**1. Свойства внутренних и вложенных классов**

* **Внутренние классы (inner classes)**: Это классы, определенные внутри другого класса. Они имеют доступ к членам внешнего класса, включая приватные, и могут использоваться для реализации вспомогательных функций.
* **Вложенные классы (nested classes)**: Это классы, которые могут быть определены как внутренние. Могут быть статическими (static nested class) или нестатическими. Статический вложенный класс не имеет доступа к нестатическим членам внешнего класса.

**2. Статический класс и его свойства**

* Статический класс в C# не может быть создан (не имеет экземпляров).
* Все члены статического класса должны быть статическими.
* Статический класс используется для реализации методов или данных, которые не относятся к конкретному объекту.

**3. Назначение перегрузки операторов**

Перегрузка операторов позволяет определить новые способы использования операторов для пользовательских типов (классов и структур). Это позволяет делать код более интуитивно понятным и удобным.

**4. Использование ключевого слова operator**

Ключевое слово operator используется для определения метода, который перегружает оператор. Оно позволяет определить, как будет выполняться операция с операндами пользовательского типа.

**5. Операции, которые нельзя перегружать в C#**

* &&, || (условные логические операторы).
* ?: (тернарный оператор).
* = (оператор присваивания).
* . (точка для доступа к членам).

**6. Можно ли перегрузкой отменить очередность выполнения операции?**

Нет, перегрузка операторов не влияет на порядок выполнения операций. Приоритет операций остается таким же, как и для стандартных операторов.

**7. Может ли операция >= быть перегружена?**

Да, операция >= может быть перегружена.

**8. Сколько аргументов требуется для определения перегруженной унарной операции?**

Для унарных операций требуется один аргумент.

**9. Можно ли перегружать операцию []?**

Нет, в C# операция [] не может быть перегружена, но можно реализовать индексатор, который предоставляет похожую функциональность.

**10. Можно ли перегружать операцию ->?**

Нет, операция -> в C# не поддерживается и не может быть перегружена.

**11. Пример оператора приведения типа**

csharp

Копировать код

public static explicit operator int(Fraction fraction)

{

return fraction.Numerator / fraction.Denominator;

}

**12. Что такое метод расширения? Как и где его можно использовать?**

Метод расширения — это метод, который добавляет новые методы к существующим типам без создания нового производного типа. Используются с ключевым словом this в первом параметре. Могут быть определены только в статических классах.

**13. Форма преобразования**

public static implicit operator Point2D(Point3D a) — это неявное преобразование типа, позволяющее конвертировать объект Point3D в Point2D без явного указания оператора преобразования.

**14. Верное утверждение о методах расширения**

1. Методы расширения могут получать доступ только к public членам расширяемого класса.

**15. Неверное правило перегрузки операторов**

1. Операторы не могут объявляться как protected, они должны быть public static.