|  |
| --- |
| What is C#? |
|  | * C# (pronounced "C sharp") is a simple, modern, object-oriented, and type-safe programming language. * It will immediately be familiar to C and C++ programmers. * C# combines the high productivity of Rapid Application Development (RAD) languages. |

|  |  |
| --- | --- |
| **2.** | What are the types of comment in C#? |
|  | There are 3 types of comments in C#.   * Single line (//) * Multi (/\* \*/) * Page/XML Comments (///). |

|  |  |
| --- | --- |
| **3.** | What are the namespaces used in C#.NET? |
|  | Namespace is a logical grouping of class.   * using System; * using System.Collections.Generic; * using System.Windows.Forms; |

|  |  |
| --- | --- |
| **4.** | What are the characteristics of C#? |
|  | There are several characteristics of C# are :   * Simple * Type safe * Flexible * Object oriented * Compatible * Consistent * Interoperable * Modern |

|  |  |
| --- | --- |
| **5.** | What are the different categories of inheritance? |
|  | Inheritance in Object Oriented Programming is of four types:   * **Single inheritance** : Contains one base class and one derived class. * **Hierarchical inheritance** : Contains one base class and multiple derived classes of the same base class. * **Multilevel inheritance** : Contains a class derived from a derived class. * **Multiple inheritance** : Contains several base classes and a derived class |

|  |
| --- |
| What are the basic concepts of object oriented programming? |
|  | It is necessary to understand some of the concepts used extensively in object oriented programming.These include   * Objects * Classes * Data abstraction and encapsulation * Inheritance * Polymorphism * Dynamic Binding * Message passing. |

|  |  |
| --- | --- |
| **7.** | Can you inherit multiple interfaces? |
|  | Yes. Multiple interfaces may be inherited in C#. |

|  |  |
| --- | --- |
| **8.** | What is inheritance? |
|  | Inheritance is deriving the new class from the already existing one. |

|  |  |
| --- | --- |
| **9.** | Define scope? |
|  | Scope refers to the region of code in which a variable may be accessed. |

|  |  |
| --- | --- |
| **10.** | What is the difference between public, static and void? |
|  | * **public :**The keyword public is an access modifier that tells the C# compiler that the Main method is accessible by anyone. * **static :**The keyword static declares that the Main method is a global one and can be called without creating an instance of the class. The compiler stores the address of the method as the entry point and uses this information to begin execution before any objects are created. * **void :** The keyword void is a type modifier that states that the Main method does not return any value. |

|  |
| --- |
| What are the modifiers in C#? |
|  | * Abstract * Sealed * Virtual * Const * Event * Extern * Override * Readonly * Static * New |

|  |  |
| --- | --- |
| **12.** | What are the types of access modifiers in C#? |
|  | Access modifiers in C# are :   * public * protect * private * internal * internal protect |

|  |  |
| --- | --- |
| **13.** | What is boxing and unboxing? |
|  | Implicit conversion of value type to reference type of a variable is known as BOXING, for example integer to object type conversion.  Conversion of reference type variable back to value type is called as UnBoxing. |

|  |  |
| --- | --- |
| **14.** | What is object? |
|  | An object is an instance of a class. An object is created by using operator new. A class that creates an object in memory will contain the information about the values and behaviours (or methods) of that specific object. |

|  |  |
| --- | --- |
| **15.** | Where are the types of arrays in C#? |
|  | * Single-Dimensional * Multidimensional * Jagged arrays. |

|  |
| --- |
| What is the difference between Object and Instance? |
|  | An instance of a user-defined type is called an object. We can instantiate many objects from one class. An object is an instance of a class. |

|  |  |
| --- | --- |
| **17.** | Define destructors? |
|  | A destructor is called for a class object when that object passes out of scope or is explicitly deleted.A destructors as the name implies is used to destroy the objects that have been created by a constructors.Like a constructor , the destructor is a member function whose name is the same as the class name but is precided by a tilde. |

|  |  |
| --- | --- |
| **18.** | What is the use of enumerated data type? |
|  | An enumerated data type is another user defined type which provides a way for attaching names to numbers thereby increasing comprehensibility of the code. The enum keyword automatically enumerates a list of words by assigning them values 0,1,2, and so on. |

