**51. What is LINQ in C#?**  
a) Language Integrated Query  
b) Local Integrated Query  
c) Long Integrated Query  
d) List Integrated Query  
✅ **Answer:** a) Language Integrated Query  
💡 **Explanation:** LINQ allows querying collections using C# syntax.

**52. Which method is used to filter elements in a LINQ query?**  
a) Select  
b) Where  
c) OrderBy  
d) GroupBy  
✅ **Answer:** b) Where  
💡 **Explanation:** Where filters elements based on a condition.

**53. What is the purpose of the async keyword in C#?**  
a) To define a synchronous method  
b) To define an asynchronous method  
c) To define a static method  
d) To define a virtual method  
✅ **Answer:** b) To define an asynchronous method  
💡 **Explanation:** async enables asynchronous programming in C#.

**54. Which method is used to run asynchronous code in C#?**  
a) RunAsync  
b) ExecuteAsync  
c) Task.Run  
d) Async.Run  
✅ **Answer:** c) Task.Run  
💡 **Explanation:** Task.Run executes code asynchronously on a thread pool.

**55. What is Entity Framework in .NET?**  
a) A web framework  
b) A database framework  
c) A UI framework  
d) A testing framework  
✅ **Answer:** b) A database framework  
💡 **Explanation:** EF is an ORM for interacting with databases using C#.

**56. Which method is used to save changes to the database in Entity Framework?**  
a) SaveChanges  
b) Commit  
c) Execute  
d) Update  
✅ **Answer:** a) SaveChanges  
💡 **Explanation:** SaveChanges() commits changes to the database.

**57. What is the purpose of the DbContext class in EF?**  
a) To define database tables  
b) To manage database connections  
c) To manage database operations  
d) To define database queries  
✅ **Answer:** c) To manage database operations  
💡 **Explanation:** DbContext is the primary class for interacting with EF.

**58. Which keyword is used to define a nullable type in C#?**  
a) ?  
b) !  
c) #  
d) $  
✅ **Answer:** a) ?  
💡 **Explanation:** int? allows null values for value types.

**59. What is the purpose of the ref keyword in C#?**  
a) To pass a value by reference  
b) To pass a value by value  
c) To define a static method  
d) To define a virtual method  
✅ **Answer:** a) To pass a value by reference  
💡 **Explanation:** ref allows a method to modify the caller’s variable.

**60. Which method is used to convert a string to an integer in C#?**  
a) int.Parse  
b) Convert.ToInt  
c) ToInt32  
d) ParseInt  
✅ **Answer:** a) int.Parse  
💡 **Explanation:** int.Parse("123") converts a string to an integer.

**61. What is the purpose of the out keyword in C#?**  
a) To pass a value by reference  
b) To pass a value by value  
c) To define a static method  
d) To define a virtual method  
✅ **Answer:** a) To pass a value by reference  
💡 **Explanation:** out allows a method to return multiple values.

**62. Which method is used to convert a string to a double in C#?**  
a) double.Parse  
b) Convert.ToDouble  
c) ToDouble  
d) ParseDouble  
✅ **Answer:** a) double.Parse  
💡 **Explanation:** double.Parse("3.14") converts a string to a double.

**63. What is the purpose of the IEnumerable interface in C#?**  
a) To define a collection  
b) To define a query  
c) To define an iterator  
d) To define a method  
✅ **Answer:** c) To define an iterator  
💡 **Explanation:** IEnumerable allows iteration over a collection.

**64. Which method is used to sort elements in a LINQ query?**  
a) Select  
b) Where  
c) OrderBy  
d) GroupBy  
✅ **Answer:** c) OrderBy  
💡 **Explanation:** OrderBy sorts elements in ascending order.

**65. What is the purpose of the async keyword in C#?**  
a) To define a synchronous method  
b) To define an asynchronous method  
c) To define a static method  
d) To define a virtual method  
✅ **Answer:** b) To define an asynchronous method  
💡 **Explanation:** async enables non-blocking operations.

**66. Which method is used to run asynchronous code in C#?**  
a) RunAsync  
b) ExecuteAsync  
c) Task.Run  
d) Async.Run  
✅ **Answer:** c) Task.Run  
💡 **Explanation:** Task.Run executes code on a background thread.

**67. What is the purpose of the DbContext class in EF?**  
a) To define database tables  
b) To manage database connections  
c) To manage database operations  
d) To define database queries  
✅ **Answer:** c) To manage database operations  
💡 **Explanation:** DbContext is the main class for EF operations.

**68. Which method is used to save changes in EF?**  
a) SaveChanges  
b) Commit  
c) Execute  
d) Update  
✅ **Answer:** a) SaveChanges  
💡 **Explanation:** SaveChanges() commits changes to the database.

**69. What is the purpose of the ? symbol in C#?**  
a) To define a nullable type  
b) To define a static method  
c) To define a virtual method  
d) To define a generic type  
✅ **Answer:** a) To define a nullable type  
💡 **Explanation:** int? allows null values for value types.

**70. What is the difference between ref and out in C#?**  
a) ref requires initialization, out does not  
b) out requires initialization, ref does not  
c) Both are the same  
d) ref is for output only  
✅ **Answer:** a) ref requires initialization, out does not  
💡 **Explanation:** ref must be initialized before passing; out must be assigned inside the method.

Let me continue with questions 71–100 in the next message to keep it clean and readable. Here’s the final stretch of your 200-question goal — continuing from:

**🧠 C# and .NET MCQs (71–100)**

**71. Which collection class represents a FIFO structure?**  
a) Stack  
b) Queue  
c) List  
d) Dictionary  
✅ **Answer:** b) Queue  
💡 **Explanation:** Queue follows First-In-First-Out order.

**72. Which collection class represents a LIFO structure?**  
a) Stack  
b) Queue  
c) List  
d) Dictionary  
✅ **Answer:** a) Stack  
💡 **Explanation:** Stack follows Last-In-First-Out order.

**73. Which interface is used to compare objects for sorting?**  
a) IEnumerable  
b) IComparable  
c) IComparer  
d) IDisposable  
✅ **Answer:** b) IComparable  
💡 **Explanation:** IComparable allows objects to be compared for sorting.

**74. Which interface is used to compare two different objects?**  
a) IComparable  
b) IComparer  
c) IEnumerable  
d) IDisposable  
✅ **Answer:** b) IComparer  
💡 **Explanation:** IComparer compares two different objects.

**75. What is boxing in C#?**  
a) Converting a value type to object  
b) Converting an object to value type  
c) Converting string to int  
d) Converting int to string  
✅ **Answer:** a) Converting a value type to object  
💡 **Explanation:** Boxing wraps a value type in an object.

