless of whether the user's answer was correct or incorrect, you should also print what percentage of questions the user has gotten correct so far:

<samp>Your current percentage is \*\*calculatedPercentage\*\*

Where **\*\*calculatedPercentage\*\*** should be a percentage value rounded to a maximum of two decimal places.

Hint, you can calculate this percentage by keeping track of the number of right and wrong answers in the `QuestionGenerator` class.

An example snippet of output could be as follows (user input in <mark>**\*\*bold highlighted\*\***</mark> text):

<pre>

Let's begin ... (press 'Q' at any time to quit)

457 - 1338 = <mark><b>-881</b></mark>

Correct! Well Done!

Your current percentage is 100%

-2006 / -8281 = <mark><b>3</b></mark>

Bad luck that was incorrect. The correct answer was 0.

Your current percentage is 50%

-1441 + 3220 = <mark><b>1779</b></mark>

Correct! Well Done!

Your current percentage is 66.67%

5415 / -5301 = <mark><b>-1</b></mark>

Correct! Well Done!

Your current percentage is 75%

</pre>

**### \*\*Final Score\*\***

Once the user has answered all the questions, print their total percentage of correct questions in the following format:

<samp>Your total percentage was \*\*totalPercentage\*\*

Where **\*\*totalPercentage\*\*** should be a percentage value rounded to a maximum of two decimal places.

You should then print a message for the user based on the **\*\*totalPercentage\*\*** value according to the following pseudocode:

\_If totalPercentage is less than 40

&emsp;Print “Bad luck. Try practicing with some lower year levels to build your confidence and skills.”

If totalPercentage is between 40 and 50 (exclusive)

&emsp;Print “That was a good effort, but you may need to work on some expressions.”

If totalPercentage is between 50 and 60  (exclusive)

&emsp;Print “Congratulations you passed. Keep practicing at this year level to improve your score.”

If totalPercentage is between 60 and 75  (exclusive)

&emsp;Print “Well done. That was a good effort.”

If totalPercentage is between 75 and 85 (exclusive)

&emsp;Print “Good job. You should try the next year level in your next test.”

If totalPercentage is more than 85

&emsp;Print “Excellent work! You really know your stuff. Try the harder levels next time.”\_

An example snippet of output could be as follows (user input in <mark>**\*\*bold highlighted\*\***</mark> text):

<pre>

Let's begin ... (press 'Q' at any time to quit)

457 - 1338 = <mark><b>-881</b></mark>

Correct! Well Done!

Your current percentage is 100%

-2006 / -8281 = <mark><b>3</b></mark>

Bad luck that was incorrect. The correct answer was 0.

Your current percentage is 50%

-1441 + 3220 = <mark><b>1779</b></mark>

Correct! Well Done!

Your current percentage is 66.67%

5415 / -5301 = <mark><b>-1</b></mark>

Correct! Well Done!

Your current percentage is 75%

3963 - 7240 = <mark><b>-3277</b></mark>

Correct! Well Done!

Your current percentage is 80%

-8210 \* -8952 = <mark><b>5564345</b></mark>

Bad luck that was incorrect. The correct answer was 73495920.

Your current percentage is 66.67%

8938 / -5924 = <mark><b>-2</b></mark>

Correct! Well Done!

Your current percentage is 71.43%

-9649 \* -8535 = <mark><b>82354215</b></mark>

Correct! Well Done!

Your current percentage is 75%

-8385 + 3916 = <mark><b>4887</b></mark>

Bad luck that was incorrect. The correct answer was -4469.

Your current percentage is 66.67%

9852 \* 1267 = <mark><b>12482484</b></mark>

Correct! Well Done!

Your current percentage is 70%

Your total percentage was 70%

Well done. That was a good effort.

</pre>

Add a call to the new <samp>**\*\*askQuestions()\*\***</samp> method at the end of your <samp>**\*\*letsPlay()\*\***</samp> method

Running your program from the main method in the `MathsHelperDriver` class should now provide a basic version of the full program.

