

Microsoft® Small Basic

Переменные

Предполагаемое время работы с этим уроком: 1 час



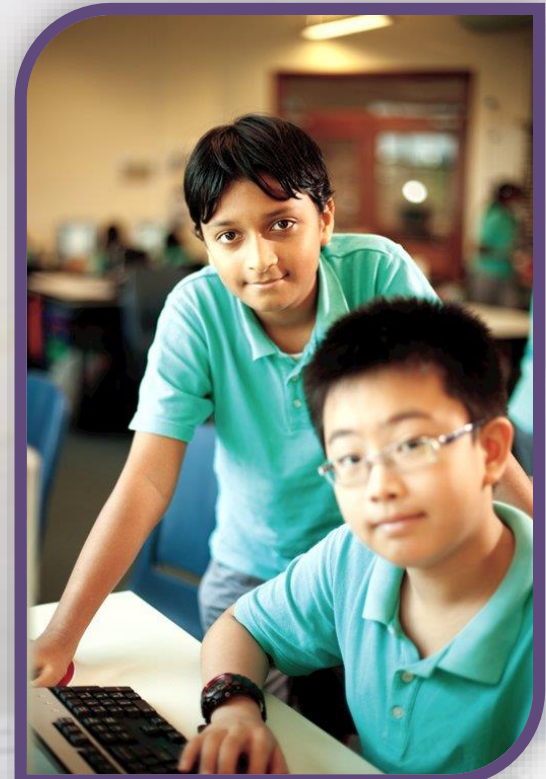
Переменные

В этом уроке вы изучите следующее.

Определение и присвоение имени переменной.

Использование переменных для сохранения текста и цифр.

Использование массивов для сохранения нескольких значений.



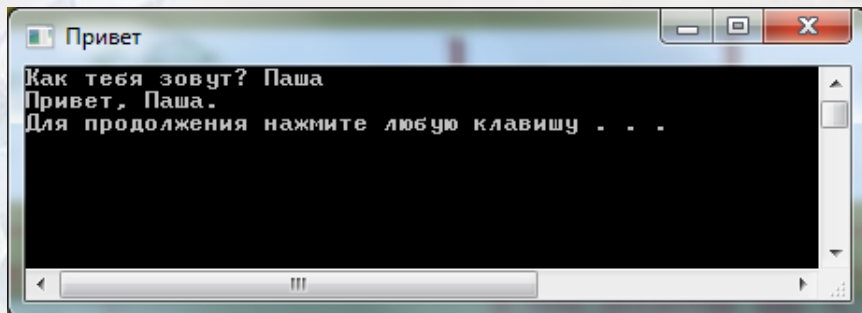
Что такое переменная?

Переменные можно использовать для хранения различной информации, например, текста или чисел. Переменные могут содержать различные значения в разные моменты времени. Большинство переменных не могут одновременно содержать несколько значений. Однако специальные переменные, называемые массивами, могут содержать несколько значений. Рассмотрим программу, в которой создается переменная для хранения имени пользователя.

В этом примере программа запрашивает ввод имени пользователя. Программа использует переменную под названием «name» для хранения информации.

```
TextWindow.Write("Как тебя зовут? ")  
name = TextWindow.Read()  
TextWindow.WriteLine("Привет, " + name + ".")
```

Нажмите кнопку  на панели управления. Запуск (F5)



При выполнении программы отображается «Привет» и информация, сохраненная в переменной.

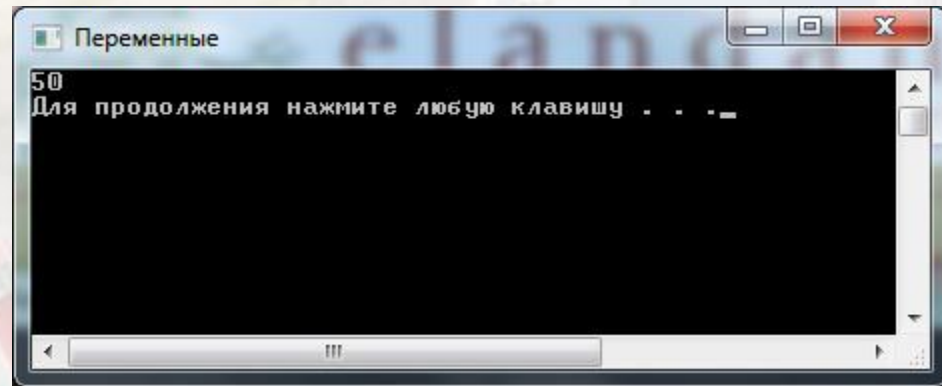
Как называть переменные?

При создании переменных необходимо следовать следующим правилам и рекомендациям.