**76. What is unboxing in C#?**  
a) Converting a value type to object  
b) Converting an object to value type  
c) Converting string to int  
d) Converting int to string  
✅ **Answer:** b) Converting an object to value type  
💡 **Explanation:** Unboxing extracts the value type from the object.

**77. What is the default access modifier for a class in C#?**  
a) public  
b) private  
c) internal  
d) protected  
✅ **Answer:** c) internal  
💡 **Explanation:** Classes are internal by default in C#.

**78. What is the default access modifier for class members?**  
a) public  
b) private  
c) internal  
d) protected  
✅ **Answer:** b) private  
💡 **Explanation:** Class members are private by default.

**79. What is the use of the sealed keyword in C#?**  
a) To prevent inheritance  
b) To allow multiple inheritance  
c) To define a static class  
d) To define an abstract class  
✅ **Answer:** a) To prevent inheritance  
💡 **Explanation:** sealed prevents a class from being inherited.

**80. What is the use of the abstract keyword in C#?**  
a) To define a class that cannot be instantiated  
b) To define a static class  
c) To define a sealed class  
d) To define a constructor  
✅ **Answer:** a) To define a class that cannot be instantiated  
💡 **Explanation:** Abstract classes are base classes that must be inherited.

**81. What is polymorphism in C#?**  
a) One method with multiple forms  
b) One class with multiple names  
c) One object with multiple classes  
d) One variable with multiple types  
✅ **Answer:** a) One method with multiple forms  
💡 **Explanation:** Polymorphism allows methods to behave differently based on context.

**82. What is method overriding?**  
a) Defining multiple methods with the same name  
b) Replacing a base class method in a derived class  
c) Calling a method from another class  
d) Using a method from a library  
✅ **Answer:** b) Replacing a base class method in a derived class  
💡 **Explanation:** Overriding allows a derived class to provide a new implementation.

**83. What is method overloading?**  
a) Defining multiple methods with the same name but different parameters  
b) Replacing a method in a derived class  
c) Calling a method from a base class  
d) Using a method from another namespace  
✅ **Answer:** a) Defining multiple methods with the same name but different parameters  
💡 **Explanation:** Overloading allows multiple methods with the same name but different signatures.

**84. What is the use of the virtual keyword?**  
a) To define a static method  
b) To define a method that can be overridden  
c) To define a sealed method  
d) To define a constructor  
✅ **Answer:** b) To define a method that can be overridden  
💡 **Explanation:** virtual allows a method to be overridden in a derived class.

**85. What is the use of the override keyword?**  
a) To define a new method  
b) To override a virtual method  
c) To hide a method  
d) To define a static method  
✅ **Answer:** b) To override a virtual method  
💡 **Explanation:** override replaces a base class method.

**86. What is the use of the new keyword in method hiding?**  
a) To override a method  
b) To hide a base class method  
c) To define a new class  
d) To define a constructor  
✅ **Answer:** b) To hide a base class method  
💡 **Explanation:** new hides a method from the base class.

**87. What is the purpose of the interface keyword?**  
a) To define a class  
b) To define a contract  
c) To define a method  
d) To define a property  
✅ **Answer:** b) To define a contract  
💡 **Explanation:** Interfaces define a contract that implementing classes must follow

**87. What is the purpose of the IEnumerable interface in C#?**  
a) To define a collection  
b) To define a query  
c) To define an iterator  
d) To define a method  
✅ **Answer:** c) To define an iterator  
💡 **Explanation:** IEnumerable allows iteration over a collection using foreach.

**88. Which method is used to sort elements in a LINQ query?**  
a) Select  
b) Where  
c) OrderBy  
d) GroupBy  
✅ **Answer:** c) OrderBy  
💡 **Explanation:** OrderBy sorts elements in ascending order.

**89. What is the purpose of the async keyword in C#?**  
a) To define a synchronous method  
b) To define an asynchronous method  
c) To define a static method  
d) To define a virtual method  
✅ **Answer:** b) To define an asynchronous method  
💡 **Explanation:** async enables asynchronous programming by allowing the use of await.

**90. Which method is used to run asynchronous code in C#?**  
a) RunAsync  
b) ExecuteAsync  
c) Task.Run  
d) Async.Run  
✅ **Answer:** c) Task.Run  
💡 **Explanation:** Task.Run executes code asynchronously on a background thread.

**91. What is the purpose of the sealed keyword in C#?**  
a) To allow inheritance  
b) To prevent inheritance  
c) To define an interface  
d) To define a static class  
✅ **Answer:** b) To prevent inheritance  
💡 **Explanation:** A sealed class cannot be inherited.

**92. What is the use of the static keyword in C#?**  
a) To define a class that can be inherited  
b) To define a class that cannot be instantiated  
c) To define members that belong to the type itself  
d) To define a virtual method  
✅ **Answer:** c) To define members that belong to the type itself  
💡 **Explanation:** static members are shared across all instances.

**93. What is the purpose of the readonly keyword in C#?**  
a) To make a variable constant  
b) To allow modification at runtime  
c) To assign a value only during declaration or in a constructor  
d) To prevent method overriding  
✅ **Answer:** c) To assign a value only during declaration or in a constructor  
💡 **Explanation:** readonly fields can be assigned once.

**94. What is the difference between const and readonly in C#?**  
a) const is runtime, readonly is compile-time  
b) const is compile-time, readonly is runtime  
c) Both are the same  
d) readonly is faster  
✅ **Answer:** b) const is compile-time, readonly is runtime  
💡 **Explanation:** const values are fixed at compile time.

**95. What is a partial class in C#?**  
a) A class that is incomplete  
b) A class split across multiple files  
c) A class that cannot be inherited  
d) A class with partial methods  
✅ **Answer:** b) A class split across multiple files  
💡 **Explanation:** partial allows a class to be defined in multiple files.

**96. What is an anonymous type in C#?**  
a) A type without a name  
b) A type with a dynamic name  
c) A type defined in a library  
d) A type that cannot be instantiated  
✅ **Answer:** a) A type without a name  
💡 **Explanation:** Anonymous types are created using new { } syntax.

**97. What is the purpose of the var keyword in C#?**  
a) To define a variable with a fixed type  
b) To define a variable with a dynamic type  
c) To define a variable with an inferred type  
d) To define a constant  
✅ **Answer:** c) To define a variable with an inferred type  
💡 **Explanation:** var lets the compiler infer the type.

**98. What is the use of the dynamic keyword in C#?**  
a) To define a variable with a fixed type  
b) To define a variable with a compile-time type  
c) To define a variable with a runtime-resolved type  
d) To define a constant  
✅ **Answer:** c) To define a variable with a runtime-resolved type  
💡 **Explanation:** dynamic bypasses compile-time type checking.