An example output could be as follows (user input in <mark>**\*\*bold highlighted\*\***</mark> text):

<pre>

+------------------------------------------------------------------------+

|                      Welcome to the Maths Helper.                      |

|       Use this application to test your knowledge of mathematics.      |

|      This program is intended for children from reception to year 7    |

+------------------------------------------------------------------------+

What is your year level? Choose an option from the list below:

+: addition, -: subtraction, \*: multiplication, /: division, %: division with remainder

[0] Reception [+]

[1] Year 1 [+, -]

[2] Year 2 [+, -]

[3] Year 3 [+, -, \*, /]

[4] Year 4 [+, -, \*, /]

[5] Year 5 [+, -, \*, /]

[6] Year 6 [+, -, \*, /]

[7] Year 7 [+, -, \*, /, %]

<mark><b>5</b></mark>

How many questions would you like to attempt? Choose an option from the list below:

[1] 10 questions

[2] 20 questions

[3] 30 questions

[4] 40 questions

[5] 50 questions

<mark><b>1</b></mark>

You are a Year 5 student and want to do 10 questions. Is this correct (Y/N)?: y

Let's begin ... (press 'Q' at any time to quit)

-410 - -801 = <mark><b>391</b></mark>

Correct! Well Done!

Your current percentage is 100%

-683 \* -916 = <mark><b>625628</b></mark>

Correct! Well Done!

Your current percentage is 100%

731 \* -813 = <mark><b>5943038</b></mark>

Bad luck that was incorrect. The correct answer was -594303.

Your current percentage is 66.67%

-159 + -459 = <mark><b>-618</b></mark>

Correct! Well Done!

Your current percentage is 75%

-339 + 766 = <mark><b>427</b></mark>

Correct! Well Done!

Your current percentage is 80%

925 + 968 = <mark><b>1893</b></mark>

Correct! Well Done!

Your current percentage is 83.33%

-724 + -918 = <mark><b>-1645</b></mark>

Bad luck that was incorrect. The correct answer was -1642.

Your current percentage is 71.43%

463 - 847 = <mark><b>-384</b></mark>

Correct! Well Done!

Your current percentage is 75%

713 + 643 = <mark><b>1356</b></mark>

Correct! Well Done!

Your current percentage is 77.78%

643 / -656 = <mark><b>-1</b></mark>

Correct! Well Done!

Your current percentage is 80%

Your total percentage was 80%

Good job. You should try the next year level in your next test.

</pre>

---

**#### REPORT TASK**

Answer the following questions:

1. Describe how you evaluated if the user's answer was correct in your askQuestions() method implementation.

2. Are there any answer inputs that the user could provide that would cause your program to break/crash?

---

**### Task 5 – Difficulty Suggestions**

For this task you should modify the program to respond to users who are either doing really well or really poorly.

After each block of five questions if the correct percentage of all questions so far is greater than 75% the program should display the message:

<samp>"You are doing really well! Maybe try a harder difficulty."</samp>

However, if the correct percentage is less than 30% then you should display the message:

<samp>"It seems you are having some trouble. Maybe try an easier difficulty."</samp>

You should also be aware that the year level cannot go above 7 or below 0, so the above checks should not be performed if the year level cannot go any higher or lower.

An example output could be as follows (user input in <mark>**\*\*bold highlighted\*\***</mark> text):

<pre>

+------------------------------------------------------------------------+

|                      Welcome to the Maths Helper.                      |

|       Use this application to test your knowledge of mathematics.      |

|      This program is intended for children from reception to year 7    |

+------------------------------------------------------------------------+

What is your year level? Choose an option from the list below:

+: addition, -: subtraction, \*: multiplication, /: division, %: division with remainder

[0] Reception [+]

[1] Year 1 [+, -]

[2] Year 2 [+, -]

[3] Year 3 [+, -, \*, /]

[4] Year 4 [+, -, \*, /]

[5] Year 5 [+, -, \*, /]

[6] Year 6 [+, -, \*, /]

[7] Year 7 [+, -, \*, /, %]

<mark><b>3</b></mark>

How many questions would you like to attempt? Choose an option from the list below:

[1] 10 questions

[2] 20 questions

[3] 30 questions

[4] 40 questions

[5] 50 questions

<mark><b>2</b></mark>

You are a Year 3 student and want to do 20 questions. Is this correct (Y/N)?: y

Let's begin ... (press 'Q' at any time to quit)

5 + 0 = <mark><b>5</b></mark>

Correct! Well Done!

