**Методичка C# Урок 5: Ввод данных. Целые числа.**

**Введение**

В прошлом уроке мы изучили понятия «входные данные» и «строковый тип данных», а также особенности алгоритмов статической программы и программы с обратной связью. В этом уроке мы продолжим обогащать наши знания и навыки программирования.

В ходе урока вы:

1. Изучите особенности конвертации из строкового формата (string) в числовой формат (int).
2. Узнаете о типичных ошибках, совершаемых новичками.

3. Потренируете навыки преобразования строковых переменных в целочисленные типы данных с помощью практических задач.

**Проблема**

Метод `Console.ReadLine()` считывает данные от пользователя в виде строки. Независимо от того, что введет пользователь (буквы, цифры, специальные символы), программа воспринимает это как строку. Если данные нужно использовать для математических операций, их необходимо преобразовать из строкового формата в числовой.

Пример:

```csharp

int a = 5;

int b = 10;

Console.WriteLine(a + b); // 15

string s1 = "5";

string s2 = "10";

Console.WriteLine(s1 + s2); // 510

```

Этот пример показывает, что операции с разными типами данных ведут себя по-разному.

**Решение**

Для преобразования строки в целое число используйте метод `Convert.ToInt32()` . Он принимает строку в качестве аргумента и возвращает целое число:

```csharp

string s = Console.ReadLine(); // Получили строку от пользователя

int number = Convert.ToInt32(s); // Преобразовали в целое число

```

Можно сократить до одной строки:

```csharp

int number = Convert.ToInt32(Console.ReadLine()); // Вводим число

```

Пример программы:

```csharp

using System;

public class MainClass

{

public static void Main()

{

int num1 = Convert.ToInt32(Console.ReadLine()); // Ввели первое число

int num2 = Convert.ToInt32(Console.ReadLine()); // Ввели второе число

Console.WriteLine(num1 + num2); // Вывели сумму

}

}

```

**Запомните:**

Для считывания одного целого числа используйте:

```csharp

int number = Convert.ToInt32(Console.ReadLine());

```

**Типичные ошибки новичков:**

Забывают менять тип данных переменной с `string` на `int` .

- Если нет конвертации, переменная должна быть строковой:

```csharp

string s = Console.ReadLine();

```

- Если есть конвертация, переменная должна быть целочисленной:

```csharp

int s = Convert.ToInt32(Console.ReadLine());

```

Другие методы конвертации:

Использование `int.Parse()` :

```csharp

int number = int.Parse(Console.ReadLine());

```

Этот метод также преобразует строку в целое число и требует, чтобы тип данных переменной был `int` .

**Практические задания**

1. **Как перевести (преобразовать) переменную строкового типа s в целочисленный тип данных?**

Выберите один вариант из списка:

1. int a = Convert.ToBoolean(s);
2. int a = Convert.ToString(s);
3. int a = Convert.ToInt32(s);
4. int a = Convert.ToDouble(s);

Ответ: 3) int a = Convert.ToInt32(s);

1. **Как считать целое число с консоли в переменную number?**

Выберите один вариант из списка:

1. int number = Convert.ToInt32(Console.ReadLine());
2. string number = Convert.ToInt32(Console.ReadLine());
3. int number = Convert.ToInt32(Console.WriteLine());
4. int number = Convert.ToString(Console.ReadLine());

Ответ: 1) int number = Convert.ToInt32(Console.ReadLine());

1. Как считать целое число с консоли в переменную number?

Выберите один вариант из списка:

1. int number = int.Parse(Console.WriteLine());
2. string number = int.Parse(Console.ReadLine());
3. int number = int.Parse(Console.ReadLine());
4. int number = string.Parse(Console.ReadLine());

Ответ: 3) int number = int.Parse(Console.ReadLine());

**Задачи на программирование:**

1. **«Квадрат числа»**

Условие: По данному целому числу найдите его квадрат.

Формат входных данных: На вход дается одно целое число, по модулю не превосходящее 100.

Формат выходных данных: Программа должна вывести квадрат данного числа.

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()

{

int number = Convert.ToInt32(Console.ReadLine());

Console.WriteLine(number \* number);

}

}

```

1. **«Расстояние в километрах»**

Условие: Напишите программу, которая находит полное число километров по заданному числу метров.

Формат входных данных: На вход программе подаётся натуральное число – количество метров.

Формат выходных данных: Программа должна вывести одно число – полное число километров.

Примечание: помните, что в 1 км = 1000 м.

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()

{

int number = Convert.ToInt32(Console.ReadLine());

Console.WriteLine(number / 1000);

}

}

```

1. **Три последовательных числа**

Условие: Напишите программу вывода на экран трех последовательно идущих чисел, каждое на отдельной строке. Первое число вводит пользователь, остальные числа вычисляются в программе.

Формат входных данных: На вход программе подается одно целое число.

Формат выходных данных: Программа должна вывести три последовательно идущих числа в соответствии с условием задачи.

