# Create a 4-bit type called Nybble

```
/*
C#: The Complete Reference
by Herbert Schildt

Publisher: Osborne/McGraw-Hill (March 8, 2002)
ISBN: 0072134852
*/

// Create a 4-bit type called Nybble.

using System;

// A 4-bit type.
class Nybble {
int val; // underlying storage

public Nybble() { val = 0; }

public Nybble(int i) {
val = i;
val = val & 0xF; // retain lower 4 bits
}

// Overload binary + for Nybble + Nybble.
public static Nybble operator +(Nybble op1, Nybble op2)
{
Nybble result = new Nybble();

result.val = op1.val + op2.val;

result.val = result.val & 0xF; // retain lower 4 bits

return result;
}
```

```
// Overload binary + for Nybble + int.
public static Nybble operator +(Nybble op1, int op2)
{
Nybble result = new Nybble();

result.val = op1.val + op2;

result.val = result.val & 0xF; // retain lower 4 bits

return result;
}

// Overload binary + for int + Nybble.
public static Nybble operator +(int op1, Nybble op2)
{
Nybble result = new Nybble();

result.val = op1 + op2.val;

result.val = result.val & 0xF; // retain lower 4 bits

return result;
}

// Overload ++.
public static Nybble operator ++(Nybble op)
{
op.val++;

op.val = op.val & 0xF; // retain lower 4 bits

return op;
}

// Overload >.
public static bool operator >(Nybble op1, Nybble op2)
{
if(op1.val > op2.val) return true;
else return false;
}
```

// Overload <. public static bool operator <(Nybble op1, Nybble op2) { if(op1.val < op2.val) return true; else return false; } // Convert a Nybble into an int. public static implicit operator int (Nybble op) { return op.val; } // Convert an int into a Nybble. public static implicit operator Nybble (int op) { return new Nybble(op); } } public class NybbleDemo { public static void Main() { Nybble a = new Nybble(1); Nybble b = new Nybble(10); Nybble c = new Nybble(); int t; Console.WriteLine("a: " + (int) a); Console.WriteLine("b: " + (int) b); // use a Nybble in an if statement if(a < b) Console.WriteLine("a is less than b "); // Add two Nybbles together c = a + b; Console.WriteLine("c after c = a + b: " + (int) c); // Add an int to a Nybble a += 5; Console.WriteLine("a after a += 5: " + (int) a); Console.WriteLine(); // use a Nybble in an int expression t = a * 2 + 3; Console.WriteLine("Result of a * 2 + 3: " + t); Console.WriteLine(); // illustrate int assignment and overflow a = 19; Console.WriteLine("Result of a = 19: " + (int) a); Console.WriteLine(); // use a Nybble to control a loop Console.WriteLine("Control a for loop with a Nybble."); for(a = 0; a < 10; a++) Console.Write((int) a + " "); Console.WriteLine(); } } [/csharp]