C# Chapter 3

1. What is a *method*?

It is a named sequence of statements (similar to a function). A method can be given some data for processing and can return information.

2. What does a *return statement* do?

It causes the method to finish. If one does not want the method to return information (void), one can use a variation of the return statement to cause an immediate exit from the method. Write the word return and follow it immediately by a semicolon.

3. What is an *expression-bodied method*?

It is a syntactic convenience. The main difference is the use of *=>* operator to reference the expression that forms the body of the method and the absence of a *return* statement.

4. What is *a scope of a variable*?

One cannot use the scope of a variable to share information between methods.

5. What is an *overloaded method*?

An overloaded method is when two identifiers have the same name and are declared in the same scope.

6. How do you call a method that requires arguments?

Supply a comma-separated list of arguments, and the number and type of the arguments are used by the compiler to select one of the overloaded methods.

7. How do you write a method, that is, specify the method definition that requires a parameter list?

8. How do you specify a parameter as optional when defining a method?

When defining a method by providing a default value for the parameter. Indicate a default value by using the assignment operator.

9. How do you pass an argument to a method as a named parameter?

Specify the name of the parameter, followed by a colon and the value to use.