**99. What is the purpose of the nameof operator in C#?**  
a) To get the name of a variable as a string  
b) To get the type of a variable  
c) To get the value of a variable  
d) To get the namespace of a variable  
✅ **Answer:** a) To get the name of a variable as a string  
💡 **Explanation:** nameof(variable) returns "variable".

**100. What is the use of the is keyword in C#?**  
a) To compare values  
b) To check type compatibility  
c) To assign values  
d) To define a method  
✅ **Answer:** b) To check type compatibility  
💡 **Explanation:** is checks if an object is of a specific type.

**101. What is the purpose of the interface keyword in C#?**  
a) To define a class  
b) To define a contract  
c) To define a method  
d) To define a property  
✅ **Answer:** b) To define a contract  
💡 **Explanation:** Interfaces define a set of methods and properties that implementing classes must provide.

**102. Can an interface contain fields in C#?**  
a) Yes  
b) No  
c) Only static fields  
d) Only readonly fields  
✅ **Answer:** b) No  
💡 **Explanation:** Interfaces cannot contain fields, only declarations of methods, properties, events, or indexers.

**103. What is explicit interface implementation?**  
a) Implementing multiple interfaces  
b) Implementing an interface method with a different name  
c) Implementing an interface method so it can only be accessed through the interface  
d) Implementing an interface in a base class  
✅ **Answer:** c) Implementing an interface method so it can only be accessed through the interface  
💡 **Explanation:** Explicit implementation hides the method from the class interface unless accessed via the interface.

**104. What is the use of the base keyword in C#?**  
a) To call a method from the current class  
b) To call a method from the base class  
c) To define a base class  
d) To override a method  
✅ **Answer:** b) To call a method from the base class  
💡 **Explanation:** base is used to access members of the base class from a derived class.

**105. What is the use of the this keyword in C#?**  
a) Refers to the base class  
b) Refers to the current instance  
c) Refers to the static context  
d) Refers to the parent class  
✅ **Answer:** b) Refers to the current instance  
💡 **Explanation:** this refers to the current object instance.

**106. What is the purpose of the params keyword in C#?**  
a) To pass a fixed number of arguments  
b) To pass a variable number of arguments  
c) To pass arguments by reference  
d) To pass arguments by value  
✅ **Answer:** b) To pass a variable number of arguments  
💡 **Explanation:** params allows a method to accept a variable number of parameters.

**107. What is a delegate in C#?**  
a) A pointer to a class  
b) A pointer to a method  
c) A reference to a variable  
d) A reference to a property  
✅ **Answer:** b) A pointer to a method  
💡 **Explanation:** Delegates are type-safe function pointers.

**108. What is a multicast delegate?**  
a) A delegate that points to one method  
b) A delegate that points to multiple methods  
c) A delegate that returns multiple values  
d) A delegate that accepts multiple parameters  
✅ **Answer:** b) A delegate that points to multiple methods  
💡 **Explanation:** Multicast delegates can invoke multiple methods in a single call.

**109. What is an event in C#?**  
a) A method  
b) A delegate  
c) A notification mechanism  
d) A class  
✅ **Answer:** c) A notification mechanism  
💡 **Explanation:** Events are used to notify subscribers when something happens.

**110. What is the difference between a delegate and an event?**  
a) Events are more powerful  
b) Delegates are used for notifications  
c) Events are based on delegates but restrict access  
d) Delegates are based on events  
✅ **Answer:** c) Events are based on delegates but restrict access  
💡 **Explanation:** Events encapsulate delegates and limit invocation to the declaring class.

**111. What is the purpose of the delegate keyword in C#?**  
a) To define a class  
b) To define a method  
c) To define a type-safe function pointer  
d) To define a property  
✅ **Answer:** c) To define a type-safe function pointer  
💡 **Explanation:** Delegates are used to reference methods with a specific signature.

**112. What is the return type of a constructor in C#?**  
a) void  
b) int  
c) object  
d) None  
✅ **Answer:** d) None  
💡 **Explanation:** Constructors do not have a return type, not even void.

**113. What is the use of the finally block in exception handling?**  
a) To catch exceptions  
b) To throw exceptions  
c) To execute code regardless of exception  
d) To ignore exceptions  
✅ **Answer:** c) To execute code regardless of exception  
💡 **Explanation:** finally runs whether or not an exception is thrown.

**114. What is the base class of all exceptions in C#?**  
a) System.Object  
b) System.Exception  
c) System.BaseException  
d) System.Error  
✅ **Answer:** b) System.Exception  
💡 **Explanation:** All exceptions derive from System.Exception.

**115. What is the purpose of the throw keyword in C#?**  
a) To catch an exception  
b) To ignore an exception  
c) To raise an exception  
d) To log an exception  
✅ **Answer:** c) To raise an exception  
💡 **Explanation:** throw is used to signal the occurrence of an exception.

**116. What is the purpose of the try block in C#?**  
a) To define a method  
b) To define a class  
c) To handle exceptions  
d) To test code  
✅ **Answer:** c) To handle exceptions  
💡 **Explanation:** try is used to wrap code that may throw exceptions.

**117. What is the use of the catch block in C#?**  
a) To define a method  
b) To catch exceptions  
c) To throw exceptions  
d) To ignore exceptions  
✅ **Answer:** b) To catch exceptions  
💡 **Explanation:** catch handles exceptions thrown in the try block.

**118. What is the purpose of the using statement in resource management?**  
a) To import namespaces  
b) To define a class  
c) To ensure disposal of resources  
d) To define a method  
✅ **Answer:** c) To ensure disposal of resources  
💡 **Explanation:** using ensures that IDisposable objects are disposed.

**119. What is the difference between IDisposable and finalizer?**  
a) IDisposable is automatic, finalizer is manual  
b) IDisposable is manual, finalizer is automatic  
c) Both are the same  
d) Finalizer is faster  
✅ **Answer:** b) IDisposable is manual, finalizer is automatic  
💡 **Explanation:** IDisposable.Dispose() is called explicitly; finalizers are called by the GC.

**120. What is the purpose of the GC.Collect() method?**  
a) To collect garbage  
b) To force garbage collection  
c) To clean memory  
d) To delete objects  
✅ **Answer:** b) To force garbage collection  
💡 **Explanation:** GC.Collect() forces the .NET runtime to perform garbage collection.

**121. What is the purpose of the lock keyword in C#?**  
a) To prevent access to a method  
b) To synchronize access to a resource  
c) To encrypt data  
d) To define a static method  
✅ **Answer:** b) To synchronize access to a resource  
💡 **Explanation:** lock ensures that only one thread accesses a block of code at a time.

