**Chapter 6: How to code methods and event handlers**

**Murach's C#**

**MULTIPLE CHOICE**

1. What will the following method return?

**private string GetName(int customerID, bool readOnly)**

| a. | an int value | c. | a bool value |
| --- | --- | --- | --- |
| b. | a string | d. | a name value |

2. What keyword do you code at the beginning of a method if you only want it to be available within the current class?

| a. | void | c. | private |
| --- | --- | --- | --- |
| b. | public | d. | return |

3. What keyword do you code for the return type of a method that doesn’t return any data?

| a. | void | c. | ref |
| --- | --- | --- | --- |
| b. | null | d. | return |

4. For each parameter in the parameter list for a method, you must code

| a. | the name of the parameter followed by the data type of the parameter |
| --- | --- |
| b. | the data type of the parameter followed by the name of the parameter |
| c. | the name of the parameter only |
| d. | the data type of the parameter only |

5. When you call a method, the arguments you pass to it

| a. | must be coded in the same sequence as the parameters |
| --- | --- |
| b. | must be declared with data types that are compatible with the parameters |
| c. | must have the same names as the parameters |
| d. | must be coded in the same sequence as the parameters and have data types that are compatible with the parameters |

6. The signature of a method is formed by the

| a. | name of the method only |
| --- | --- |
| b. | name of the method and its return type |
| c. | name of the method and its parameter list |
| d. | name of the method, its return type, and its parameter list |

7. If you pass a variable by reference, the called method

| a. | can’t change the name of the variable in the calling method |
| --- | --- |
| b. | can change the name of the variable in the calling method |
| c. | can’t change the value of the variable in the calling method |
| d. | can change the value of the variable in the calling method |

8. If you pass a variable by value, the called method

| a. | can’t change the name of the variable in the calling method |
| --- | --- |
| b. | can change the name of the variable in the calling method |
| c. | can’t change the value of the variable in the calling method |
| d. | can change the value of the variable in the calling method |

9. If you declare a parameter for a method as optional,

| a. | the user must not provide a value for the parameter |
| --- | --- |
| b. | the parameter must be assigned a default value |
| c. | it must be declared before any required parameters |
| d. | the value for the parameter must be passed by name |

10. Which of the following code snippets passes an argument by name?

| a. | **subtotal:subtotal** | c. | **subtotal=subtotal** |
| --- | --- | --- | --- |
| b. | **subtotal:=subtotal** | d. | **subtotal=:subtotal** |

11. Which of the following is *not* an advantage of passing arguments by name?

| a. | You don’t have to indicate if an optional parameter has been omitted |
| --- | --- |
| b. | You don’t have to know the name of the associated parameter |
| c. | You can pass some arguments by name even if other arguments are passed by position |
| d. | You can pass the arguments in any order |

12. To generate an event handler for a control event, you can display the Events list for the control and then

| a. | select a name from the drop-down list for the event |
| --- | --- |
| b. | click on the event |
| c. | double-click on the event |
| d. | right-click on the event and select the Generate Handler command |

13. Once you display the list of events for a control, you can wire an existing event handler to any event by

| a. | selecting the event handler from the drop-down list for the event |
| --- | --- |
| b. | clicking on the name of the event and selecting the event handler |
| c. | double-clicking on the name of the event and selecting the event handler |
| d. | right-clicking on the name of the event and selecting the event handler |

14. To generate a method call and a method from existing code, you can use a feature called

| a. | IntelliSense | c. | refactoring |
| --- | --- | --- | --- |
| b. | code snippets | d. | debugging |

15. When coding an expression-bodied method, what operator do you code after the method signature?

| a. | assignment operator (**=**) | c. | lambda operator (**=>**) |
| --- | --- | --- | --- |
| b. | equality operator (**==**) | d. | conditional operator (**?:**) |

16. To define a tuple as the return type for a method, you define its members by separating them with commas and enclosing them in

| a. | parentheses - **()** | c. | brackets - **[]** |
| --- | --- | --- | --- |
| b. | braces - **{}** | d. | angle brackets - **<>** |

17. To delete an event handler, you not only have to delete its method but also its

| a. | signature | c. | tuple |
| --- | --- | --- | --- |
| b. | event wiring | d. | button |

18. If you have an int variable named yrs and two decimal variables named prin and rate, which statement can you use to call the method with the following method declaration?