Пример вывода:

```

8

9

10

```

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()

{

int number = Convert.ToInt32(Console.ReadLine());

Console.WriteLine(number);

Console.WriteLine(number + 1);

Console.WriteLine(number + 2);

}

}

```

1. **«Стоимость покупки»**

Условие: Напишите программу, которая считает стоимость трех комплектов телефона. Один комплект состоит из самого устройства, чехла, зарядки и наушников.

Формат входных данных: На вход программе подаётся четыре целых числа, каждое на отдельной строке. В первой строке — стоимость самого устройства, во второй строке — стоимость чехла, в третьей строке — стоимость зарядки и в четвертой строке — стоимость наушников.

Формат выходных данных: Программа должна вывести одно число – стоимость покупки (трех комплектов телефона).

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()

{

int a = Convert.ToInt32(Console.ReadLine());

int b = Convert.ToInt32(Console.ReadLine());

int c = Convert.ToInt32(Console.ReadLine());

int d = Convert.ToInt32(Console.ReadLine());

Console.WriteLine((a + b + c + d) \* 3);

}

}

```

**Бонусные задания для продвинутых**

1. **«Стоимость покупки 2»**

Условие: Пирожок в столовой стоит *a* рублей и *b* копеек. Определите, сколько рублей и копеек нужно заплатить за *nn* пирожков.

Формат входных данных: Программа получает на вход три целых числа: a,b,n.

Формат выходных данных: Программа должна вывести два числа: стоимость покупки в рублях и копейках.

Пример вывода:

На вход:

```

2

20

2

```

На выходе:

```

11 0

```

Шаблон:

```csharp

using System;

public class MainClass

{

public static void Main()

{

// put your c# code here

}

}

```

Подсказки:

1. Переводите рубли в копейки.
2. Суммируете копейки и умножайте на количество пирожков.
3. Для вывода суммы рублей используете  деление нацело (т.е. берете **результат пункта два** и делите нацело на число копеек в одном рубле).
4. Для вывода суммы копеек используете остаток от деления (т.е. берете **результат пункта два** и применяете остаток от деления).

Ответ:

```csharp

using System;

public class MainClass

{

public static void Main()

{

int a = Convert.ToInt32(Console.ReadLine());

int b = Convert.ToInt32(Console.ReadLine());

int n = Convert.ToInt32(Console.ReadLine());

a = (a \* 100) + b; // Переводим рубли в копейки

Console.WriteLine((a \* n / 100) + " " + (a \* n % 100));

}

}

```

1. **«Пересчет временного интервала»**

Условие: Напишите программу для пересчёта величины временного интервала, заданного в минутах, в величину, выраженную в часах и минутах.

Формат входных данных: На вход программе подаётся целое число – количество минут.

Формат выходных данных: Программа должна вывести текст в соответствии с условием задачи.

Пример вывода:

На вход:

```

150

```

На выходе:

```

150 мин – это 2 час 30 минут.

```

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()

{

int number = Convert.ToInt32(Console.ReadLine());

int hours = number / 60;

int minutes = number % 60;

Console.WriteLine($"{number} мин - это {hours} час {minutes} минут.");

}

}

```

**БОНУС:**

**Игра "Угадай число"**

Описание игры: Компьютер загадывает случайное число от 1 до 100, а игрок должен его угадать. После каждой попытки программа сообщает, больше или меньше загаданное число.

1. Создание проекта в VS CODE:

```bash

dotnet new console -o GuessTheNumberAPP

```

1. Код в модуле `Program.cs` :

```csharp

using System;

class Program

{

static void Main()

{

Random random = new Random();

int numberToGuess = random.Next(1, 101); // Загадать число от 1 до 100

int numberOfTries = 0;

int userGuess = 0;

Console.WriteLine("Добро пожаловать в игру 'Угадай число'!");

Console.WriteLine("Я загадал число от 1 до 100. Попробуйте угадать его!");

while (userGuess != numberToGuess)

{

Console.Write("Введите ваше предположение: ");

string userInput = Console.ReadLine();

// Проверка на корректность ввода

if (int.TryParse(userInput, out userGuess))

{

numberOfTries++;

if (userGuess < numberToGuess)

{

Console.WriteLine("Загаданное число больше вашего предположения.");

}

else if (userGuess > numberToGuess)

{

Console.WriteLine("Загаданное число меньше вашего предположения.");

}

else

{

Console.WriteLine($"Поздравляю! Вы угадали число {numberToGuess} за {numberOfTries} попыток.");

Console.WriteLine("Нажмите любую клавишу, чтобы выйти...");

Console.ReadKey(); // Ожидание нажатия клавиши перед закрытием

}

}

else

{

Console.WriteLine("Пожалуйста, введите корректное число.");

}

}

}

}

```

1. Перейдите в директорию GuessTheNumberAPP:

```bash

cd GuessTheNumberAPP

```

1. Соберите проект (преобразуйте в исполняемый файл):

```bash

dotnet publish -c Release -o ./output // Это создаст исполняемый файл в папке output.

```

**Как это работает:**

1. Запустите ваш .exe файл.

2. Введите имя и нажмите Enter.

3. Программа выведет приветствие.

4. После этого она попросит вас нажать любую клавишу, и только после этого консоль закроется.

**Заключение**

В этом уроке мы изучили, как вводить данные от пользователя, преобразовывать строки в целочисленные типы данных и избегать распространенных ошибок. Вы также получили возможность попрактиковаться в программировании через различные задачи.

Не забывайте, что работа с вводом данных — это важный аспект программирования, и правильное преобразование типов данных поможет избежать многих ошибок в будущем. Если у вас возникнут вопросы или вам потребуется помощь, не стесняйтесь обращаться!

Удачи в программировании! 😊