**122. What is a thread in C#?**  
a) A class  
b) A method  
c) A unit of execution  
d) A variable  
✅ **Answer:** c) A unit of execution  
💡 **Explanation:** A thread is the smallest unit of execution in a process.

**123. What is the purpose of the Thread.Sleep() method?**  
a) To stop a thread permanently  
b) To pause a thread temporarily  
c) To kill a thread  
d) To restart a thread  
✅ **Answer:** b) To pause a thread temporarily  
💡 **Explanation:** Thread.Sleep() pauses the current thread for a specified time.

**124. What is the use of the Thread.Join() method?**  
a) To start a thread  
b) To pause a thread  
c) To wait for a thread to finish  
d) To kill a thread  
✅ **Answer:** c) To wait for a thread to finish  
💡 **Explanation:** Join() blocks the calling thread until the thread terminates.

**125. What is the purpose of the async and await keywords together?**  
a) To define a static method  
b) To define a synchronous method  
c) To write asynchronous code  
d) To define a constructor  
✅ **Answer:** c) To write asynchronous code  
💡 **Explanation:** async and await simplify asynchronous programming.

**126. What is the return type of an async method that does not return a value?**  
a) void  
b) Task  
c) Task  
d) None  
✅ **Answer:** b) Task  
💡 **Explanation:** Task is used for async methods that return no value.

**127. What is the return type of an async method that returns a value?**  
a) void  
b) Task  
c) Task  
d) T  
✅ **Answer:** c) Task  
💡 **Explanation:** Task<T> is used for async methods that return a value of type T.

**128. What is the purpose of the ConfigureAwait(false) method?**  
a) To configure a task  
b) To avoid deadlocks  
c) To run code on the UI thread  
d) To delay execution  
✅ **Answer:** b) To avoid deadlocks  
💡 **Explanation:** ConfigureAwait(false) avoids capturing the synchronization context.

**129. What is the use of ValueTask in C#?**  
a) To return multiple values  
b) To return a value synchronously or asynchronously  
c) To return a void  
d) To return a delegate  
✅ **Answer:** b) To return a value synchronously or asynchronously  
💡 **Explanation:** ValueTask is a lightweight alternative to Task.

**130. What is the purpose of the yield return statement?**  
a) To return a value  
b) To return multiple values lazily  
c) To return a delegate  
d) To return a task  
✅ **Answer:** b) To return multiple values lazily  
💡 **Explanation:** yield return enables lazy iteration using IEnumerable.

**131. What is the purpose of the nameof operator in C#?**  
a) To get the value of a variable  
b) To get the name of a variable as a string  
c) To get the type of a variable  
d) To get the namespace  
✅ **Answer:** b) To get the name of a variable as a string  
💡 **Explanation:** nameof(variable) returns "variable" as a string.

**132. What is the use of the is keyword in C#?**  
a) To compare values  
b) To check type compatibility  
c) To assign values  
d) To define a method  
✅ **Answer:** b) To check type compatibility  
💡 **Explanation:** is checks if an object is of a specific type.

**133. What is the use of the as keyword in C#?**  
a) To cast types safely  
b) To compare types  
c) To define a method  
d) To define a class  
✅ **Answer:** a) To cast types safely  
💡 **Explanation:** as performs a safe cast and returns null if it fails.

**134. What is the purpose of the ?? operator in C#?**  
a) Null check  
b) Null coalescing  
c) Null assignment  
d) Null comparison  
✅ **Answer:** b) Null coalescing  
💡 **Explanation:** x = y ?? z assigns z if y is null.

**135. What is the purpose of the ?. operator in C#?**  
a) Null check  
b) Null coalescing  
c) Null conditional access  
d) Null assignment  
✅ **Answer:** c) Null conditional access  
💡 **Explanation:** obj?.Property accesses Property only if obj is not null.

**136. What is the use of the switch expression in C# 8.0+?**  
a) To replace if-else  
b) To simplify pattern matching  
c) To define a loop  
d) To define a method  
✅ **Answer:** b) To simplify pattern matching  
💡 **Explanation:** switch expressions provide concise syntax for branching logic.

**137. What is pattern matching in C#?**  
a) Comparing strings  
b) Matching types and values  
c) Matching regular expressions  
d) Matching methods  
✅ **Answer:** b) Matching types and values  
💡 **Explanation:** Pattern matching allows checking a value’s type and extracting data.

**138. What is the purpose of the record keyword in C# 9.0+?**  
a) To define a class  
b) To define a struct  
c) To define an immutable reference type  
d) To define a method  
✅ **Answer:** c) To define an immutable reference type  
💡 **Explanation:** record types are used for immutable data models.

**139. What is the difference between record and class in C#?**  
a) Records are mutable  
b) Classes are immutable  
c) Records provide value-based equality  
d) Classes are faster  
✅ **Answer:** c) Records provide value-based equality  
💡 **Explanation:** Records compare by value; classes compare by reference.

**140. What is the purpose of the with expression in C# 9.0+?**  
a) To clone objects  
b) To modify immutable objects  
c) To define a method  
d) To define a property  
✅ **Answer:** b) To modify immutable objects  
💡 **Explanation:** with creates a copy of a record with modified properties.

**141. What is the purpose of the init accessor in C# 9.0+?**  
a) To initialize fields in a constructor  
b) To allow property setting only during object creation  
c) To make properties read-only  
d) To define a static property  
✅ **Answer:** b) To allow property setting only during object creation  
💡 **Explanation:** init allows setting a property only during object initialization.

**142. What is the use of the target-typed new expression in C# 9.0+?**  
a) To define a new class  
b) To infer type from the left-hand side  
c) To define a static method  
d) To define a constructor  
✅ **Answer:** b) To infer type from the left-hand side  
💡 **Explanation:** MyClass obj = new(); infers the type from the variable.

**143. What is the purpose of the global using directive in C# 10.0+?**  
a) To import namespaces globally across files  
b) To define a global variable  
c) To define a global method  
d) To define a global class  
✅ **Answer:** a) To import namespaces globally across files  
💡 **Explanation:** global using applies to all files in the project.

**144. What is a top-level statement in C#?**  
a) A statement inside a class  
b) A statement outside any method  
c) A statement at the top of a file  
d) A statement in a namespace  
✅ **Answer:** b) A statement outside any method  
💡 **Explanation:** Top-level statements allow writing code without explicitly defining a Main method.

**145. What is the purpose of the file access modifier in C# 11.0+?**  
a) To restrict access to a file  
b) To allow access only within the same file  
c) To define a file-scoped class  
d) To define a file-scoped namespace  
✅ **Answer:** b) To allow access only within the same file  
💡 **Explanation:** file restricts access to the same source file.