|  |  |
| --- | --- |
| **19.** | Define Constructors? |
|  | A constructor is a member function with the same name as its class. The constructor is invoked whenever an object of its associated class is created.It is called constructor because it constructs the values of data members of the class. |

|  |  |
| --- | --- |
| **20.** | What is encapsulation? |
|  | The wrapping up of data and functions into a single unit (called class) is known as encapsulation. Encapsulation containing and hiding information about an object, such as internal data structures and code. |

|  |
| --- |
| Does c# support multiple inheritance? |
|  | No,its impossible which accepts multi level inheritance. |

|  |  |
| --- | --- |
| **22.** | What is ENUM? |
|  | Enum are used to define constants. |

|  |  |
| --- | --- |
| **23.** | What is a data set? |
|  | A DataSet is an in memory representation of data loaded from any data source. |

|  |  |
| --- | --- |
| **24.** | What is the difference between private and public keyword? |
|  | * **Private :** The private keyword is the default access level and most restrictive among all other access levels. It gives least permission to a type or type member. A private member is accessible only within the body of the class in which it is declared. * **Public :** The public keyword is most liberal among all access levels, with no restrictions to access what so ever. A public member is accessible not only from within, but also from outside, and gives free access to any member declared within the body or outside the body. |

|  |  |
| --- | --- |
| **25.** | Define polymorphism? |
|  | Polymorphism means one name, multiple forms. It allows us to have more than one function with the same name in a program.It allows us to have overloading of operators so that an operation can exhibit different behaviours in different instances. |

|  |
| --- |
| What is Jagged Arrays? |
|  | * A jagged array is an array whose elements are arrays. * The elements of a jagged array can be of different dimensions and sizes. * A jagged array is sometimes called an **array–of–arrays**. |

|  |  |
| --- | --- |
| **27.** | what is an abstract base class? |
|  | An abstract class is a class that is designed to be specifically used as a base class. An abstract class contains at least one pure virtual function. |

|  |  |
| --- | --- |
| **28.** | How is method overriding different from method overloading? |
|  | When overriding a method, you change the behavior of the method for the derived class. Overloading a method simply involves having another method with the same name within the class. |

|  |  |
| --- | --- |
| **29.** | What is the difference between ref & out parameters? |
|  | An argument passed to a ref parameter must first be initialized. Compare this to an out parameter, whose argument does not have to be explicitly initialized before being passed to an out parameter. |

|  |  |
| --- | --- |
| **30.** | What is the use of using statement in C#? |
|  | The using statement is used to obtain a resource, execute a statement, and then dispose of that resource. |

|  |
| --- |
| What is serialization? |
|  | Serialization is the process of converting an object into a stream of bytes. De-serialization is the opposite process of creating an object from a stream of bytes.  Serialization / De-serialization is mostly used to transport objects. |

|  |  |
| --- | --- |
| **32.** | What are the difference between Structure and Class? |
|  | * Structures are value type and Classes are reference type * Structures can not have contractors or destructors. * Classes can have both contractors and destructors. * Structures do not support Inheritance, while Classes support Inheritance. |

|  |  |
| --- | --- |
| **33.** | What is difference between Class And Interface? |
|  | **Class** : is logical representation of object. It is collection of data and related sub procedures with defination. **Interface** : is also a class containg methods which is not having any definations.Class does not support multiple inheritance. But interface can support. |

|  |  |
| --- | --- |
| **34.** | What is Delegates? |
|  | Delegates are a type-safe, object-oriented implementation of function pointers and are used in many situations where a component needs to call back to the component that is using it. |

|  |  |
| --- | --- |
| **35.** | What is Authentication and Authorization? |
|  | **Authentication** is the process of identifying users. Authentication is identifying/validating the user against the credentials (username and password). **Authorization** performs after authentication. Authorization is the process of granting access to those users based on identity. Authorization allowing access of specific resource to user. |

|  |
| --- |
| What is a base class? |
|  | A class declaration may specify a base class by following the class name with a colon and the name of the base class. omitting a base class specification is the same as deriving from type object. |

|  |  |
| --- | --- |
| **37.** | Can “this” be used within a static method? |
|  | No ‘This’ cannot be used in a static method. As only static variables/methods can be used in a static method. |

|  |  |
| --- | --- |
| **38.** | What is difference between constants, readonly and, static ? |
|  | * Constants: The value can’t be changed. * Read-only: The value will be initialized only once from the constructor of the class. * Static: Value can be initialized once. |

