

# Типы данных.

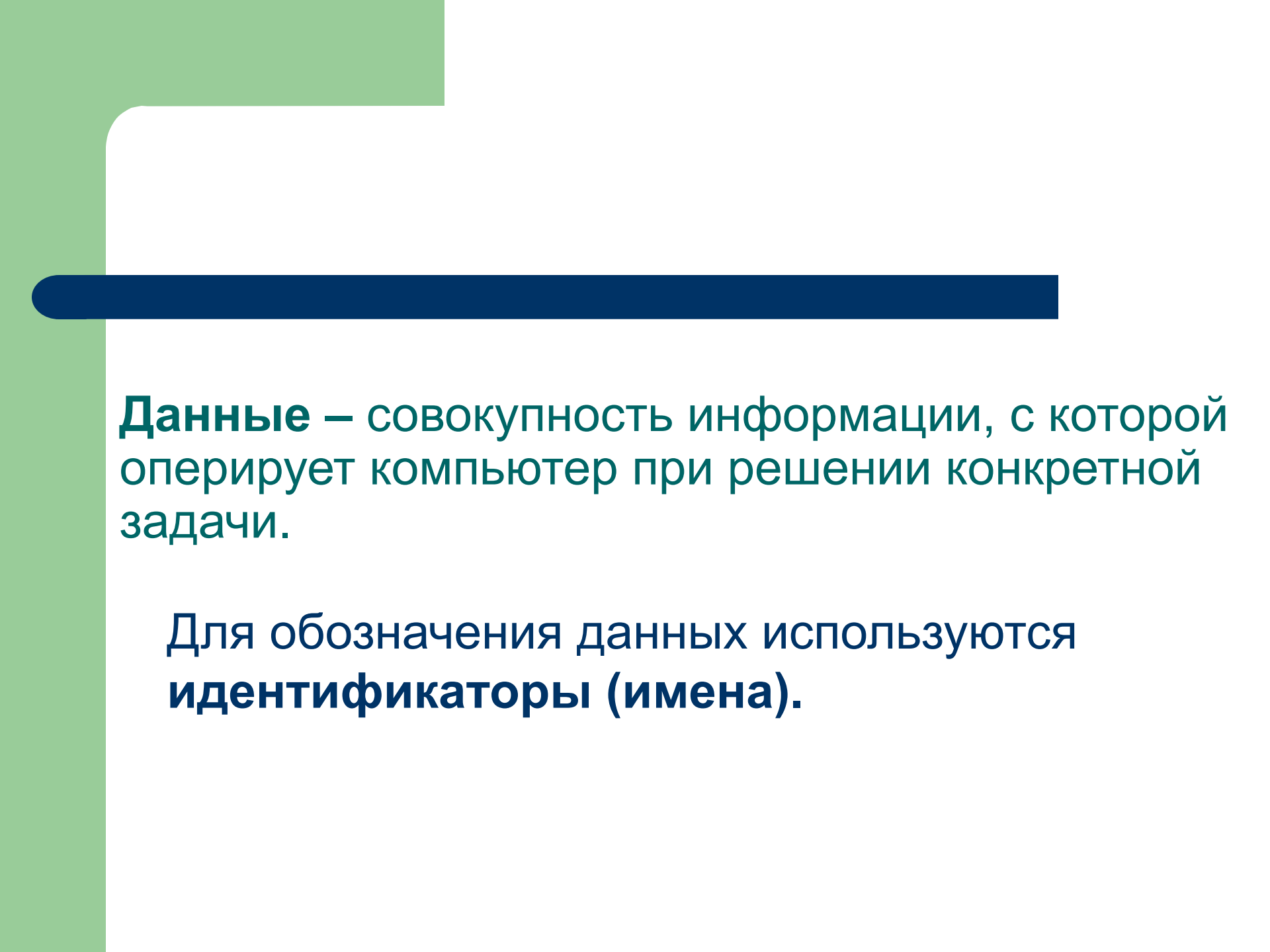
Программирование на  
Паскале.



Функционирование любой программы связано с обработкой данных.

Данные, предназначенные для обработки, называются **ИСХОДНЫМИ** и задаются обычно в начале программы.

В процессе выполнения программы исходные данные преобразуются в **результаты**.



**Данные** – совокупность информации, с которой оперирует компьютер при решении конкретной задачи.

Для обозначения данных используются **идентификаторы (имена)**.

**Идентификатор** – последовательность английских букв, цифр и знака нижней черты, которая начинается с буквы. В качестве идентификатора нельзя использовать зарезервированные слова Паскаля.

Например: K, a12, Elena, Number\_1

```
graph TD; A[Данные] --- B[Константы]; A --- C[Переменные];
```

Данные

Константы

- постоянные величины, значения которых не меняются в процессе выполнения программы

Переменные

- такие данные, которые могут меняться в процессе выполнения программы.

# Константы

**Константы - это постоянные величины, значения которых не меняются в процессе выполнения программы.**

Константы описываются в блоке описания констант в следующем виде:

**CONST** <идентификатор>=<значение константы>;

*Например:*

**CONST** {Служебное слово, с которого начинается блок описания констант}

PI=3.14;

G=9.8;

Year=2009;

Gr='Я'; {символьные константы заключаются в апострофы}

# Переменные

Переменные - это такие данные, которые могут меняться в процессе выполнения программы.

