

# Основные понятия языка Pascal

# Pascal

Паскаль был разработан швейцарским ученым Никлаусом Виртом. Паскаль считается важнейшим инструментом для обучения методам структурного программирования и с 1983 г. введен в учебные курсы в школах для учащихся, которые специализируются в области информатики.

В дальнейшем язык Паскаль совершенствовался и приобрел новые свойства, отличные от авторского варианта.

Язык Паскаль относительно прост в изучении, довольно ясен и логичен и, будучи первым изучаемым языком программирования, приучает к хорошему стилю.

- *Как и естественные языки, каждый язык программирования имеет свой стиль и свои правила.*
- **Синтаксис** языка программирования – это набор правил, которые определяют способы построения правильных программ из символов алфавита.

# Алфавит языка Паскаль

**26 латинских строчных и 26 латинских прописных букв:**

**A B C D E F G H I J K L M N O P Q R S T U V W X Y Z**  
**a b c d e f g h i j k l m n o p q r s t u v w x y z**

**подчеркивание \_**

**10 цифр:**

**0 1 2 3 4 5 6 7 8 9**

**знаки операций:**

**+ - \* / = <> < > <= >= := @**

**ограничители (разделители):**

**. , ' ( ) [ ] ( . . ) { } ( \* \* ) .. : ;**

**спецификаторы:**

**^ # \$**

# Структура программы

*{1. заголовок программы}*

**program**      Имя\_Программы;

*{2. раздел указания используемых модулей}*

**uses**      Список\_Используемых\_Модулей;

*{3. Раздел описаний}*

**label**      Описания\_меток;

**const**      Описания\_Констант;

**type**      Описания\_Типов;

**var**      Описания\_Переменных;

**procedure**      Описания\_Процедур\_и\_функций;

**function**

**exports**      Описания\_Экспортируемых\_Имен;

*{4. Раздел операторов}*

**begin**

    Операторы

**end.**

# Упрощенная структура программы

*{1. заголовок программы}*

**program** Имя\_Программы;

*{2. раздел указания используемых модулей}*

**uses** Список\_Используемых\_Модулей;

*{3. Раздел описаний}*

**const** Описания\_Констант;

**var** Описания\_Переменных;

*{4. Раздел операторов}*

**begin**

Операторы программы

**end.**

- Команды языка программирования называются операторами
- Разделителем операторов в Паскале является `;` (точка с запятой)



***Комментарий*** представляет собой фрагмент текста программы, ограниченный символами **{}**.  
Комментарии в программе выполняют информационную функцию.

***{Моя первая программа }***

```
program first;  
begin  
    writeln('Hello, World!')  
end.
```

## Пример: периметр прямоугольника

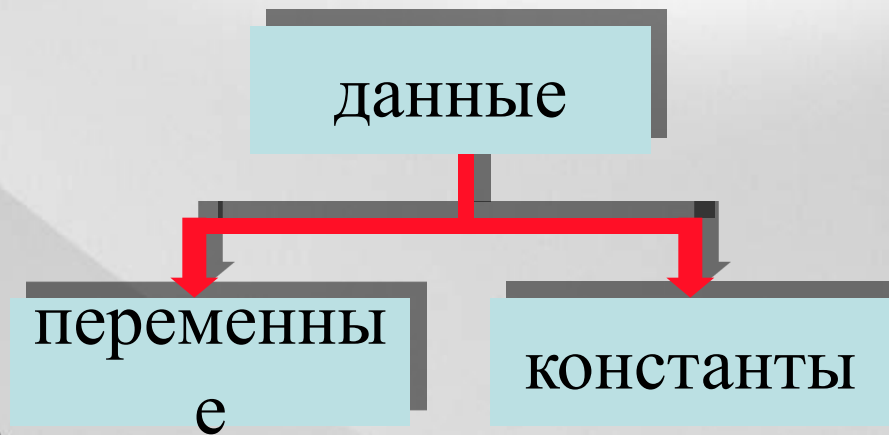
```
program perimetr;           {заголовок программы}
uses crt;                 {crt – необходим для очистки экрана}
var a,b:integer;         {объявление переменных}
    P:integer;
Begin                     {начало программы}
    clrscr;                {очистка экрана}
    a:=12;                 {присваиваем переменной a значение 12}
    b:=7;                  {присваиваем переменной b значение 7}
    P:=2*(a+b);           {значение выражения присваиваем P}
    write('P = ',P);      {выводим на экран значение P}
end.                     {конец программы}
```

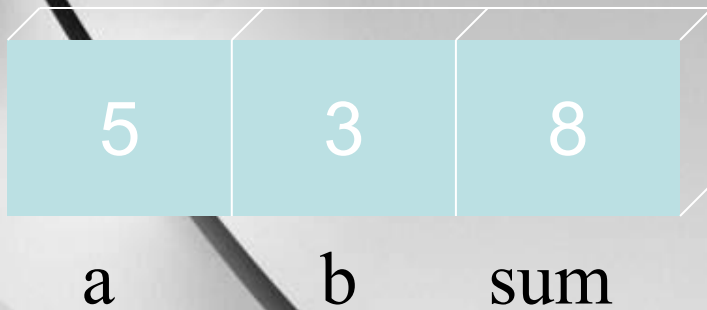
***Программа в своей работе имеет дело с данными.***

Некоторые данные устанавливаются еще до того, как программа начнет выполняться, а после ее запуска сохраняют свои значения неизменными на всем протяжении работы программы. Это ***константы***.