|  |  |
| --- | --- |
| **39.** | What are the different types of statements supported in C#? |
|  | C# supports several different kinds of statements are   * Block statements * Declaration statements * Expression statements * Selection statements * Iteration statements * Jump statements * Try catch statements * Checked and unchecked * Lock statement |

|  |  |
| --- | --- |
| **40.** | What is an interface class? |
|  | It is an abstract class with public abstract methods all of which must be implemented in the inherited classes. |
| **41.** | what are value types and reference types? |
|  | Value types are stored in the Stack. Examples : bool, byte, chat, decimal, double, enum , float, int, long, sbyte, short, strut, uint, ulong, ushort.  Reference types are stored in the Heap.  Examples : class, delegate, interface, object, string. |

|  |  |
| --- | --- |
| **42.** | What is the difference between string keyword and System.String class? |
|  | String keyword is an alias for Syste.String class. Therefore, System.String and string keyword are the same, and you can use whichever naming convention you prefer. The String class provides many methods for safely creating, manipulating, and comparing strings. |

|  |  |
| --- | --- |
| **43.** | What are the two data types available in C#? |
|  | * Value type * Reference type |

|  |  |
| --- | --- |
| **44.** | What are the different types of Caching? |
|  | There are three types of Caching :   * Output Caching: stores the responses from an asp.net page. * Fragment Caching: Only caches/stores the portion of page (User Control) * Data Caching: is Programmatic way to Cache objects for performance. |

|  |  |
| --- | --- |
| **45.** | What is the difference between Custom Control and User Control? |
|  | **Custom Controls** are compiled code (Dlls), easier to use, difficult to create, and can be placed in toolbox. Drag and Drop controls. Attributes can be set visually at design time. Can be used by Multiple Applications (If Shared Dlls), Even if Private can copy to bin directory of web application add reference and use. Normally designed to provide common functionality independent of consuming Application.   **User Controls** are similar to those of ASP include files, easy to create, can not be placed in the toolbox and dragged - dropped from it. A User Control is shared among the single application files. |

|  |
| --- |
| What is methods? |
|  | A method is a member that implements a computation or action that can be performed by an object or class. Static methods are accessed through the class. Instance methods are accessed through instances of the class. |

|  |  |
| --- | --- |
| **47.** | What is fields? |
|  | A field is a variable that is associated with a class or with an instance of a class. |

|  |  |
| --- | --- |
| **48.** | What is events? |
|  | An event is a member that enables a class or object to provide notifications. An event is declared like a field except that the declaration includes an event keyword and the type must be a delegate type. |

|  |  |
| --- | --- |
| **49.** | What is literals and their types? |
|  | Literals are value constants assigned to variables in a program. C# supports several types of literals are   * Integer literals * Real literals * Boolean literals * Single character literals * String literals * Backslash character literals |

|  |  |  |
| --- | --- | --- |
| **50.** | What is the difference between value type and reference type? | |
|  | * Value types are stored on the stack and when a value of a variable is assigned to another variable. * Reference types are stored on the heap, and when an assignment between two reference variables occurs. | |
| What are the features of c#? | |
|  | | * C# is a simple and powerful programming language for writing enterprise edition applications. * This is a hybrid of C++ and VB. It retains many C++ features in the area statements,expressions, and operators and incorporated the productivity of VB. * C# helps the developers to easily build the web services that can be used across the Internet through any language, on any platform. * C# helps the developers accomplishing with fewer lines of code that will lead to the fewer errors in the code. * C# introduces the considerable improvement and innovations in areas such as type safety,versioning. events and garbage collections. |

|  |  |
| --- | --- |
| **52.** | What are the types of errors? |
|  | * Syntax error * Logic error * Runtime error |

|  |  |
| --- | --- |
| **53.** | What is the difference between break and continue statement? |
|  | The **break statement** is used to terminate the current enclosing loop or conditional statements in which it appears. We have already used the break statement to come out of switch statements. The **continue statement** is used to alter the sequence of execution. Instead of coming out of the loop like the break statement did, the continue statement stops the current iteration and simply returns control back to the top of the loop. |

|  |  |
| --- | --- |
| **54.** | Define namespace? |
|  | The namespace are known as containers which will be used to organize the hierarchical set of .Net classes. |

