**Вводный урок по C# 2.**

**Вывод Данных. Практика.**

**Закрепляем материал | Практика**

**Ошибки:**

Самая распространенная ошибка на начальном этапе обучения — это забытые кавычки, точка с запятой или фигурная скобка.

**Практикуемся:**

**Задача 1:**

Найдите и исправьте ошибки в коде, чтобы консоль вывела все 4 аргумента:

```csharp

using System;

namespace FirstApp

{

class Program

{

static void Main(string[] args)

{

console.WriteLine("row1");

Console.WriteLine(row2);

Console.writeline("row3");

Console.WriteLine('row4');

}

}

}

```

**Ответ:**

```csharp

using System;

namespace FirstApp

{

class Program

{

static void Main(string[] args)

{

Console.WriteLine("row1");

Console.WriteLine("row2");

Console.WriteLine("row3");

Console.WriteLine("row4");

}

}

}

```

**Задача 2: Половина ёлки.**

Напишите программу, которая выводит половину елки, состоящую из звездочек (\*).

**Шаблон:**

```csharp

using System;

public class MainClass

{

public static void Main()

{

// put your c# code here

}

}

```

**Ответ:**

```csharp

using System;

public class MainClass

{

public static void Main()

{

Console.WriteLine("\*");

Console.WriteLine("\*\*");

Console.WriteLine("\*\*\*");

Console.WriteLine("\*\*\*\*");

}

}

```

**Закрепление материала:**

1. **Вывод чисел и математических выражений:**

Чтобы вывести на консоль данные, их нужно обрамлять двойными кавычками. Если мы хотим вывести на консоль число, то можем указать его без кавычек. Это работает только для чисел:

```csharp

static void Main(string[] args)

{

Console.WriteLine("25");

Console.Write(25);

}

```

**Результат на консоли будет идентичен:**

```

25

25

```

Кроме того, в аргументах команд вывода информации можно писать математические выражения:

```csharp

static void Main(string[] args)

{

Console.WriteLine(40 + 10);

Console.WriteLine(40 - 10);

Console.WriteLine(40 \* 10);

Console.WriteLine(40 / 10);

}

```

Сначала будет вычисляться результат математического выражения, а затем выводиться на консоль. Вычисления будут производиться по правилам математики.

1. **Приоритет операций:**

Можно писать сложные математические выражения, состоящие из нескольких операций. Приоритет операций такой же, как в математике: сначала выполняются умножение и деление, затем сложение и вычитание.

```csharp

static void Main(string[] args)

{

Console.WriteLine(40 + 10 \* 2 - 9 / 3); // Вывод: 57

}

```

Изменить приоритет операций можно с помощью круглых скобок:

```csharp

static void Main(string[] args)

{

Console.WriteLine((40 + 10) \* 2 - 9 / 3); // Вывод: 97

}

```

1. **Форматированный вывод:**

Часто требуется выводить числа и строки вместе. Для этого строку нужно обозначить в кавычках, а между числом и строкой поставить знак плюс (+), который соединяет строку и число.

Пример:

```csharp

static void Main(string[] args)

{

Console.WriteLine("40 + 10 \* 2 - 9 / 3 = " + (40 + 10 \* 2 - 9 / 3));

Console.WriteLine("(40 + 10) \* 2 - 9 / 3 = " + ((40 + 10) \* 2 - 9 / 3));

}

```

Если мы знаем результат, можем заменить математическое выражение на число:

```csharp

static void Main(string[] args)

{

Console.WriteLine("40 + 10 \* 2 - 9 / 3 = " + 57);

Console.WriteLine("(40 + 10) \* 2 - 9 / 3 = " + 97);

}

```

1. **Правила сложения строк и чисел**

**Строка + число:**

Если к строке прибавить число, результатом будет строка, которая является их склеиванием.

Примеры:

```csharp

static void Main(string[] args)

{

Console.WriteLine("Мне " + 28);

Console.WriteLine("Я родился в " + 1995);

Console.WriteLine("Я люблю число " + 7);

}

```

Результат выполнения программы:

```

Мне 28

Я родился в 1995

Я люблю число 7

```

Обратите внимание, что внутри строки последним символом является пробел, чтобы вывод был читаемым. Если убрать пробелы внутри строки, результат будет выглядеть менее аккуратно:

```csharp

static void Main(string[] args)

{

Console.WriteLine("Мне" + 28);

Console.WriteLine("Я родился в" + 1995);

Console.WriteLine("Я люблю число" + 7);

}

```

Результат:

```

Мне28

Я родился в1995

Я люблю число7

```

**Число + строка:**

Если к числу прибавить строку, результатом также будет строка, которая является их склеиванием.

Примеры:

```csharp

static void Main(string[] args)

{

Console.WriteLine(5 + " яблок");

Console.WriteLine(18 + " студентов в 12 группе");

Console.WriteLine(80000 + " студентов на курсе");

}

```

Результат выполнения программы:

```

5 яблок

18 студентов в 12 группе

80000 студентов на курсе

```

Обратите внимание на пробелы для читаемости.

**Строка + строка:**

Если к строке прибавить другую строку, результатом будет строка, полученная путем их склеивания.

Пример:

```csharp

static void Main(string[] args)

{

Console.WriteLine("Привет " + "мир");

}

```

Результат выполнения программы:

```

Привет мир

```

1. **Вывод сложных выражений**

При выводе сложных выражений, где смешиваются числа и строки, важно соблюдать порядок выполнения операций слева направо.

Пример:

```csharp

static void Main(string[] args)

{

Console.WriteLine("5 + 7 = " + 12); // 5 + 7 = 12

Console.WriteLine(5 + " + " + 7 + " = " + 12); // 5 + 7 = 12

Console.WriteLine(3 + " " + 6); // вывод двух чисел через пробел, то есть 3 6

}

```

* Заметьте, что пробел (" ") — это тоже строка, причем не пустая. Один и тот же результат можно достичь разными способами.