Другие данные могут изменяться во время выполнения программы. Они называются ***переменными***.

- ✓ Различие между **переменной** и **константой** довольно очевидно: во время выполнения программы значение переменной может быть изменено, а значение константы нет.





Под переменной мы будем понимать ячейку («коробку»), куда компьютер может записывать («складывать») данные.

Под ячейкой мы в действительности подразумеваем «кусочек памяти» в котором хранится информация.

Чтобы воспользоваться информацией, хранящейся в ячейке, нужно, чтобы каждая ячейка имела свое **имя** или, как часто говорят, - **идентификатор**.

# **Идентификаторы.**

- ✓ **Именами** (идентификаторами) называют элементы языка - константы, метки, типы, переменные, процедуры, функции, модули, объекты.
- ✓ **Идентификатором** является последовательность букв, цифр и знаков подчеркивания, которая начинается с буквы или символа подчеркивания и не содержит пробелов.

- ✓ Имя может содержать произвольное количество символов, но значащими являются 63 символа.
- ✓ Не разрешается в языке ПАСКАЛЬ использовать в качестве имен **служебные слова** и **стандартные имена**, которыми названы стандартные константы, типы, процедуры, функции и файлы.
- ✓ Примеры имен языка ПАСКАЛЬ:
  - A   b12   r1m   SIGMA   gamma   I80\_86

В Паскале разница между строчными и прописными буквами игнорируется, поэтому имена **NaMe** и **name** одинаковы.



- **Служебное слово** – это слово, которое в языке ПАСКАЛЬ имеет определенное смысловое значение, которое не может быть изменено. Иногда его называют **ключевым словом**.

## **Служебные (зарезервированные) слова:**

|                    |                       |                  |                |
|--------------------|-----------------------|------------------|----------------|
| <b>ABSOLUTE</b>    | <b>EXPORTS</b>        | <b>LIBRARY</b>   | <b>SET</b>     |
| <b>ASSEMBLER</b>   | <b>EXTERNAL</b>       | <b>MOD</b>       | <b>SHL</b>     |
| <b>AND</b>         | <b>FAR</b>            | <b>NAME</b>      | <b>SHR</b>     |
| <b>ARRAY</b>       | <b>FILE</b>           | <b>NIL</b>       | <b>STRING</b>  |
| <b>ASM</b>         | <b>FOR</b>            | <b>NEAR</b>      | <b>THEN</b>    |
| <b>ASSEMBLER</b>   | <b>FORWARD</b>        | <b>NOT</b>       | <b>TO</b>      |
| <b>BEGIN</b>       | <b>FUNCTION</b>       | <b>OBJECT</b>    | <b>TYPE</b>    |
| <b>CASE</b>        | <b>GOTO</b>           | <b>OF</b>        | <b>UNIT</b>    |
| <b>CONST</b>       | <b>IF</b>             | <b>OR</b>        | <b>UNTIL</b>   |
| <b>CONSTRUCTOR</b> | <b>IMPLEMENTATION</b> | <b>PACKED</b>    | <b>USES</b>    |
| <b>DESTRUCTOR</b>  | <b>IN</b>             | <b>PRIVATE</b>   | <b>VAR</b>     |
| <b>DIV</b>         | <b>INDEX</b>          | <b>PROCEDURE</b> | <b>VIRTUAL</b> |
| <b>DO</b>          | <b>INHERITED</b>      | <b>PROGRAM</b>   | <b>WHILE</b>   |
| <b>DOWNTO</b>      | <b>INLINE</b>         | <b>PUBLIC</b>    | <b>WITH</b>    |
| <b>ELSE</b>        | <b>INTERFACE</b>      | <b>RECORD</b>    | <b>XOR</b>     |
| <b>END</b>         | <b>INTERRUPT</b>      | <b>REPEAT</b>    |                |
| <b>EXPORT</b>      | <b>LABEL</b>          | <b>RESIDENT</b>  |                |

***Пробелы*** нельзя использовать внутри  
сдвоенных символов и зарезервированных  
слов.

***Существуют имена которые  
называются стандартными.***

sin cos real true

В отличие от служебных слов смысл  
стандартных имен ***может быть  
переопределен*** программистом.

# Объявление переменных

- ✓ **Это указание компилятору, сколько памяти необходимо зарезервировать для переменных нашей программы.**
- ✓ **В откомпилированной программе для всех переменных отведено место в памяти, и всем переменным присвоены нулевые значения.**
- *Все переменные, используемые в программе необходимо объявить в разделе описания переменных после зарезервированного слова **var**.*

# *Типы данных*

- Имена *стандартных* типов являются predetermined идентификаторами и действуют в любой точке программы. Они описаны в стандартном модуле System, который по умолчанию подключается в список используемых модулей.
- *Пользовательские* типы – это дополнительные типы (простые и структурированные) описанные пользователем.