Описание переменных имеет следующий вид:

```
VAR <идентификатор>:<тип переменной>;
```

Под **типом данных** понимается множество допустимых значений этих данных, а также совокупность операций над ними.

переменные

простые

Предназначены для хранения одного значения в данный момент времени.

В оперативной памяти компьютера отводится одна ячейка памяти.

*Например:*

Оперативная память



Идентификаторы (имена)

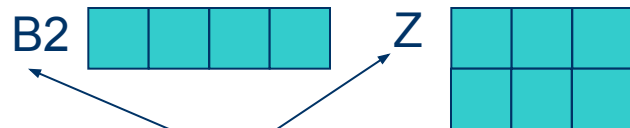
сложные

Предназначены для хранения нескольких значений в данный момент времени.

В оперативной памяти компьютера отводится много ячеек памяти.

*Например:*

Оперативная память



Идентификаторы (имена)



# Стандартные типы простых переменных:

Типы

Числовые

Символьны  
е

Логические

Целые

Вещественные

?

?

?

?

*Задание:* Вместо знаков вопроса записать служебные слова, предназначенные для описания этих типов (найти дальше)

# Целый тип

Переменные целого типа могут содержать только целые числовые значения.

Примером величин, которые могут принимать только целое значение может быть:

год рождения,  
номер квартиры,  
количество детей и т.п.

Константы целого типа записываются так же, как в математике: 234 , 10000 , -11111 , 9999 .

# Целый тип

В языке Pascal имеется несколько целых типа:

Тип	Диапазон	Размер в байтах
SHORTINT	-128; 127	1
<b>INTEGER</b>	<b>-32768; 32767</b>	2
LONGINT	-2147483648; 2147483647	4
BYTE	0; 65	1
WORD	0; 65535	2

# Вещественный тип

Вещественный тип данных используется для описания вещественных переменных, т.е. таких переменных, значением которых может быть вещественное или целое число.

Например: 19.56 , -879.17, 0.05.

Следует обратить внимание на то, что в записи вещественных чисел вместо десятичной запятой (как принято в математике), используется **ДЕСЯТИЧНАЯ ТОЧКА!**

Очень большие и очень маленькие числа в математике принято записывать в специальной форме с помощью умножения значащих цифр на степень 10. В языке Pascal такая запись называется *записью числа в плавающем виде* и имеет вид:

Обычная запись

$$1.3652 \cdot 10^8$$

$$9.109 \cdot 10^{-28}$$

Запись в плавающем виде:

$$1.3652E+8$$

$$9.109E-28$$

# Вещественный тип

В языке Pascal определено четыре стандартных вещественных типа

Тип	Диапазон значений	Значащих цифр	Размер в байтах
REAL	2.9E-39;1.7E38	12	6
SINGLE	1.5E-45;3.4E38	7	4
DOUBLE	5E-324;1.7E308	16	8
EXTENDED	3.4-E4392;1.1E4932	20	10

**Задание:** Запишите диапазон значений типа REAL так, как вы записываете числа на математике.

# СИМВОЛЬНЫЙ ТИП

Символьный тип используется для описания данных, значением которых может быть любой символ, имеющийся на клавиатуре ЭВМ. Если точнее, то полный набор ASCII-символов.

Для описания символьных данных используется ключевое слово **CHAR**.

Константы символьного типа обязательно ЗАКЛЮЧАЮТСЯ в АПОСТРОФЫ. *Например:*

'F' '@' 'Ю' '7'

Значением переменной символьного типа может быть ОДИН СИМВОЛ (буква, цифра, знак препинания и т.п.).

*Например:*

```
VAR CLASS: CHAR; {Объявлена переменная, значением которой может быть один символ}
```

В памяти компьютера под переменную данного типа отводится один байт.

# Логический тип

Логический или, так называемый, булевский тип может принимать одно из двух значений:  
True (истина) или False (ложь).

Для описания логических переменных используется ключевое слово **BOOLEAN**.

Над значениями логического типа допустимы операции сравнения, причем считается, что

`False < True`

Значения логического типа занимают 1 байт памяти.

## Пример описания данных:

Пусть в программе требуется обрабатывать следующую информацию: число и номер месяца, количество отгруженных вагонов, вес вагона, номер вагона, литера вагона (одна буква) и общий вес вагонов. Как следует описать эти данные?



Для начала необходимо для каждой переменной придумать название - идентификатор.

Идентификатор	Смысл переменной
Date	число месяца
Month	номер месяца
Kol	количество отгруженных вагонов
Wes	вес вагона
Number	номер вагона
Lit	литера вагона (одна буква)
Sum	общий вес вагонов

## Пример описания данных:

Далее следует проанализировать, какие значения могут принимать эти данные? В результате блок описания переменных может получить следующий вид:

**VAR**

Date,Month,Kol,Number: INTEGER;

Wes,Sum: REAL;

lit: CHAR;

# Задание:

Запишите раздел описания для переменных:

Идентификатор	Смысл переменной
Num	Номер
A,B,C	Длины сторон треугольника
S	Площадь треугольника
Mt	Проверка существования треугольника