1. **Provide a definition for C# properties.**

C# properties are similar to regular data fields but allow the programmer to access private data without calling obtrusive get/set methods, by streaming-lining the same methods into a ‘property’.

1. **Explain how to use/activate an instance method.**

According to the book, “To call an instance method, however, an object must be instantiated and associated with the method.” – basically using the Class name, followed by a variable for the instantiation, and then use the new keyword and then a call to the constructor.

MyClass myVariableName = new MyClass();

Or if there are arguments for the constructor,

MyClass myVariableName = new MyClass(arg1, arg2);

1. **What are the differences between mutator and accessor methods?**

Mutator methods modify private data, while accessor methods retrieve private data.

1. **How do mutator/accessor methods differ from properties?**

Mutator/accessor methods are method calls to the class that contains them, while properties can be used like regular variables, but man-in-the-middle functions in the class do their work behind the scenes when called.

1. **Provide definitions for the public and private accessibility modifiers.**

Public accessibility modifiers are accessible to any class, at any point in the object.

Private accessibility modifiers are only accessible to the class that created it, so the memory area is protected from bugs/hackers/security flaws.