**private decimal GetInterest(int years,**

**decimal interestRate, decimal principle)**

| a. | **decimal interest = GetInterest(yrs, rate, prin);** |
| --- | --- |
| b. | **decimal interest = GetInterest(rate, prin, yrs);** |
| c. | **int interest = GetInterest(yrs, rate, prin);** |
| d. | **int interest = GetInterest(rate, prin, yrs);** |

19. If the following GetInterest() method uses a decimal variable named interest to store the interest amount, which statement can you use to return the interest amount?

**private decimal GetInterest(int years,**

**decimal interestRate, decimal principle)**

| a. | **Return Interest;** |
| --- | --- |
| b. | **return dec interest;** |
| c. | **return decimal interest;** |
| d. | **return interest;** |

20. Which of the following statements would you use to call a private method named InitializeVariables() that accepts no parameters and doesn’t return a value?

| a. | **InitializeVariables(void);** |
| --- | --- |
| b. | **void InitializeVariables();** |
| c. | **InitializeVariables();** |
| d. | **void InitializeVariables(void);** |

21. Which of the following statements would you use to pass a variable named message by reference to a method named DisplayMessage?

| a. | **DisplayMessage(reference message);** |
| --- | --- |
| b. | **DisplayMessage(ref message);** |
| c. | **DisplayMessage(message reference);** |
| d. | **DisplayMessage(message ref);** |
| e. | **DisplayMessage(message);** |

22. Which statement calls a method in the current form named SetButtons() and passes it a value of false?

| a. | **this.SetButtons(false);** |
| --- | --- |
| b. | **this.SetButtons(disable);** |
| c. | **this.SetButtons(bool false);** |
| d. | **this.SetButtons(value=false);** |

23. Which of the following declares a method named GetMessage() that returns a string value and requires one decimal parameter named currentBalance and is only available within the current form?

| a. | **private void GetMessage(currentBalance)** |
| --- | --- |
| b. | **private string GetMessage(currentBalance)** |
| c. | **private string GetMessage(decimal currentBalance)** |
| d. | **protected string GetMessage(decimal currentBalance)** |

24. Which of the following declares a private method named GetMessage() that returns a string value and requires a string parameter named fullName and a decimal parameter named currentBalance?

| a. | **private string GetMessage(fullName : currentBalance)** |
| --- | --- |
| b. | **private string GetMessage(fullName, currentBalance)** |
| c. | **private string GetMessage(string fullName : decimal currentBalance)** |
| d. | **private string GetMessage(string fullName, decimal currentBalance)** |

25. Which of the following is a shorter way to code the following GetTotal() method?

**private decimal GetTotal(decimal subtotal, decimal tax)**

**{**

**return subtotal + tax;**

**}**

| a. | **private decimal GetTotal(subtotal, tax) { return subtotal + tax }** |
| --- | --- |
| b. | **private decimal GetTotal(decimal subtotal, decimal tax)**  **{ subtotal + tax }** |
| c. | **private decimal GetTotal(decimal subtotal, decimal tax)**  **return subtotal + tax;** |
| d. | **private decimal GetTotal(decimal subtotal, decimal tax) =>**  **subtotal + tax;** |

26. Which of the following declares a method named GetTotals() that returns a tuple that has two members named TotalQty and TotalAmt?

| a. | **private [int TotalQty, decimal TotalAmt] GetTotals(int year) {}** |
| --- | --- |
| b. | **private (int TotalQty, decimal TotalAmt) GetTotals(int year) {}** |
| c. | **private [TotalQty, TotalAmt] GetTotals(int year) {}** |
| d. | **private (TotalQty, TotalAmt) GetTotals(int year) {}** |

27. Which of the following statements would be used to wire the Click event of a button named btnClear to an event handler named ClearControls()?

| a. | **btnClear.Click += System.EventHandler(ClearControls);** |
| --- | --- |
| b. | **btnClear.Click += ClearControls;** |
| c. | **ClearControls += System.EventHandler(btnClear.Click);** |
| d. | **ClearControls += btnClear.Click;** |