Your current percentage is 100%

3 - 1 = <mark><b>2</b></mark>

Correct! Well Done!

Your current percentage is 100%

1 - 5 = <mark><b>-4</b></mark>

Correct! Well Done!

Your current percentage is 100%

7 + 7 = <mark><b>14</b></mark>

Correct! Well Done!

Your current percentage is 100%

3 + 4 = <mark><b>7</b></mark>

Correct! Well Done!

Your current percentage is 100%

You are doing really well! Maybe try a harder difficulty.

5 - 2 = <mark><b>3</b></mark>

Correct! Well Done!

Your current percentage is 100%

9 - 3 = <mark><b>6</b></mark>

Correct! Well Done!

Your current percentage is 100%

2 + 7 = <mark><b>7</b></mark>

Bad luck that was incorrect. The correct answer was 9.

Your current percentage is 87.5%

1 - 8 = <mark><b>8</b></mark>

Bad luck that was incorrect. The correct answer was -7.

Your current percentage is 77.78%

9 + 4 = <mark><b>13</b></mark>

Correct! Well Done!

Your current percentage is 80%

You are doing really well! Maybe try a harder difficulty.

12 - 17 = <mark><b>-8</b></mark>

Bad luck that was incorrect. The correct answer was -5.

Your current percentage is 72.73%

48 - 48 = <mark><b>55</b></mark>

Bad luck that was incorrect. The correct answer was 0.

Your current percentage is 66.67%

67 - 83 = <mark><b>32</b></mark>

Bad luck that was incorrect. The correct answer was -16.

Your current percentage is 61.54%

28 \* 96 = <mark><b>76</b></mark>

Bad luck that was incorrect. The correct answer was 2688.

Your current percentage is 57.14%

81 - 42 = <mark><b>99</b></mark>

Bad luck that was incorrect. The correct answer was 39.

Your current percentage is 53.33%

5 \* 35 = <mark><b>76</b></mark>

Bad luck that was incorrect. The correct answer was 175.

Your current percentage is 50%

71 \* 34 = <mark><b>54</b></mark>

Bad luck that was incorrect. The correct answer was 2414.

Your current percentage is 47.06%

28 + 66 = <mark><b>34</b></mark>

Bad luck that was incorrect. The correct answer was 94.

Your current percentage is 44.44%

7 + 51 = <mark><b>58</b></mark>

Correct! Well Done!

Your current percentage is 47.37%

43 \* 99 = <mark><b>77</b></mark>

Bad luck that was incorrect. The correct answer was 4257.

Your current percentage is 45%

Your total percentage was 45%

That was a good effort, but you may need to work on some expressions.

</pre>

---

**#### REPORT TASK**

Answer the following questions:

1. Describe how you checked whether you needed to print a suggestion message (i.e., how did you know that the user has just completed a block of 5 questions?)

2. Would it be easy to change the frequency of the check if required, e.g. after each block of 7 questions? Describe how this could be achieved.

---

**### Task 6 – Providing Hints and Quitting**

With this task you will implement a system that allows the user to type 'q' or 'Q' and quit the application completely.

They can also type 'h' or 'H' to receive a hint about the current question.

These user inputs will be caught at the same point where the user provides answers to each question.

**### \*\*Quitting\*\***

You should check whether the user has entered 'q' or 'Q' when answering a question to quit the program.

If they have, then you need to end your program.

Note, you should end your program by calling <samp>return;</samp> from the let's play method.

Do not use <samp>System.exit()</samp>.

After the use finishes answering all questions, they should be shown the following message and wait for an input:

<samp>Did you want to start a new Session or Quit (S/Q)?

If the user responds with 'q' or 'Q', then end the program.

