

Exercise - Calculate the average of the student's assignment scores

10 minutes

In this exercise, you'll calculate and store the average of the assignment scores from each student. Since you know the number of scored assignments for each student, the average is calculated by dividing the sum of the assignment scores by the number of assignments. To store the averages, you'll be using the Decimal data type.

Create variables to store the average

In this task, you'll create a variable for each student that can be used to store the average score for their graded assignments.

1. In the .NET Editor, locate the `Console.WriteLine()` statements that are used to display each student's summed scores.
2. Create a blank code line above the `Console.WriteLine()` statements.
3. On the blank code line that you created, to declare the Decimal variables that will be used for the students current scores, enter the following code:

C#

```
decimal sophiaScore;  
decimal nicolasScore;  
decimal zahirahScore;  
decimal jeongScore;
```

Notice that you're only declaring the `decimal` variables and not initializing them. You chose the `decimal` type because you're representing an average grade and want to include a decimal place that wouldn't be available if you used an integer. This way, you can see if a student made a score of 89.9 and bump them up from a B to an A.

In the previous exercise, you initialized the integer variables so that you could immediately use them in your console output. In this case, these `decimal` variables will be initialized in the next step using calculations with your existing data, starting with Sophia's score.

4. To assign Sophia's current score in the class to the decimal `sophiaScore`, update the variable with the following code:

C#

```
decimal sophiaScore = sophiaSum / currentAssignments;
```

To calculate a student's current score for the class, you divide the sum of assignment scores by the total number of assignments. Each student in the class has five assignments, represented by `currentAssignments` that you initialized during setup.

5. To assign the rest of the students' scores, enter the following code:

C#

```
decimal nicolasScore = nicolasSum / currentAssignments;  
decimal zahirahScore = zahirahSum / currentAssignments;  
decimal jeongScore = jeongSum / currentAssignments;
```

Ultimately, you want to display the grades of each student in this application. In the next step, you'll modify the code to display each student's score rather than their assignment sum.

6. To display each student's current score, replace the sum variables in the display statements with the score variables:

C#

```
Console.WriteLine("Sophia: " + sophiaScore);  
Console.WriteLine("Nicolas: " + nicolasScore);  
Console.WriteLine("Zahirah: " + zahirahScore);  
Console.WriteLine("Jeong: " + jeongScore);
```

7. Take a minute to consider the incremental approach you're using to develop this application.

Breaking down a problem into smaller pieces is an important skill for developers. Building your code incrementally and checking your work frequently allows you to quickly develop reliable applications. In this case, you're able to repurpose `Console.WriteLine()` to verify that your calculations are correct as you complete each stage in the process.

8. To view the values of each student's current grade, select **Run**.

You should see the following output:

Output

```
Sophia: 94  
Nicolas: 83  
Zahirah: 83  
Jeong: 95
```

9. Notice that the scores are displayed as integers rather than decimals.

When you want the result of a division calculation to be a decimal value, either the dividend or divisor must be of type decimal (or both). When you use integer variables in the calculation, you need to apply a technique known as casting to "convert" an integer to a decimal.

For the score calculation, you can obtain a decimal result by casting either the sum variable or `currentAssignments` as a decimal type. In this case, you'll cast the sum as a decimal.

10. In your division operations, to cast an integer variable as a decimal, update your code as follows:

C#

```
decimal sophiaScore = (decimal) sophiaSum / currentAssignments;  
decimal nicolasScore = (decimal) nicolasSum / currentAssignments;  
decimal zahirahScore = (decimal) zahirahSum / currentAssignments;  
decimal jeongScore = (decimal) jeongSum / currentAssignments;
```

You only need the dividend or divisor to be a decimal type for the division to result in a decimal value. Here, you cast only the sum variable that's used as the dividend.

11. Review the following grading scale that the teacher uses to assign letter grades:

Output

97 - 100	A+
93 - 96	A
90 - 92	A-
87 - 89	B+
83 - 86	B

The next step is to include a letter grade for each student based on their total score. Adding the letter grade to the displayed output will be a manual process.

12. To determine the value of each student's current grade, select **Run**.

Use the current grade for each student to determine the appropriate letter grade, rounding up or down as necessary.

13. To append letter grade after each student's numeric score, update your code as follows:

C#

```
Console.WriteLine("Sophia: " + sophiaScore + " A");  
Console.WriteLine("Nicolas: " + nicolasScore + " B");  
Console.WriteLine("Zahirah: " + zahirahScore + " B");  
Console.WriteLine("Jeong: " + jeongScore + " A");
```

Check Your Work

In this task, you'll run the code and verify that the output is correct.

1. Compare your code to the following:

C#

```
int currentAssignments = 5;  
  
int sophia1 = 93;  
int sophia2 = 87;  
int sophia3 = 98;  
int sophia4 = 95;  
int sophia5 = 100;  
  
int nicolas1 = 80;  
int nicolas2 = 83;  
int nicolas3 = 82;  
int nicolas4 = 88;  
int nicolas5 = 85;
```

```
int zahirah1 = 84;
int zahirah2 = 96;
int zahirah3 = 73;
int zahirah4 = 85;
int zahirah5 = 79;

int jeong1 = 90;
int jeong2 = 92;
int jeong3 = 98;
int jeong4 = 100;
int jeong5 = 97;

int sophiaSum = sophia1 + sophia2 + sophia3 + sophia4 + sophia5;
int nicolasSum = nicolas1 + nicolas2 + nicolas3 + nicolas4 + nicolas5;
int zahirahSum = zahirah1 + zahirah2 + zahirah3 + zahirah4 + zahirah5;
int jeongSum = jeong1 + jeong2 + jeong3 + jeong4 + jeong5;

decimal sophiaScore = (decimal) sophiaSum / currentAssignments;
decimal nicolasScore = (decimal) nicolasSum / currentAssignments;
decimal zahirahScore = (decimal) zahirahSum / currentAssignments;
decimal jeongScore = (decimal) jeongSum / currentAssignments;

Console.WriteLine("Sophia: " + sophiaScore + " A");
Console.WriteLine("Nicolas: " + nicolasScore + " B");
Console.WriteLine("Zahirah: " + zahirahScore + " B");
Console.WriteLine("Jeong: " + jeongScore + " A");
```

2. To run your code and display the student scores with letter grades, select **Run**.
3. To verify that your code is working as expected, compare the output of your application with the following output:

Output

```
Sophia: 94.6 A
Nicolas: 83.6 B
Zahirah: 83.4 B
Jeong: 95.4 A
```

Your code should now be displaying the student scores as a decimal value, and you should see the letter grade that you assigned.

If your code displays different results, you'll need to review your code to find your error and make updates. Run the code again to see if you've fixed the problem. Continue updating and running your code until your code produces the expected results.

Important

Be sure not to delete any of the code you've written so far. You'll build on this code in the next exercise.

Module complete:

[Unlock achievement](#)[English \(United States\)](#)[✔ × Your Privacy Choices](#)[⚙ Theme ▾](#)[Manage cookies](#)[Previous Versions](#)[Blog ¹²](#)[Contribute](#)[Privacy ¹²](#)[Terms of Use](#)[Trademarks ¹²](#)

© Microsoft 2023

.NET Editor

Press **CTRL** + **M**, **TAB** to exit the editor

[🗑 Clear](#)[▶ Run](#)[ⓘ](#)

1

Output	