Select at least **two** of the following topics on Java methods for your initial post. Provide a code example where necessary to elaborate your thoughts.

* Creating a Class vs. Creating an Object
* Multiple Objects Instances
* Working with Multiple Classes
* Accessing and Modifying Attributes
* Java Library Classes
* Class Methods

In Java, we have the option to create a class and create an object. Creating a class and creating an object go hand in hand with each other. The way to create a class is to include the keyword “Main.” The class established earlier is used to create an object with a specified object name and the word new. An object has assigned attributes, while the class is more of a structure. A class should be declared before an object or object. When creating multiple classes within a file, only one of them can be a public class (Liang, 2019). It must also have the name of the file (Liang, 2019).

Class methods are also utilized in Java. According to Liang (2019), a class “defines the properties of objects and provides constructors for creating objects and methods for manipulating them.” It also serves as a reference for object variables. A method is declared in a class and then operates as coded. Class methods can be modified to fit the desired accessibility level for users. Having options allows for different levels of security, so it is important to be mindful of which visibility modifier to use. When writing out methods, two possibilities are static or public. If a static method is utilized, then a class object is not needed but is when it is public. When it is public, an instance of a class is needed to call the method. Creating class methods can be helpful since you can use multiple classes.

**Reference**

Liang, Y. D. (2019). *Introduction to Java programming and data structures: comprehensive version*. Pearson. https://plus.pearson.com/home?utm\_source=ereader

**Assignment Requirements and Grading:**

* An initial post of approximately 250 words is due by **Thursday, 11:59 p.m., CST**.
* Submit your post by clicking on the assignment link above, then Create Thread. You must create a thread in order to view your peers' posts. Tip: Create your post in a Word document and then copy and paste your work into the thread.
* A minimum of three (3) responses, to the original threads of other students, of 100-200 words each are due by **Sunday, 11:59 p.m., CST**.
* To view the rubric grading criteria, click on the following link: [Discussion Board Grading Rubric](https://content.bellevue.edu/cst/csd/rubricdbv3.pdf).

Jessica, you did an excellent job on your post and explaining the differences between creating a class versus creating an object. You are spot on when you say that a class does nothing until an object is made from it. A class needs an instance for it to work. Your example for the class and objects perfectly capture your explanation. You also did a good job of explaining how to access and modify attributes. I am glad that you mentioned that Java only allows direct access to attributes when they are public. This is an important aspect to remember when writing a Java program.

Miles, you did a fantastic job on your post for this module. GeeksforGeeks really did provide a great explanation of creating a class vs. creating an object. I like to reference it when I feel like I need an additional understanding of a topic. Sometimes seeking other resources can be a great aid when feeling confused. I am glad that you mentioned that a class is simply a framework. I liked your example of a laptop as a class. Without all the other objects, it cannot function on its own. I like how you further elaborated on it in your second topic embellishment.

Anton, your discussion post for this module is very nicely said and explained! Java truly is a rich ecosystem! I have really been enjoying all the built-in functions and library classes available. Building on code other developers have already established is a great time saver and makes coding and the code more efficient. It is important to remember to import it into the code when using a package. Performing math would be much more difficult without packages. You are spot on that methods define the behavior of classes and objects.