|  |  |
| --- | --- |
| **55.** | What is a code group? |
|  | A code group is a set of assemblies that share a security context. |

|  |
| --- |
| What are sealed classes in C#? |
|  | The sealed modifier is used to prevent derivation from a class. A compile-time error occurs if a sealed class is specified as the base class of another class. |

|  |  |
| --- | --- |
| **57.** | What is the difference between static and instance methods? |
|  | A method declared with a static modifier is a static method. A static method does not operate on a specific instance and can only access static members.  A method declared without a static modifier is an instance method. An instance method operates on a specific instance and can access both static and instance members. The instance on which an instance method was invoked can be explicitly accessed as this. It is an error to refer to this in a static method. |

|  |  |
| --- | --- |
| **58.** | What are the different types of variables in C#? |
|  | Different types of variables used in C# are :   * static variables * instance variable * value parameters * reference parameters * array elements * output parameters * local variables |

|  |  |
| --- | --- |
| **59.** | What is meant by method overloading? |
|  | Method overloading permits multiple methods in the same class to have the same name as long as they have unique signatures. When compiling an invocation of an overloaded method, the compiler uses overload resolution to determine the specific method to invoke |

|  |  |
| --- | --- |
| **60.** | What is parameters? |
|  | Parameters are used to pass values or variable references to methods. The parameters of a method get their actual values from the arguments that are specified when the method is invoked. There are four kinds of parameters: value parameters, reference parameters, output parameters, and parameter arrays. |

|  |
| --- |
| Is C# is object oriented? |
|  | YEs, C# is an OO langauge in the tradition of Java and C++. |

|  |  |
| --- | --- |
| **62.** | What is the difference between Array and Arraylist? |
|  | An array is a collection of the same type. The size of the array is fixed in its declaration. A linked list is similar to an array but it doesn’t have a limited size. |

|  |  |
| --- | --- |
| **63.** | What are the special operators in C#? |
|  | C# supports the following special operators.   * is (relational operator) * as (relational operator) * typeof (type operator) * sizeof (size operator) * new (object creator) * .dot (member access operator) * checked (overflow checking) * unchecked (prevention of overflow checking) |

|  |  |
| --- | --- |
| **64.** | What is meant by operators in c#? |
|  | An operator is a member that defines the meaning of applying a particular expression operator to instances of a class. Three kinds of operators can be defined: unary operators, binary operators, and conversion operators. All operators must be declared as public and static. |

|  |  |
| --- | --- |
| **65.** | What is a parameterized type? |
|  | A parameterized type is a type that is parameterized over another value or type. |

|  |
| --- |
| What are the features of abstract class? |
|  | * An abstract class cannot be instantiated, and it is an error to use the new operator on an abstract class. * An abstract class is permitted (but not required) to contain abstract methods and accessors. * An abstract class cannot be scaled. |

|  |  |
| --- | --- |
| **67.** | What is the use of abstract keyword? |
|  | The modifier **abstract** is a keyword used with a class, to indicate that this class cannot itself have direct instances or objects, and it is intended to be only a 'base' class to other classes. |

|  |  |
| --- | --- |
| **68.** | What is the use of goto statement? |
|  | The goto statement is also included in the C# language. This goto can be used to jump from inside a loop to outside. But jumping from outside to inside a loop is not allowed. |

|  |  |
| --- | --- |
| **69.** | What is the difference between console and window application? |
|  | * A console application, which is designed to run at the command line with no user interface. * A Windows application, which is designed to run on a user’s desktop and has a user interface. |

|  |  |
| --- | --- |
| **70.** | What is the use of return statement? |
|  | The return statement is associated with procedures (methods or functions). On executing the return statement, the system passes the control from the called procedure to the calling procedure. This return statement is used for two purposes :   * to return immediately to the caller of the currently executed code * to return some value to the caller of the currently executed code. |

|  |
| --- |
| What are the features of abstract class? |
|  | * An abstract class cannot be instantiated, and it is an error to use the new operator on an abstract class. * An abstract class is permitted (but not required) to contain abstract methods and accessors. * An abstract class cannot be scaled. |

|  |  |
| --- | --- |
| **67.** | What is the use of abstract keyword? |
|  | The modifier **abstract** is a keyword used with a class, to indicate that this class cannot itself have direct instances or objects, and it is intended to be only a 'base' class to other classes. |

