

Материал для подготовки к семинару 2

- Тернарная операция;
- Блочный оператор;
- Условный оператор;
- Цикл с предусловием;
- Цикл с постусловием;
- Логические операции.

Учимся писать код правильно:

- **Coding Conventions:** <https://docs.microsoft.com/ru-ru/dotnet/csharp/programming-guide/inside-a-program/coding-conventions>
- <https://cleancodegame.github.io/>

Тернарная операция (?:)



операнд_1 - логическое выражение (тип bool)

операнд_2 и **операнд_3** - выражения одного типа или выражения, которые могут быть неявно приведены к одному типу.

Пример: вычисление абсолютного значения числа.

$x < 0 ? -x : x$

Блочный оператор

```
int b1_i1 = 1; // блок 1
{
    // блок 1.1
    int b1_1_i1 = 11;
    {
        // блок 1.1.1
        int b1_1_1_i1 = 111;
        Console.WriteLine("b1_i1=" + b1_i1 + ", b1_1_1_i1=" +
b1_1_1_i1);
    }
    {
        // блок 1.1.2
        int b1_1_2_i1 = 112;
        Console.WriteLine("b1_1_i1=" + b1_1_i1 + ", b1_1_2_i1=" +
b1_1_2_i1);
        //Console.WriteLine("b1_1_1_i1=" + b1_1_1_i1);
    }
}
// Console.WriteLine("b1_1_i1=" + b1_1_i1); // ошибка
```

// b1_i1=1, b1_1_1_i1=111

// b1_1_i1=11, b1_1_2_i1=112

Условный оператор (if)

if (логическое_выражение)

оператор_1

else

оператор_2

- опционально

Пример: заменить число на его
абсолютное значение.

```
if (a < 0) a = -a;
```

Условный оператор, пример вложенности

```
int x = 12;  
int y = 18;  
if (x > 10)  
    if (y > 20)  
        Console.WriteLine("Действия при x>10 И y>20");  
    else  
        Console.WriteLine("Действия при x>10 И y<=20");
```

Условный оператор, пример вложенности

```
int x = 12;  
int y = 18;  
if (x > 10)  
{  
    if (y > 20)  
        Console.WriteLine("x>10 И y>20");  
}  
else  
    Console.WriteLine("x<=10 (y - любое)");
```

Цикл с предусловием (while)

while (*выражение-условие*)
тело_цикла

```
int i = 0;
while (i < 5)
    i = i + 1;
Console.Write("i={0}", i);
```

Результат:
i=5

```
int i = 0;
while (i < 5) {
    i = i + 1;
    Console.WriteLine("i={0}", i);
}
Console.WriteLine("i2={0}", i);
```

Результат:
i=1
i=2
i=3
i=4
i=5
i2=5

Цикл с постусловием (do...while)

do

тело_цикла

while (*выражение-условие*);

```
string name;  
do {  
    Console.Write("Введите имя: ");  
    name = Console.ReadLine();  
} while (string.IsNullOrEmpty(name));
```


Логические операции

- ! - отрицание (логическое НЕ, NOT)
- & - конъюнкция (логическое И, AND)
- ^ - исключающее ИЛИ (XOR)
- | - дизъюнкция (логическое ИЛИ, OR)

Проверить принадлежность X отрезку $[E;F]$:

bool factor = $X \geq E \ \& \ X \leq F$;

Проверить принадлежность точки (x,y) кругу с центром в $(0,0)$ и радиусом r :

bool inside = $x*x + y*y < r*r$;

Условные логические операции

Operator	Name	Description
&&	Logical AND	true if both operands are true; false otherwise
	Logical OR	true if at least one operand is true; false otherwise

Объяснение действия “short circuit”:

$x \ \&\& \ y$ is evaluated as $(\text{bool})x \ ? \ (\text{bool})y \ : \ \text{false}$

$x \ || \ y$ эквивалентно $(\text{bool})x \ ? \ \text{true} \ : \ (\text{bool})y$

Условные логические операции

Пример (пропуск побочного эффекта):

```
bool bVal; int iVal = 10;
```

```
bVal = (1 == 2) && (9 == iVal++);    // result: bVal = False, iVal = 10
```

 ↑ ↑
 False Never evaluated

Пример 2:

```
double x= 5, y = 8, z = 2;
```

```
bool sign = x<y+z && y<x+z && z<x+y;
```

Переменная **sign** получит значение **false**
после вычисления **y<x+z**.