



variety of questions covering different aspects of C#. Here are some C# questions categorized by topics:

Basics:

1. What is C#?
2. Explain the difference between value types and reference types.
3. What is the purpose of the `Main` method in C#?
4. How do you declare a variable in C#?
5. Explain the difference between `int` and `float` in C#.
6. What is a constant in C#?
7. Describe the concept of boxing and unboxing in C#.
8. What is the significance of the `using` statement in C#?
9. How do you comment a single line and multiple lines in C#?
10. Explain the purpose of the `var` keyword in C#.

Control Flow:

1. What is the difference between `if` and `switch` statements in C#?
2. How do you use the `foreach` loop in C#?
3. Explain the purpose of the `break` and `continue` statements.
4. What is the difference between `while` and `do-while` loops?
5. How can you exit a loop prematurely using a specific condition?

Object-Oriented Programming (OOP):

1. What is encapsulation?
2. Explain the concept of inheritance in C#.
3. What is polymorphism, and how is it implemented in C#?
4. Describe the difference between an interface and an abstract class.
5. What is the `base` keyword used for in C#?
6. How do you implement method overloading in C#?

Message ChatGPT...



ChatGPT can make mistakes. Consider checking important information.



7. What is the purpose of the ``sealed`` keyword in C#?

Exception Handling:

1. How do you handle exceptions in C#?
2. Explain the difference between ``try``, ``catch``, and ``finally`` blocks.
3. What is the purpose of the ``throw`` statement in C#?

LINQ (Language Integrated Query):

1. What is LINQ?
2. Explain the difference between ``IEnumerable`` and ``IQueryable``.
3. How do you use LINQ to filter a collection of objects?
4. What is deferred execution in LINQ?

C# Collections:

1. Describe the difference between ``List`` and ``Array`` in C#.
2. What is the purpose of the ``Dictionary`` class?
3. How do you sort a collection in C#?

File Handling:

1. How do you read from a file in C#?
2. Explain the purpose of the ``FileStream`` class.
3. How do you write to a file in C#?

Delegates and Events:

1. What is a delegate in C#?
2. Explain the purpose of the ``Action`` and ``Func`` delegates.
3. How do you implement events in C#?

Asynchronous Programming:



Message ChatGPT...



ChatGPT can make mistakes. Consider checking important information.



Asynchronous Programming:

1. What is asynchronous programming in C#?
2. Explain the difference between `async` and `await` keywords.
3. How do you handle exceptions in asynchronous code?

Attributes:

1. What are attributes in C#?
2. Provide an example of using custom attributes.

Reflection:

1. What is reflection in C#?
2. How can you use reflection to get information about types?

Dependency Injection:

1. What is dependency injection?
2. How is dependency injection implemented in C#?

Unit Testing:

1. What is NUnit, and how is it used for unit testing in C#?
2. Explain the purpose of the `[SetUp]` and `[TearDown]` attributes in NUnit.

Entity Framework:

1. What is Entity Framework?
2. Explain the difference between Code-First and Database-First approaches in Entity Framework.

These are just the beginning, and you can expand on each topic with more detailed questions based on your level of expertise. If you have specific areas you'd like more questions on, feel free to let me know!



Message ChatGPT...



ChatGPT can make mistakes. Consider checking important information.