|  |  |
| --- | --- |
| **68.** | What is the use of goto statement? |
|  | The goto statement is also included in the C# language. This goto can be used to jump from inside a loop to outside. But jumping from outside to inside a loop is not allowed. |

|  |  |
| --- | --- |
| **69.** | What is the difference between console and window application? |
|  | * A console application, which is designed to run at the command line with no user interface. * A Windows application, which is designed to run on a user’s desktop and has a user interface. |

|  |  |
| --- | --- |
| **70.** | What is the use of return statement? |
|  | The return statement is associated with procedures (methods or functions). On executing the return statement, the system passes the control from the called procedure to the calling procedure. This return statement is used for two purposes :   * to return immediately to the caller of the currently executed code * to return some value to the caller of the currently executed code. |

|  |
| --- |
| What is the difference between Array and LinkedList? |
|  | Array is a simple sequence of numbers which are not concerned about each others positions. they are independent of each others positions. adding,removing or modifying any array element is very easy. Compared to arrays ,linked list is a comlicated sequence of numbers. |

|  |  |
| --- | --- |
| **72.** | Does C# have a throws clause? |
|  | No, unlike Java, C# does not require the developer to specify the exceptions that a method can throw. |

|  |  |
| --- | --- |
| **73.** | Does C# support a variable number of arguments? |
|  | Yes, uisng the params keyword. The arguments are specified as a list of arguments of a specific type. |

|  |  |
| --- | --- |
| **74.** | Can you override private virtual methods? |
|  | No, private methods are not accessible outside the class. |

|  |  |
| --- | --- |
| **75.** | What is a multi cast delegates? |
|  | Each delegate object holds reference to a single method. However, it is possible for a delegate object to hold references of and invoke multiple methods. Such delegate objects are called multicast delegates or combinable delegates. |

|  |
| --- |
| Which is an exclusive feature of C#? |
|  | Xml documentation. |

|  |  |
| --- | --- |
| **77.** | Is using of exceptions in C# recommended? |
|  | Yes, exceptions are the recommended error handling mechanism in .NET Framework. |

|  |  |
| --- | --- |
| **78.** | What does a break statement do in switch statements? |
|  | * The break statement terminates the loop in which it exists. It also changes the flow of the execution of a program. * In switch statements, the break statement is used at the end of a case statement. The break statement is mandatory in C# and it avoids the fall through of one case statement to another. |

|  |  |
| --- | --- |
| **79.** | Is C# object oriented? |
|  | Yes, C# is an OO language in the tradition of java and C++. |

|  |  |
| --- | --- |
| **80.** | What is smart navigation? |
|  | The cursor position is maintained when the page gets refreshed due to the server side validation and the page gets refreshed. |

|  |
| --- |
| What is the difference between CONST and READONLY? |
|  | Both are meant for constant values. A const field can only be initialized at the declaration of the field. A readonly field can be initialized either at the declaration or. |

|  |  |
| --- | --- |
| **82.** | Does C# have a throws clause? |
|  | No, unlike Java, C# does not require (or even allow) the developer to specify the exceptions that a method can throw. |

|  |  |
| --- | --- |
| **83.** | What are the different ways a method can be overloaded? |
|  | Different parameter data types, different number of parameters, different order of parameters. |

|  |  |
| --- | --- |
| **84.** | Do events have return type? |
|  | No, events do not have return type. |

|  |  |
| --- | --- |
| **85.** | What is event? |
|  | * An event is an action performed based on another method of the program. * An event is a delegate type dass member that is used by an object or a class to provide a notification to other objects that an event has occurred. * An event can be declared with the help of the event keyword. |

|  |
| --- |
| What is an identifier? |
|  | Identifiers are nothing but names given to various entities uniquely identified in a program. |

|  |  |
| --- | --- |
| **87.** | What are the different types of literals in C#? |
|  | * Boolean literals : True and False are literals of the Boolean type that map to the true and false state, respectively. * Integer literals : Used to write values of types Int, ulnt, long, and ulong. * Real literals : Used to write values of types float, double, and dedmal. * Character literals : Represents a single character and usually consists of a character in quotes, such as 'a'. * String literals : C# supports two types of string literals, regular string literal and verbatim string literals. A regular string literal consists of zero or more characters enclosed in double quotes, such as "116110". A verbatim string literal consists of an @ character followed by a double–quote character, such as ©"hello". * The Null literal : Represents the null–type. |