If the user responds with 's' or 'S', then you should start the program again from the beginning.

if the user responds with any other input, then you should print the following message and wait for another input:

<samp>Sorry that input was not valid. Did you want to start a new Session or Quit (S/Q)?

For example (user input in <mark>**\*\*bold highlighted\*\***</mark> text):

<pre>

...

Well done. That was a good effort.

Did you want to start a new Session or Quit (S/Q)? <mark><b>Q</b></mark>

</pre>

For example (user input in <mark>**\*\*bold highlighted\*\***</mark> text):

<pre>

...

Well done. That was a good effort.

Did you want to start a new Session or Quit (S/Q)? <mark><b>S</b></mark>

+------------------------------------------------------------------------+

|                      Welcome to the Maths Helper!                      |

|       Use this application to test your knowledge of mathematics       |

|     This program is intended for children from reception to year 7     |

+------------------------------------------------------------------------+

What is your year level?

Choose an option from the list below:

(where: + = addition, - = subtraction, \* = multiplication, / = division, % = division with remainder)

[0] Reception [+]

[1] Year 1 [+, -]

[2] Year 2 [+, -]

[3] Year 3 [+, -, \*, /]

[4] Year 4 [+, -, \*, /]

...

</pre>

For example (user input in <mark>**\*\*bold highlighted\*\***</mark> text):

<pre>

...

Well done. That was a good effort.

Did you want to start a new Session or Quit (S/Q)? <mark><b>F</b></mark>

Sorry that input was not valid. Did you want to start a new Session or Quit (S/Q)? <mark><b>Q</b></mark>

</pre>

**### \*\*Hints\*\***

You should check whether the user has entered 'h' or 'H' when answering a question to request a hint.

If the user request a hint, you should reveal the last digit/character of the answer.

Any other digits/characters in the answer should be replaced by an underscore character ('\_').

Note, negative signs and remainder ('r') characters still count as digits/characters.

For example, given the following expression (user input in <mark>**\*\*bold highlighted\*\***</mark> text):

<pre>

-6804 + 2820 = <mark><b>h</b></mark>

</pre>

As the user supplied '**\*\*h\*\***' for their answer (requested a hint) then the expression should be restructured and display:

<pre>

-6804 + 2820 = \_\_\_\_4:

</pre>

The program should then wait for the user to supply a new input response:

For example (user input in <mark>**\*\*bold highlighted\*\***</mark> text):

<pre>

-6804 + 2820 = <mark><b>h</b></mark>

-6804 + 2820 = \_\_\_\_4: <mark><b>-3984</b></mark>

Correct! Well Done!

Your current percentage is 100%

</pre>

**### \*\*Multiple Hints\*\***

Subsequent hints for the same question, after the first hint has already been requested, should reveal another digit/character of the answer (user input in <mark>**\*\*bold highlighted\*\***</mark> text):

<pre>

-6804 + 2820 = \_\_\_\_4: <mark><b>h</b></mark>

-6804 + 2820 = \_\_\_84: <mark><b>h</b></mark>

-6804 + 2820 = \_\_984: <mark><b>-3984</b></mark>

Correct! Well Done!

Your current percentage is 100%

</pre>

If the user requests too many hints (i.e. a hint for every digit/character in the answer) then the answer should be registered as

incorrect.

For example (user input in <mark>**\*\*bold highlighted\*\***</mark> text):

<pre>

-6804 + 2820 = \_\_\_\_4: <mark><b>h</b></mark>

-6804 + 2820 = \_\_\_84: <mark><b>h</b></mark>

-6804 + 2820 = \_\_984: <mark><b>h</b></mark>

-6804 + 2820 = \_3984: <mark><b>h</b></mark>

Bad luck that was incorrect. The correct answer was -3984

Your current percentage is 0%

</pre>

---

**#### REPORT TASK**

Answer the following question:

1. How could you update your program so that correctly answering a question that a hint was requested on was only worth half as many marks.

2. The company has asked you to write a Graphical User Interface for the program you have written. Could it be used as is, or would you need to rewrite parts of the code to accomplish this? Discuss and explain.