- ❖ Имена переменных всегда должны начинаться с буквы.
- ❖ В именах переменных можно использовать буквы, цифры и знаки подчеркивания.
- ❖ Имена переменных должны описывать хранящиеся в них значения.
- ❖ При присвоении имен переменным в них не следует включать зарезервированные слова, такие как **If**, **For** и **Then**.

```
number_1 = 20  
number_2 = 30  
number_sum = number_1 + number_2  
TextWindow.WriteLine(number_sum)
```

ВЫВОД

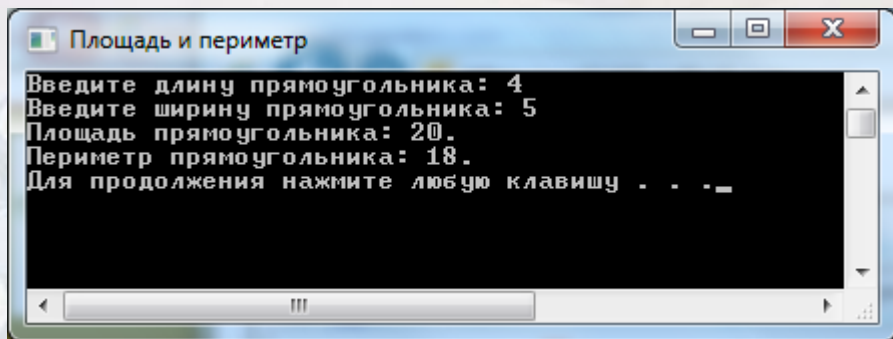


Сохранение числовых значений в переменной

Для лучшего понимания переменных для хранения чисел напишем простую программу, вычисляющую площадь и периметр прямоугольника.

```
TextWindow.Write("Введите длину прямоугольника: ")
length = TextWindow.ReadNumber()
TextWindow.Write("Введите ширину прямоугольника: ")
width = TextWindow.ReadNumber()
area = length * width
perimeter = 2 * length + 2 * width
TextWindow.WriteLine("Площадь прямоугольника: " + area + ".")
TextWindow.WriteLine("Периметр прямоугольника: " + perimeter + ".")
```

ВЫВОД



```
Площадь и периметр
Введите длину прямоугольника: 4
Введите ширину прямоугольника: 5
Площадь прямоугольника: 20.
Периметр прямоугольника: 18.
Для продолжения нажмите любую клавишу . . .
```

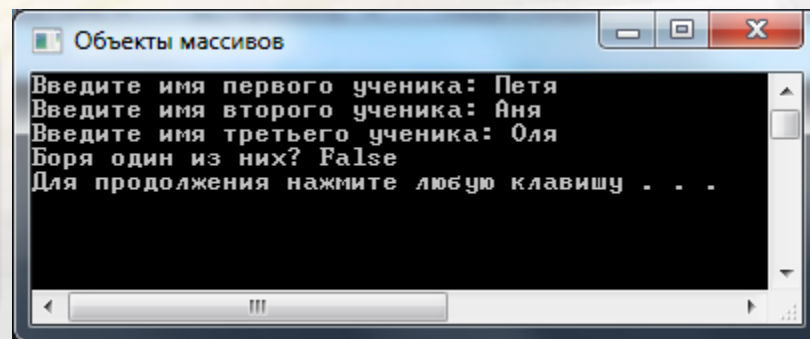
Программа запрашивает у пользователя длину и ширину прямоугольника. При нажатии пользователем клавиши ВВОД программа вычисляет и отображает значения площади и периметра прямоугольника.

Сохранение нескольких значений в переменной

Можно сохранить несколько значений одного типа в одной переменной, используя массив. Массив — это тип переменной, в которой одновременно может содержаться несколько значений. Рассмотрим пример использования массива.

```
TextWindow.Write("Введите имя первого ученика: ")
students[1] = TextWindow.Read()
TextWindow.Write("Введите имя второго ученика: ")
students[2] = TextWindow.Read()
TextWindow.Write("Введите имя третьего ученика: ")
students[3] = TextWindow.Read()
TextWindow.WriteLine("Боря один из них? " + Array.ContainsValue (students, "Боря"))
```

В этом примере создается массив с именем **students**, в котором сохраняются три различных имени. Затем можно получить сохраненные значения, используя различные операции объекта **Array**.



```
Объекты массивов
Введите имя первого ученика: Петя
Введите имя второго ученика: Аня
Введите имя третьего ученика: Оля
Боря один из них? False
Для продолжения нажмите любую клавишу . . .
```

В программах можно использовать некоторые другие функции объекта **Array**. Эти функции будут подробно рассмотрены далее.



Поздравляем! Вы изучили следующее.

- ✚ Создание и присвоение имен переменным и написание инструкций, содержащих переменные.
- ✚ Использование переменных для сохранения текста и цифр.
- ✚ Использование массивов для хранения нескольких значений одного типа.

Продемонстрируйте свои знания

Напишите программу, вычисляющую площадь и длину окружности круга на основе его диаметра.

- ❖ Запросите у пользователя диаметр круга.
- ❖ Создайте переменную с именем **diameter** (диаметр) и сохраните в ней введенное пользователем значение.
- ❖ Создайте переменную под названием **radius** (радиус), вычислите радиус круга и сохраните результат в этой переменной.
- ❖ Создайте переменные под названием **area** (площадь) и **circumference** (длина окружности), вычислите площадь и длину окружности круга и сохраните результаты в этих значениях.
- ❖ Выведите на экран площадь и длину окружности круга.