|  |  |
| --- | --- |
| **88.** | What is meant by data encapsulation? |
|  | Data encapsulation, also referred to as data hiding, is the mechanism whereby the implementation details of a class are kept hidden from the user. The user can only perform a restricted set of operations on the hidden members of the class by executing special functions called methods. |

|  |  |
| --- | --- |
| **89.** | Can you override private virtual methods? |
|  | No. Private methods are not accessible outside the class. |

|  |  |
| --- | --- |
| **90.** | What is the main difference between a subprocedure and a function? |
|  | Subprocedures do not return a value, while functions do. |

|  |
| --- |
| How does C# differ from C++? |
|  | * C# does not support #include statement. It uses only **using** statement. * In C# , class definition does not use a semicolon at the end. * C# does not support multiple code inheritance. * Casting in C# is much safer than in c++. * In C# switch can also be used on string values. * Command line parameters array behave differently in C# as compared to C++. |

|  |  |
| --- | --- |
| **92.** | What is nested class? |
|  | * A Nested classes are classes within classes. * A nested class is any class whose declaration occurs within the body of another class or interface. |

|  |  |
| --- | --- |
| **93.** | Can you have parameters for static constructors? |
|  | No, static constructors cannot have parameters. |

|  |  |
| --- | --- |
| **94.** | Is String is Value Type or Reference Type in C#? |
|  | String is an object(Reference Type). |

|  |  |
| --- | --- |
| **95.** | Does C# provide copy constructor? |
|  | No, C# does not provide copy constructor. |

|  |
| --- |
| Can a class or a struct have multiple constructors? |
|  | Yes, a class or a struct can have multiple constructors. Constructors in C# can be overloaded. |

|  |  |
| --- | --- |
| **97.** | Can you create an instance of an interface? |
|  | No, you cannot create an instance of an interface. |

|  |  |
| --- | --- |
| **98.** | Can an Interface contain fields? |
|  | No, an Interface cannot contain fields. |

|  |  |
| --- | --- |
| **99.** | Can a class have static constructor? |
|  | Yes, a class can have static constructor. Static constructors are called automatically, immediately before any static fields are accessed, and are generally used to initialize static class members. It is called automatically before the first instance is created or any static members are referenced. Static constructors are called before instance constructors. An example is shown below. |

|  |  |
| --- | --- |
| **100.** | What is the main use of delegates in C#? |
|  | Delegates are mainly used to define call back methods. |

|  |
| --- |
| What is the difference between Shadowing and Overriding? |
|  | * Overriding redefines only the implementation while shadowing redefines the whole element. * In overriding derived classes can refer the parent class element by using "ME" keyword, but in shadowing you can access it by "MYBASE". |

|  |  |
| --- | --- |
| **102.** | Can events have access modifiers? |
|  | Yes, you can have access modifiers in events. You can have events with the protected keyword, which will be accessible only to inherited classes. You can have private events only for objects in that class. |

|  |  |
| --- | --- |
| **103.** | Why is the virtual keyword used in code? |
|  | The Virtual keyword is used in code to define methods and the properties that can be overridden in derived classes. |

|  |  |
| --- | --- |
| **104.** | What are constructors and destructors? |
|  | * Constructors and destructors are special methods. * Constructors and destructors are special methods of every class. * Each class has its own constructor and destructor and are called automatically when the instance of a class is created or destroyed. * The constructor initializes all class members whenever you access the class and the destructor destroys them when the objects are not required anymore. |

|  |  |  |
| --- | --- | --- |
| **105.** | How can we suppress a finalize method? | |
|  | GC.SuppressFinalize() | |
| Does C# support a variable number of arguments? | |
|  | | * Yes, using the params keyword. * The arguments are specified as a list of arguments of a specific type, e.g., int. For ultimate flexibility, the type can be object. * The standard example of a method which uses this approach is System.console.writeLine(). | |

|  |  |
| --- | --- |
| **107.** | Which method will you call to start a thread? |
|  | Start |

|  |  |
| --- | --- |
| **108.** | What is Generic? |
|  | * Generic help us to create flexible strong type collection. * Generic basically seperate the logic from the datatype in order maintain better reusability, better maintainability etc. |