---

**### \*\*Final Program\*\***

Your Maths Helper program is now complete!

A full example run-through of your completed program could look as follows (user input in <mark>**\*\*bold highlighted\*\***</mark> text):

<pre>

+------------------------------------------------------------------------+

|                      Welcome to the Maths Helper.                      |

|       Use this application to test your knowledge of mathematics.      |

|      This program is intended for children from reception to year 7    |

+------------------------------------------------------------------------+

What is your year level? Choose an option from the list below:

+: addition, -: subtraction, \*: multiplication, /: division, %: division with remainder

[0] Reception [+]

[1] Year 1 [+, -]

[2] Year 2 [+, -]

[3] Year 3 [+, -, \*, /]

[4] Year 4 [+, -, \*, /]

[5] Year 5 [+, -, \*, /]

[6] Year 6 [+, -, \*, /]

[7] Year 7 [+, -, \*, /, %]

<mark><b>6</b></mark>

How many questions would you like to attempt? Choose an option from the list below:

[1] 10 questions

[2] 20 questions

[3] 30 questions

[4] 40 questions

[5] 50 questions

<mark><b>2</b></mark>

You are a Year 6 student and want to do 20 questions. Is this correct (Y/N)?: <mark><b>N</b></mark>

What is your year level? Choose an option from the list below:

+: addition, -: subtraction, \*: multiplication, /: division, %: division with remainder

[0] Reception [+]

[1] Year 1 [+, -]

[2] Year 2 [+, -]

[3] Year 3 [+, -, \*, /]

[4] Year 4 [+, -, \*, /]

[5] Year 5 [+, -, \*, /]

[6] Year 6 [+, -, \*, /]

[7] Year 7 [+, -, \*, /, %]

<mark><b>7</b></mark>

How many questions would you like to attempt? Choose an option from the list below:

[1] 10 questions

[2] 20 questions

[3] 30 questions

[4] 40 questions

[5] 50 questions

<mark><b>1</b></mark>

You are a Year 7 student and want to do 10 questions. Is this correct (Y/N)?: <mark><b>y</b></mark>

Let's begin ... (press 'Q' at any time to quit)

5229 % 8792 = <mark><b>0r5229</b></mark>

Correct! Well Done!

Your current percentage is 100%

-6158 % -5927 = <mark><b>h</b></mark>

-6158 % -5927 = \_\_\_\_\_1: <mark><b>-231</b></mark>

Bad luck that was incorrect. The correct answer was 1r-231.

Your current percentage is 50%

-837 + 5938 = <mark><b>5101</b></mark>

Correct! Well Done!

Your current percentage is 66.67%

-568 - -5090 = <mark><b>4522</b></mark>

Correct! Well Done!

Your current percentage is 75%

-6959 \* 2723 = <mark><b>-18949357</b></mark>

Correct! Well Done!

Your current percentage is 80%

3970 % -1714 = <mark><b>h</b></mark>

3970 % -1714 = \_\_\_\_\_2: <mark><b>h</b></mark>

3970 % -1714 = \_\_\_\_42: <mark><b>h</b></mark>

3970 % -1714 = \_\_\_542: <mark><b>h</b></mark>

3970 % -1714 = \_\_r542: <mark><b>h</b></mark>

3970 % -1714 = \_2r542: <mark><b>h</b></mark>

Bad luck that was incorrect. The correct answer was -2r542.

Your current percentage is 66.67%

-9521 % 4990 = <mark><b>459</b></mark>

Bad luck that was incorrect. The correct answer was -1r-4531.

Your current percentage is 57.14%

5812 % 7749 = <mark><b>0r5812</b></mark>

Correct! Well Done!

Your current percentage is 62.5%

1433 / 6862 = <mark><b>h</b></mark>

Bad luck that was incorrect. The correct answer was 0.

Your current percentage is 55.56%

239 - -6613 = <mark><b>6852</b></mark>

Correct! Well Done!

Your current percentage is 60%

Your total percentage was 60%

Well done. That was a good effort.

Did you want to start a new Session or Quit (S/Q)? <mark><b>Q</b></mark>

</pre>