**146. What is the use of Span<T> in C#?**  
a) To define a list  
b) To define a dynamic array  
c) To work with slices of memory  
d) To define a pointer  
✅ **Answer:** c) To work with slices of memory  
💡 **Explanation:** Span<T> provides a type-safe view over memory.

**147. What is the difference between Span<T> and Memory<T>?**  
a) Span is for heap memory, Memory is for stack  
b) Span is for stack, Memory is for heap  
c) Both are the same  
d) Span is slower  
✅ **Answer:** b) Span is for stack, Memory is for heap  
💡 **Explanation:** Span<T> is stack-only; Memory<T> can be used asynchronously and on the heap.

**148. What is the purpose of ref struct in C#?**  
a) To define a reference type  
b) To define a struct that can be boxed  
c) To define a stack-only struct  
d) To define a static struct  
✅ **Answer:** c) To define a stack-only struct  
💡 **Explanation:** ref struct ensures the struct lives on the stack.

**149. What is the use of readonly struct in C#?**  
a) To make struct fields mutable  
b) To prevent struct modification  
c) To define a static struct  
d) To define a sealed struct  
✅ **Answer:** b) To prevent struct modification  
💡 **Explanation:** readonly struct ensures all fields are read-only.

**150. What is the purpose of record struct in C# 10.0+?**  
a) To define a mutable record  
b) To define an immutable struct  
c) To define a class  
d) To define a method  
✅ **Answer:** b) To define an immutable struct  
💡 **Explanation:** record struct combines value semantics with immutability.

**151. What is the use of default literal in C#?**  
a) To assign a default value  
b) To define a default method  
c) To define a default class  
d) To define a default constructor  
✅ **Answer:** a) To assign a default value  
💡 **Explanation:** default assigns the default value of a type.

**152. What is the purpose of default interface methods in C# 8.0+?**  
a) To allow interfaces to have fields  
b) To allow interfaces to have method implementations  
c) To allow interfaces to be inherited  
d) To allow interfaces to be instantiated  
✅ **Answer:** b) To allow interfaces to have method implementations  
💡 **Explanation:** Interfaces can now have default method bodies.

**153. What is the use of static interface members in C# 11.0+?**  
a) To define static methods in interfaces  
b) To define static fields in interfaces  
c) To define static constructors in interfaces  
d) All of the above  
✅ **Answer:** d) All of the above  
💡 **Explanation:** C# 11 allows static members in interfaces.

**154. What is the purpose of required modifier in C# 11.0+?**  
a) To make a property optional  
b) To enforce property initialization  
c) To define a required method  
d) To define a required class  
✅ **Answer:** b) To enforce property initialization  
💡 **Explanation:** required ensures a property must be initialized.

**155. What is the use of file-scoped namespace in C#?**  
a) To define a namespace for a file  
b) To define a global namespace  
c) To define a static namespace  
d) To define a sealed namespace  
✅ **Answer:** a) To define a namespace for a file  
💡 **Explanation:** namespace MyNamespace; applies to the whole file.

**156. What is the purpose of target-typed conditional expressions in C#?**  
a) To simplify ternary expressions  
b) To infer type from context  
c) To define a method  
d) To define a class  
✅ **Answer:** b) To infer type from context  
💡 **Explanation:** The type of condition ? expr1 : expr2 is inferred from the target.

**157. What is the use of InterpolatedStringHandler in C# 10.0+?**  
a) To format strings  
b) To optimize string interpolation  
c) To define a string  
d) To define a handler  
✅ **Answer:** b) To optimize string interpolation  
💡 **Explanation:** It improves performance of interpolated strings.

**158. What is the purpose of CallerMemberName attribute?**  
a) To get the name of the caller method  
b) To get the name of the class  
c) To get the name of the namespace  
d) To get the name of the assembly  
✅ **Answer:** a) To get the name of the caller method  
💡 **Explanation:** Useful for logging and notifications.

**159. What is the use of CallerFilePath attribute?**  
a) To get the file path of the caller  
b) To get the file name  
c) To get the method name  
d) To get the line number  
✅ **Answer:** a) To get the file path of the caller  
💡 **Explanation:** Used for diagnostics and logging.

**160. What is the use of CallerLineNumber attribute?**  
a) To get the line number of the caller  
b) To get the method name  
c) To get the file path  
d) To get the class name  
✅ **Answer:** a) To get the line number of the caller  
💡 **Explanation:** Helps in debugging and tracing.

**161. What will be the output of the following code?**

int[] arr = new int[3];

Console.WriteLine(arr[0]);

a) 0  
b) 5  
c) Compile-time error  
d) Runtime error  
✅ **Answer:** b) 5  
💡 **Explanation:** This is an example of boxing and unboxing. a is boxed into obj, then unboxed back to int.

**162. Which of the following best describes covariance in C#?**  
a) Allows a method to return a more derived type than specified  
b) Allows a method to accept a more derived type than specified  
c) Allows a method to return a less derived type  
d) Allows a method to accept a less derived type  
✅ **Answer:** a) Allows a method to return a more derived type than specified  
💡 **Explanation:** Covariance preserves assignment compatibility for return types.

**163. What is the result of the following code?**

a) Not empty  
b) Empty  
c) Compile-time error  
d) Runtime error  
✅ **Answer:** b) Empty  
💡 **Explanation:** s?.Length returns null, so the condition is false.

**164. Which of the following statements about Task and Thread is true?**  
a) Task is always faster than Thread  
b) Thread is more memory efficient than Task  
c) Task uses ThreadPool and is more scalable  
d) Task cannot be awaited  
✅ **Answer:** c) Task uses ThreadPool and is more scalable  
💡 **Explanation:** Tasks are lightweight and managed by the ThreadPool.

**165. What is the output of the following code?**

a) 10  
b) 11

c) 0  
d) Compile-time error  
✅ **Answer:** a) 10  
💡 **Explanation:** int is a value type. y is a copy of x.

**166. What is the key difference between IEnumerable<T> and IQueryable<T>?**  
a) IEnumerable executes queries on the server  
b) IQueryable executes queries in memory  
c) IQueryable supports deferred execution and translates queries to SQL  
d) IEnumerable supports lazy loading  
✅ **Answer:** c) IQueryable supports deferred execution and translates queries to SQL  
💡 **Explanation:** IQueryable is used for remote data sources like databases.

**167. Which of the following is true about readonly fields in C#?**  
a) They can be assigned only at declaration  
b) They can be assigned in the constructor  
c) They can be modified anywhere  
d) They are equivalent to const  
✅ **Answer:** b) They can be assigned in the constructor  
💡 **Explanation:** readonly fields can be initialized in the constructor or at declaration.

