**Класс** – основной тип данных языка **C#**. **Класс** представляет собой конструкцию, которая объединяет поля, свойства и методы. **Класс** является определением для создания объектов или экземпляров **класса**.

1. Перечислите свойства внутренних и вложенных классов?

**Вложенный класс** это **класс** определенный внутри **класса**. Вложенный класс должна иметь **public ч**тобы создать объект вложенного класса во внешнем коде.

Зачем нужны вложенные классы C#?

Главной особенностью **вложенных классов** является то, что они могут получать доступ к закрытым членам внешнего **класса**, имея при этом всю мощь самого **класса**.

2. Что такое статический класс и какие у него свойства?

**Статический класс** может использоваться как обычный контейнер для наборов методов, работающих на входных параметрах, и не должен возвращать или устанавливать каких-либо внутренних полей экземпляра.

3. Каково назначение перегрузки операторов?

Перегрузка операторов заключается в определении в классе, для объектов которого мы хотим определить оператор, специального метода:

|  |  |
| --- | --- |
| 1|  2| | public static возвращаемый\_тип operator оператор(параметры)  {  } |

Этот метод должен иметь модификаторы **public static**, так как перегружаемый оператор будет использоваться для всех объектов данного класса. Далее идет название возвращаемого типа. Возвращаемый тип представляет тот тип, объекты которого мы хотим получить.

4. Как используется ключевое слово operator?

public static bool operator <(тип\_параметра операнд)

{

// запросы

}

5. Какие операции нельзя перегружать в C#?

И есть ряд операторов, которые нельзя перегрузить, например, операцию равенства = или тернарный оператор ?:, а также ряд других.

6. Можно ли перегрузкой отменить очередность выполнения операции?

При перегрузке операторов также следует помнить, что мы не можем изменить приоритет оператора или его ассоциативность, мы не можем создать новый оператор или изменить логику операторов в типах, который есть по умолчанию в .NET.

7. Истинно ли следующее утверждение: операция >= может быть перегружена. Да, она перегружается парой с <=

8. Сколько аргументов требуется для определения перегруженной унарной операции?

Если унарная операция перегружается дружественной функцией, то она должна иметь один аргумент – объект, для которого она выполняется.

9. Можно ли перегружать операцию [ ]? Доступ к элементам не считается перегружаемым оператором, но вы можете определить индексатор.

10.Можно ли перегружать операцию ->? нет

11.Приведите пример оператора приведения типа (???)

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

Методы расширения (extension methods) позволяют добавлять новые методы в уже существующие типы без создания нового производного класса. Эта функциональность бывает особенно полезна, когда нам хочется добавить в некоторый тип новый метод, но сам тип (класс или структуру) мы изменить не можем, поскольку у нас нет доступа к исходному коду.

Для того, чтобы создать метод расширения, вначале надо создать статический класс, который и будет содержать этот метод. Затем объявляем статический метод.

Собственно метод расширения - это обычный статический метод, который в качестве первого параметра всегда принимает такую конструкцию: this имя\_типа название\_параметра