

### Как реализовать перегрузку оператора + в языке C#?

```
// Перегрузка бинарного оператора для (Number + Number)
public static Number operator + (Number lhs, Number rhs)
{
    int newNum = lhs.Num + rhs.Num;
    Number Result = new Number(newNum);
    return Result;
    //return new Number(lhs.Num + rhs.Num);
}

// Перегрузка бинарного оператора для (Number + int)
public static Number operator +(Number lhs, int rhs)
{
    return new Number(lhs.Num + rhs);
}
```

Класс содержит свойство Num, которое используется в качестве основы для перегруженных операторов и конструктор, инициализирующий данное свойство.

В большинстве перегруженных операторов для обозначения параметров методов применяют следующие сокращения: lhs – left hand side (левый оператор), rhs – right hand side (правый оператор).

Для перегрузки операторов используется ключевое слово operator. Перегрузка операторов в языке C# может быть реализована только в виде статического метода, который принимает параметр типа Number и возвращает значение типа Number.

В случае перегрузки оператора, содержащего левую и правую части в метод передается два параметра:

```
public static Number operator + (Number lhs, Number rhs)
```

Методы перегрузки операторов поддерживают полиморфизм. Например, можно перегрузить оператор +, который в качестве второго параметра будет принимать тип не Number, а int:

```
public static Number operator + (Number lhs, int rhs)
```

Перегрузка операторов в языке C# предполагает, что возвращаемое значение будет новым объектом, например при перегрузке оператора + новый объект создается прямо в операторе return:  
Return new Number(lhs.Num + rhs);