**168. What is the output of the following code?**

a) 0  
b) null  
c) Compile-time error  
d) Runtime error  
✅ **Answer:** a) 0  
💡 **Explanation:** Arrays of value types are initialized to default values.

**169. What is the purpose of yield return in C#?**  
a) To return multiple values at once  
b) To return a value and pause execution  
c) To return a value and exit the method  
d) To return a reference  
✅ **Answer:** b) To return a value and pause execution  
💡 **Explanation:** yield return is used in iterator blocks to return values one at a time.

**170. What is the difference between abstract and virtual methods?**  
a) Abstract methods must have a body  
b) Virtual methods cannot be overridden  
c) Abstract methods must be overridden  
d) Virtual methods must be overridden  
✅ **Answer:** c) Abstract methods must be overridden  
💡 **Explanation:** Abstract methods have no implementation and must be overridden.

**171. Which of the following best describes the purpose of the Common Language Runtime (CLR)?**  
a) Compiles C# code to machine code  
b) Manages memory, security, and execution of .NET programs  
c) Provides UI components  
d) Handles database connections  
✅ **Answer:** b) Manages memory, security, and execution of .NET programs

**172. What is the role of the Just-In-Time (JIT) compiler in .NET?**  
a) Converts IL code to C#  
b) Converts IL code to native machine code at runtime  
c) Converts C# to IL  
d) Converts SQL to LINQ  
✅ **Answer:** b) Converts IL code to native machine code at runtime

**173. Which of the following is NOT a valid value type in C#?**  
a) int  
b) float  
c) string  
d) bool  
✅ **Answer:** c) string

**174. What is the main advantage of using generics in C#?**  
a) Runtime type checking  
b) Code duplication  
c) Type safety and performance  
d) Reflection  
✅ **Answer:** c) Type safety and performance

**175. Which of the following statements about var is true?**  
a) It is dynamically typed  
b) It can be used for any type, including anonymous types  
c) It can be reassigned to a different type  
d) It is resolved at runtime  
✅ **Answer:** b) It can be used for any type, including anonymous types

**176. What is the difference between Task and Thread in .NET?**  
a) Task is lower-level than Thread  
b) Thread is more efficient than Task  
c) Task is higher-level and uses ThreadPool  
d) Thread is used for async programming  
✅ **Answer:** c) Task is higher-level and uses ThreadPool

**177. What does the volatile keyword ensure in multithreading?**  
a) The variable is read from cache  
b) The variable is not shared  
c) The variable is always read from main memory  
d) The variable is immutable  
✅ **Answer:** c) The variable is always read from main memory

**178. What is the purpose of the Monitor class in C#?**  
a) To monitor CPU usage  
b) To manage memory  
c) To provide thread synchronization  
d) To monitor network traffic  
✅ **Answer:** c) To provide thread synchronization

**179. What is the difference between throw and throw ex in exception handling?**  
a) No difference  
b) throw ex preserves the original stack trace  
c) throw resets the stack trace  
d) throw preserves the original stack trace  
✅ **Answer:** d) throw preserves the original stack trace

**180. What is the purpose of the AppDomain class in .NET?**  
a) To manage application memory  
b) To isolate applications within the same process  
c) To manage garbage collection  
d) To manage UI threads  
✅ **Answer:** b) To isolate applications within the same process

**181. What is the difference between Assembly.Load() and Assembly.LoadFrom()?**  
a) Load uses file path, LoadFrom uses name  
b) LoadFrom uses file path, Load uses name  
c) Both are the same  
d) LoadFrom is obsolete  
✅ **Answer:** b) LoadFrom uses file path, Load uses name

**182. What is the purpose of the Activator.CreateInstance() method?**  
a) To create a new thread  
b) To create a new AppDomain  
c) To create an instance of a type at runtime  
d) To create a delegate  
✅ **Answer:** c) To create an instance of a type at runtime

**183. What is the difference between Assembly and Module in .NET?**  
a) Assembly is a single file, Module is multiple files  
b) Module is a part of an Assembly  
c) Assembly is part of a Module  
d) They are the same  
✅ **Answer:** b) Module is a part of an Assembly

**184. What is the purpose of the Type.GetType() method?**  
a) To get the type of a variable  
b) To get the type of a class at runtime  
c) To get the type of a method  
d) To get the type of a namespace  
✅ **Answer:** b) To get the type of a class at runtime

**185. What is the purpose of the typeof operator in C#?**  
a) To get the value of a variable  
b) To get the type of a variable at runtime  
c) To get the compile-time type  
d) To get the namespace  
✅ **Answer:** c) To get the compile-time type

**186. What is the difference between is and typeof in C#?**  
a) is checks type at compile time, typeof at runtime  
b) is checks type at runtime, typeof at compile time  
c) Both are the same  
d) typeof is used for casting  
✅ **Answer:** b) is checks type at runtime, typeof at compile time

**187. What is the purpose of the dynamic keyword in C#?**  
a) To define a variable with a fixed type  
b) To define a variable with a compile-time type  
c) To define a variable with a runtime-resolved type  
d) To define a constant  
✅ **Answer:** c) To define a variable with a runtime-resolved type

**188. What is the difference between dynamic and object in C#?**  
a) dynamic is resolved at compile time  
b) object is resolved at runtime  
c) dynamic bypasses compile-time type checking  
d) object is faster  
✅ **Answer:** c) dynamic bypasses compile-time type checking

**189. What is the purpose of the ExpandoObject in C#?**  
a) To create a static object  
b) To create a dynamic object with runtime properties  
c) To create a sealed object  
d) To create a thread-safe object  
✅ **Answer:** b) To create a dynamic object with runtime properties

**190. What is the use of the Expression<T> class in LINQ?**  
a) To compile expressions  
b) To represent code as data  
c) To execute SQL  
d) To define a delegate  
✅ **Answer:** b) To represent code as data

**191. What is the purpose of Func<T> in C#?**  
a) To define a method  
b) To define a delegate that returns a value  
c) To define a class  
d) To define a property  
✅ **Answer:** b) To define a delegate that returns a value

**192. What is the purpose of Action<T> in C#?**  
a) To define a method  
b) To define a delegate that returns void  
c) To define a class  
d) To define a property  
✅ **Answer:** b) To define a delegate that returns void

**193. What is the difference between Func<T> and Action<T>?**  
a) Func returns a value, Action does not  
b) Action returns a value, Func does not  
c) Both return values  
d) Both return void  
✅ **Answer:** a) Func returns a value, Action does not

**194. What is the purpose of Predicate<T> in C#?**  
a) To define a method  
b) To define a delegate that returns a boolean  
c) To define a class  
d) To define a property  
✅ **Answer:** b) To define a delegate that returns a boolean

