1. C# is a case-sensitive language.(True/False)
2. C# can be referred to as strongly-typed programming language.(True/False)
3. What is the difference between instance and static variables?  
   What is the scope of Local Variables?
4. What do you mean by Primitive Data Types? How many primitive data types are there in C#?
5. What is byte in data types?
6. What is the syntax of declaring a Boolean variable?
7. Why do you use underscore Character in Numeric Literals?
8. What is enumeration?
9. The \_\_\_\_\_\_\_\_\_\_\_\_ provides you with an organized view of your projects and their files in c#
10. Why do we use **placeholder** method while printing a value on screen?
11. What is the difference between implicit and explicit type casting? Give an example to clear your point.
12. \_\_\_\_\_ class and its \_\_\_\_\_\_ method allows the user to read or accept values of various data types from the keyboard.
13. What is the difference between if/else ladder and multiple ifs?
14. What is the difference between switch case and if/else ladder?
15. Which Logical operator works in a similar method as Nested if?
16. What is the use of **break** in switch case? Does it cause any error if you don’t use break statement.
17. The inner if condition is evaluated only when the outer if condition is false(True/False)
18. A nested-if statement cannot be used inside else block(True/False)
19. You can implement polymorphism in C# through \_\_\_\_\_\_\_\_\_\_
20. What is infinite loop?
21. What is the correct syntax of for each loop explain it?
22. What is the difference between break and continue?
23. What do you understand by the word **instance**?
24. Why do we need to have different of objects of a single class?
25. \_\_\_\_\_\_\_\_\_\_\_\_\_ is keyword used for instantiation?
26. Why do we use dot (.) operator?
27. encapsulation, also called data hiding.(True.False)
28. Access Modifier is an optional keyword specifying the access level of an instance method. (True/False)
29. Items saved in hashtable are sorted (True/False)
30. What are the differences between constructors and method?
31. What is the return type of a constructor?
32. What do you mean by overloading? Is there any difference between constructor overloading and method overloading?
33. To invoke a constructor dot operator is used (True/False)
34. What is default constructor?
35. In C#, a method cannot be declared as sealed(True/False)
36. What are Sealed Classes?
37. What do you know about encapsulation? what keywords are used for this purpose?
38. Which access modifier provides the least permissive access level?
39. What do you know about protected keyword?
40. A sealed class cannot have any protected members, Explain why?
41. How do you invoke a method?
42. If the method’s return type is set to void, then, a call to the method results in execution of the statements within the method without returning any value to the caller.(True/ False)
43. How do you Pass Values to Methods and how do you take a value from method?
44. You can override a base class method in the derived class using appropriate C# keywords, what are thsoe?
45. Why would you use ArrayList?
46. he instance of the derived class will always first invoke the constructor of the base class followed by the constructor of the derived class(True/False)?
47. You just need to insert a \_\_\_\_\_\_\_\_\_\_\_ after the name of the derived class followed by the name of the base class?
48. The purpose of \_\_\_\_\_\_\_\_\_\_\_ is to reuse common methods and attributes among classes without recreating them?
49. What do you understand by abstract method?
50. you have no control on when the destructor is going to be executed. (True/False)
51. the class that is derived from another class is called \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
52. A subclass inherits all the members such as fields, nested classes, and methods from its super class except those with private access specifier(True/False)
53. When a parent class has more than one child classes at different levels, it is called \_\_\_\_\_\_\_\_\_\_\_\_\_\_ inheritance
54. A C# class can contain one or more special member functions having the same name as the class, called \_\_\_\_\_\_\_\_\_\_\_\_\_\_
55. C# allows creation of an instance method in a subclass having the same signature and return type as an instance method of the super class. What is the name of this process?
56. \_\_\_\_\_\_\_\_\_\_ keyword is used to invoke parent class’s method in child class?
57. Describe the use of **this** keyword?
58. When the compiler resolves the binding of methods and method calls at compile time, it is called \_\_\_\_\_\_\_\_\_\_ binding
59. Overridden methods are bounded using \_\_\_\_\_\_\_\_\_\_\_\_\_\_ binding.
60. A method without any body. What does this statement mean?
61. The abstract method ends with a semicolon(True/False)
62. Polymorphism refers to an object that can have child classes (True/False)
63. When the compiler resolves the binding of methods and method calls at compile time, it is called static binding or early binding. If the compiler resolves the method calls and the binding at runtime, it is called dynamic binding or late binding.(True/Fale)
64. What is multiple inheritance does C# support it? It not then what is the alternative for this?
65. Visibility of an interface is always public(True/False)
66. An interface can extend multiple interfaces.(True/False)
67. What is the difference between abstract classes and interfaces? Can you instantiate an interface?
68. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ is the process of hiding the unnecessary details and revealing only the essential features of an object to the user.
69. abstract classes and interfaces are used to implement the concept of abstraction.(True/False)
70. What do you understand by the try and catch block? What is the use of finally block?
71. What is the difference between throw and throws keyword?
72. What is the use of multiple batch block?
73. The concept of declaring more than one constructor in a class is called\_\_\_\_\_\_\_
74. C# creates a default constructor for a class if no constructor is specified within the class. How many parameters does it has?
75. \_\_\_\_\_\_\_\_\_\_\_ constructors are used to prevent class instantiation
76. Private constructors are used if a class contains \_\_\_\_\_\_ data members because they don’t need object to be called.
77. No class is allowed to contain two methods with the same name and same signature(True/False)
78. The \_\_\_\_\_\_\_\_\_\_access modifier allows the class members to be accessible only within the classes of the same assembly.
79. Only one copy of a \_\_\_\_\_\_\_\_\_\_ variable is shared by all the objects of the class
80. Static classes cannot be inherited.
81. demonstrates how to use optional arguments.
82. Demonstrates the use of Named arguments
83. \_\_\_\_\_\_\_\_\_ is a template or blueprint which defines the state and behavior of all objects belonging to it.
84. C# supports zero-based index feature(True/False)
85. Difference between normal array and **Array** and **ArrayList** classes
86. What is jagged array?
87. What is the difference between declaration, initialization and usage
88. \_\_\_\_\_\_ loop is always executed at least once without the condition being checked.
89. What is No-Fall-Through Rule?
90. The \_\_\_\_\_\_\_\_ statement in C# allows you to programmatically throw exceptions
91. a catch block with the base class Exception is called \_\_\_\_\_\_\_\_\_\_
92. what is the Difference between Properties and Fields?
93. Properties can implement polymorphism, Explain how?
94. An interface can inherit multiple interfaces but cannot implement them. The implementation has to be done by a class.(True/False)
95. Interfaces cannot contain constants, data fields, constructors, destructors, and static members(True/False)
96. it is not mandatory for the abstract class to contain only abstract methods. It can contain nonabstract methods too. An abstract class cannot be sealed.