|  |  |
| --- | --- |
| **109.** | What are the different types of polymorphism? |
|  | There are two types of polymorphism. They are   * Compile time Polymorphism * Run time Polymorphism |

|  |  |  |  |
| --- | --- | --- | --- |
| **110.** | What is the difference between compile time polymorphism and run time polymorphism? | | |
|  | **Compile time Polymorphism**   * Compile time Polymorphism also known as method overloading. * Method overloading means having two or more methods with the same name but with different signatures.   **Run time Polymorphism**   * Run time Polymorphism also known as method overriding. * Method overriding means having two or more methods with the same name , same signature but with different implementation | | |
| Which namespace enables multithreaded programming in XML? | |
|  | | System.Threading |

|  |  |
| --- | --- |
| **112.** | Can we declare a block as static in c#? |
|  | No, because c# doesnot support static block, but it supports static method. |

|  |  |
| --- | --- |
| **113.** | Can we declare a method as sealed? |
|  | In C# a method can't be declared as sealed. However when we override a method in a derived class, we can declare the overridden method as sealed. By declaring it as sealed, we can avoid further overriding of this method. |

|  |  |
| --- | --- |
| **114.** | What Command is used to implement properties in C#? |
|  | get & set access modifiers are used to implement properties in c#. |

|  |  |  |
| --- | --- | --- |
| **115.** | What is static member? | |
|  | The member defined as static which can be invoked directly from the class level, rather than from its instance. | |
| What is the syntax to inherit from a class in C#? | |
|  | | * When a class is derived from another class, then the members of the base class become the members of the derived class. * The access modifier used while accessing members of the base class specifies the access status of the base class members inside the derived class. * The syntax to inherit a class from another class In C# is as follows : **class MyNewClass : MyBaseClass** |

|  |  |
| --- | --- |
| **117.** | What is a basic difference between the while loop and do while loop in C#? |
|  | The while loop tests its condition at the beginning, which means that the enclosed set of statements run for zero or more number of times if the condition evaluates to true. The do while loop iterates a set of statements at least once and then checks the condition at the end. |

|  |  |
| --- | --- |
| **118.** | What is the main difference between a subprocedure and a function? |
|  | Subprocedures do not return a value, while functions do. |

|  |  |
| --- | --- |
| **119.** | What are sealed classes in c#? |
|  | * The sealed modifier is used to prevent derivation from a class. * A compile time error occurs if a sealed class is specified as the base class of another class. |

|  |  |  |
| --- | --- | --- |
| **120.** | What is the difference between class and an Interface? | |
|  | * Abstract classes can have implementations for some of its members, but the interface can't have implementation for any of its members. * Interfaces cannot have fields where as an abstract class can have fields. * An interface can inherit from another interface only and cannot inherit from an abstract class, where as an abstract class can inherit from another abstract class or another interface. * A class can inherit from multiple interfaces at the same time, where as a class cannot inherit from multiple classes at the same time. * Abstract class members can have access modifiers where as interface members cannot have access modifiers. | |
| What is the difference between an abstract method & virtual method? | |
|  | | An Abstract method does not provide an implementation and forces overriding to the deriving class (unless the deriving class also an abstract class), where as the virtual method has an implementation and leaves an option to override it in the deriving class. Thus Virtual method has an implementation & provides the derived class with the option of overriding it. Abstract method does not provide an implementation & forces the derived class to override the method. |

|  |  |
| --- | --- |
| **122.** | What is Static Method? |
|  | It is possible to declare a method as Static provided that they don't attempt to access any instance data or other instance methods. |

|  |  |
| --- | --- |
| **123.** | What is a New modifier? |
|  | The new modifier hides a member of the base class. C# supports only hide by signature. |

|  |  |
| --- | --- |
| **124.** | What are the advantages of get and set properties in C#? |
|  | * The get property accessor is used to return the property value. * The set property accessor is used to assign a new value. |

|  |  |
| --- | --- |
| **125.** | What are the difference between const and readonly? |
|  | * A const can not be static, while readonly can be static. * A const need to be declared and initialized at declaration only, while a readonly can be initialized at declaration or by the code in the constructor. * A const's value is evaluated at design time, while a readonly's value is evaluated at runtime. |