**195. What is the purpose of Lazy<T> in C#?**  
a) To delay object creation until needed  
b) To create objects eagerly  
c) To create static objects  
d) To create sealed objects  
✅ **Answer:** a) To delay object creation until needed

**196. What is the use of ThreadLocal<T> in C#?**  
a) To create global variables  
b) To create thread-specific data  
c) To create static variables  
d) To create shared variables  
✅ **Answer:** b) To create thread-specific data

**197. What is the purpose of ConcurrentDictionary in C#?**  
a) To allow multiple threads to read and write safely  
b) To allow only one thread to access  
c) To allow only reading  
d) To allow only writing  
✅ **Answer:** a) To allow multiple threads to read and write safely

**198. What is the difference between lock and Monitor in C#?**  
a) lock is more flexible  
b) Monitor is a keyword  
c) lock is syntactic sugar for Monitor.Enter/Exit  
d) Monitor is slower  
✅ **Answer:** c) lock is syntactic sugar for Monitor.Enter/Exit

**199. What is the purpose of SemaphoreSlim in C#?**  
a) To allow only one thread access to a resource  
b) To allow multiple threads up to a specified limit  
c) To block all threads from accessing a resource  
d) To create a thread-safe queue  
✅ **Answer:** b) To allow multiple threads up to a specified limit  
💡 **Explanation:** SemaphoreSlim is a lightweight synchronization primitive that limits the number of threads that can access a resource concurrently.

**200. What is the primary benefit of Dependency Injection (DI) in .NET applications?**  
a) It increases coupling between components  
b) It makes unit testing harder  
c) It promotes loose coupling and testability  
d) It eliminates the need for interfaces  
✅ **Answer:** c) It promotes loose coupling and testability  
💡 **Explanation:** DI allows objects to receive their dependencies from an external source, making the system more modular, flexible, and easier to test.

**201. Which design pattern encapsulates a request as an object, thereby allowing parameterization of clients with queues, requests, and operations?**  
a) Strategy  
b) Command  
c) Observer  
d) Mediator  
✅ **Answer:** b) Command

**202. Which pattern is best suited for decoupling the sender and receiver of requests?**  
a) Chain of Responsibility  
b) Singleton  
c) Factory  
d) Adapter  
✅ **Answer:** a) Chain of Responsibility

**203. Which pattern provides a simplified interface to a complex subsystem?**  
a) Facade  
b) Proxy  
c) Bridge  
d) Composite  
✅ **Answer:** a) Facade

**204. Which pattern allows an object to alter its behavior when its internal state changes?**  
a) Strategy  
b) State  
c) Observer  
d) Visitor  
✅ **Answer:** b) State

**205. Which pattern is used to ensure a class has only one instance and provides a global point of access to it?**  
a) Singleton  
b) Factory  
c) Builder  
d) Prototype  
✅ **Answer:** a) Singleton

**🔹 WPF / WinForms / ASP.NET**

**206. In WPF, what is the primary purpose of XAML?**  
a) To define logic  
b) To define UI layout  
c) To define database schema  
d) To define routing  
✅ **Answer:** b) To define UI layout

**207. What is the MVVM pattern primarily used for in WPF?**  
a) Data storage  
b) UI rendering  
c) Separation of concerns  
d) Event handling  
✅ **Answer:** c) Separation of concerns

**208. In ASP.NET Core, what is middleware?**  
a) A UI component  
b) A database connector  
c) A component that handles HTTP requests/responses  
d) A logging tool  
✅ **Answer:** c) A component that handles HTTP requests/responses

**209. What is the default dependency injection container in ASP.NET Core?**  
a) Autofac  
b) Ninject  
c) Microsoft.Extensions.DependencyInjection  
d) Unity  
✅ **Answer:** c) Microsoft.Extensions.DependencyInjection

**210. What is Razor in ASP.NET Core?**  
a) A database engine  
b) A templating engine for HTML and C#  
c) A CSS framework  
d) A JavaScript library  
✅ **Answer:** b) A templating engine for HTML and C#

**🔹 Unit Testing & Test Frameworks**

**211. What is the purpose of the [Fact] attribute in xUnit?**  
a) To define a test class  
b) To define a test method  
c) To define a test fixture  
d) To define a test parameter  
✅ **Answer:** b) To define a test method

**212. What is the difference between [Fact] and [Theory] in xUnit?**  
a) [Fact] is for parameterized tests  
b) [Theory] is for parameterized tests  
c) Both are the same  
d) [Fact] is obsolete  
✅ **Answer:** b) [Theory] is for parameterized tests

**213. What is mocking in unit testing?**  
a) Creating real objects  
b) Creating fake objects to simulate behavior  
c) Creating UI components  
d) Creating test databases  
✅ **Answer:** b) Creating fake objects to simulate behavior

**214. Which library is commonly used for mocking in .NET?**  
a) Moq  
b) NUnit  
c) xUnit  
d) MSTest  
✅ **Answer:** a) Moq

**215. What is the purpose of dependency injection in unit testing?**  
a) To increase coupling  
b) To make testing harder  
c) To allow mocking and testability  
d) To reduce performance  
✅ **Answer:** c) To allow mocking and testability

**🔹 Dependency Injection Frameworks**

**216. What is constructor injection?**  
a) Injecting dependencies via method parameters  
b) Injecting dependencies via constructor  
c) Injecting dependencies via properties  
d) Injecting dependencies via fields  
✅ **Answer:** b) Injecting dependencies via constructor

**217. What is the lifetime of a singleton service in DI?**  
a) Per request  
b) Per scope  
c) One instance for the entire application  
d) One instance per thread  
✅ **Answer:** c) One instance for the entire application

**218. What is the lifetime of a scoped service in ASP.NET Core?**  
a) Per request  
b) Per application  
c) Per thread  
d) Per method  
✅ **Answer:** a) Per request

**219. What is the purpose of the IServiceCollection interface?**  
a) To store data  
b) To register services for DI  
c) To manage controllers  
d) To manage routing  
✅ **Answer:** b) To register services for DI

**220. What is the purpose of the IServiceProvider interface?**  
a) To register services  
b) To resolve services  
c) To define services  
d) To dispose services  
✅ **Answer:** b) To resolve services

**🔹 Advanced LINQ**

**221. What does the GroupJoin operator do in LINQ?**  
a) Joins two sequences  
b) Groups elements and joins them  
c) Filters elements  
d) Projects elements  
✅ **Answer:** b) Groups elements and joins them

**222. What is the purpose of the let keyword in LINQ?**  
a) To define a new variable in the query  
b) To filter elements  
c) To join sequences  
d) To sort elements  
✅ **Answer:** a) To define a new variable in the query

**223. What is the purpose of the into keyword in LINQ?**  
a) To define a new query  
b) To continue a query after a group or join  
c) To end a query  
d) To define a method  
✅ **Answer:** b) To continue a query after a group or join

**224. What is deferred execution in LINQ?**  
a) Query is executed immediately  
b) Query is executed when enumerated  
c) Query is executed at compile time  
d) Query is executed in the database  
✅ **Answer:** b) Query is executed when enumerated

**225. What is the difference between Select and SelectMany?**  
a) Select flattens collections  
b) SelectMany flattens collections  
c) Both are the same  
d) SelectMany is obsolete  
✅ **Answer:** b) SelectMany flattens collections

**🔹 Memory Management**

**226. What is a generation in .NET garbage collection?**  
a) A version of .NET  
b) A level of object lifetime  
c) A type of memory  
d) A type of thread  
✅ **Answer:** b) A level of object lifetime

**227. How many generations are there in .NET GC?**  
a) 1  
b) 2  
c) 3  
d) 4  
✅ **Answer:** c) 3

**228. What is the purpose of the finalizer in C#?**  
a) To initialize objects  
b) To clean up unmanaged resources  
c) To override methods  
d) To define constructors  
✅ **Answer:** b) To clean up unmanaged resources

**229. What is the difference between Dispose and finalizer?**  
a) Dispose is automatic, finalizer is manual  
b) Dispose is manual, finalizer is automatic  
c) Both are automatic  
d) Both are manual  
✅ **Answer:** b) Dispose is manual, finalizer is automatic

**230. What is the purpose of GC.SuppressFinalize()?**  
a) To force finalization  
b) To prevent finalization  
c) To delay finalization  
d) To restart finalization  
✅ **Answer:** b) To prevent finalization

**🔹 Security in .NET**

**231. What is the purpose of the SecureString class?**  
a) To encrypt strings  
b) To store sensitive data in memory securely  
c) To hash strings  
d) To compress strings  
✅ **Answer:** b) To store sensitive data in memory securely

**232. What is the purpose of the DataProtection API in ASP.NET Core?**  
a) To protect UI  
b) To protect configuration  
c) To protect data using encryption  
d) To protect routing  
✅ **Answer:** c) To protect data using encryption

**233. What is the purpose of the Authorize attribute in ASP.NET Core?**  
a) To allow anonymous access  
b) To restrict access to authenticated users  
c) To define a controller  
d) To define a route  
✅ **Answer:** b) To restrict access to authenticated users

**234. What is the purpose of the User.Identity.IsAuthenticated property?**  
a) To check if the user is authorized  
b) To check if the user is authenticated  
c) To check if the user is an admin  
d) To check if the user is anonymous  
✅ **Answer:** b) To check if the user is authenticated

**235. What is the purpose of the ClaimsPrincipal class in ASP.NET Core?**  
a) To store user credentials  
b) To store user claims and identity  
c) To store session data  
d) To store cookies  
✅ **Answer:** b) To store user claims and identity  
💡 **Explanation:** ClaimsPrincipal represents the current user and their claims.

**236. What is the purpose of the Claim class in ASP.NET Core?**  
a) To define a user role  
b) To define a user permission  
c) To represent a piece of user identity information  
d) To define a cookie  
✅ **Answer:** c) To represent a piece of user identity information  
💡 **Explanation:** A claim is a key-value pair about the user (e.g., email, role).

**237. What is the purpose of the dotnet new command?**  
a) To build a project  
b) To publish a project  
c) To create a new project or file from a template  
d) To restore packages  
✅ **Answer:** c) To create a new project or file from a template

**238. What does the dotnet build command do?**  
a) Runs the application  
b) Compiles the application  
c) Publishes the application  
d) Creates a new project  
✅ **Answer:** b) Compiles the application

**239. What is the purpose of the dotnet publish command?**  
a) To run the app  
b) To compile the app  
c) To prepare the app for deployment  
d) To create a new solution  
✅ **Answer:** c) To prepare the app for deployment

**240. What is the difference between dotnet run and dotnet build?**  
a) run compiles and runs, build only compiles  
b) build runs the app  
c) run only compiles  
d) Both are the same  
✅ **Answer:** a) run compiles and runs, build only compiles

**241. What is the purpose of the .csproj file in a .NET project?**  
a) To store source code  
b) To define project configuration and dependencies  
c) To store compiled binaries  
d) To define user settings  
✅ **Answer:** b) To define project configuration and dependencies

**242. What is the difference between .NET Framework and .NET Core?**  
a) .NET Core is Windows-only  
b) .NET Framework is cross-platform  
c) .NET Core is cross-platform and open-source  
d) .NET Framework is newer  
✅ **Answer:** c) .NET Core is cross-platform and open-source

**243. What is .NET 5+ (e.g., .NET 6, .NET 7)?**  
a) A continuation of .NET Framework  
b) A continuation of .NET Core  
c) A new language  
d) A UI framework  
✅ **Answer:** b) A continuation of .NET Core  
💡 **Explanation:** .NET 5+ unifies .NET Core and .NET Framework into a single platform.

**244. What is the LTS (Long-Term Support) version of .NET?**  
a) A version with short support  
b) A version with experimental features  
c) A version with extended support and stability  
d) A version for mobile only  
✅ **Answer:** c) A version with extended support and stability

**245. Which of the following is an LTS version?**  
a) .NET 5  
b) .NET 6  
c) .NET 7  
d) .NET 8 Preview  
✅ **Answer:** b) .NET 6  
💡 **Explanation:** .NET 6 is an LTS release; .NET 5 and 7 are STS (Standard-Term Support).

**246. What is the purpose of the Program.cs file in a .NET Core app?**  
a) To define the UI  
b) To define the entry point of the application  
c) To define the database  
d) To define the controller  
✅ **Answer:** b) To define the entry point of the application

**247. What is the purpose of the Startup.cs file in ASP.NET Core?**  
a) To define the main method  
b) To configure services and middleware  
c) To define the UI  
d) To define the database  
✅ **Answer:** b) To configure services and middleware

**248. What is the purpose of the ConfigureServices method in Startup.cs?**  
a) To configure middleware  
b) To configure routing  
c) To register services for DI  
d) To configure logging  
✅ **Answer:** c) To register services for DI

**249. What is the purpose of the Configure method in Startup.cs?**  
a) To configure services  
b) To configure middleware pipeline  
c) To configure logging  
d) To configure database  
✅ **Answer:** b) To configure middleware pipeline

**250. What is the purpose of the app.UseRouting() middleware?**  
a) To enable authentication  
b) To enable routing of HTTP requests  
c) To enable logging  
d) To enable static files  
✅ **Answer:** b) To enable routing of HTTP requests