

# Среда программирования Pascal ABC



СТРУКТУРА ПРОГРАММЫ

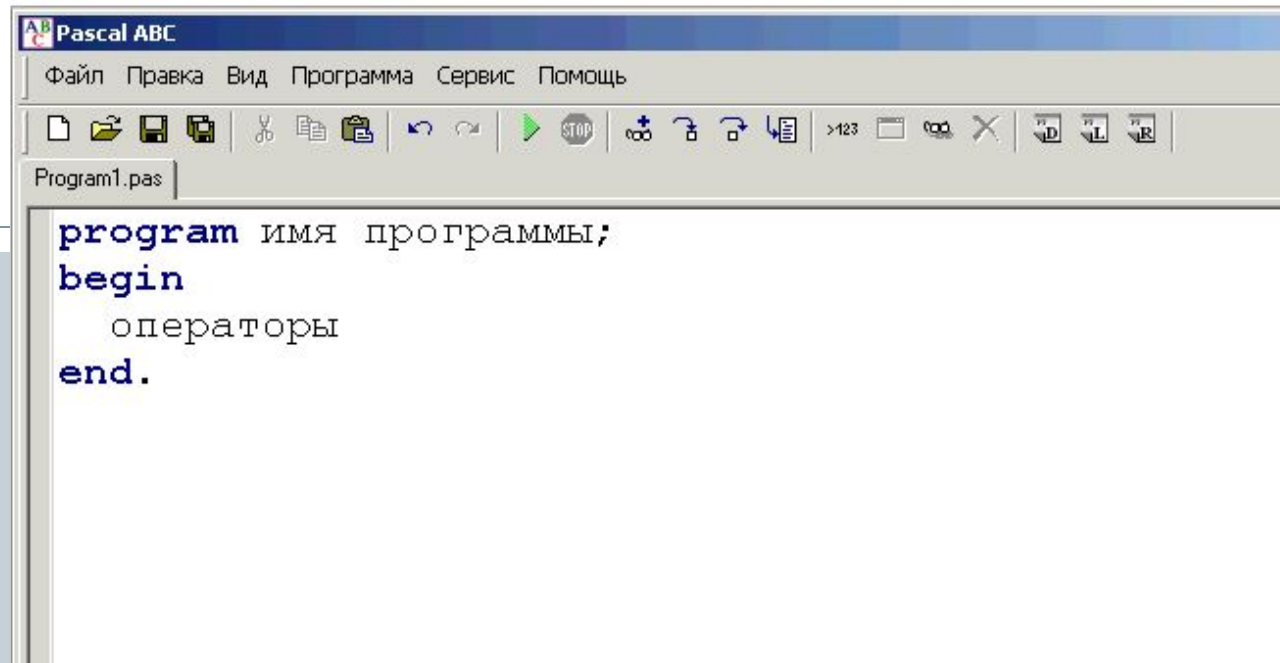
# Структура программы



- Программа на языке **Pascal ABC** имеет следующий вид:
- **program** имя программы;  
раздел подключения модулей  
раздел описаний  
**begin**  
операторы  
**end.**
- Первая строка называется *заголовком программы* и не является обязательной.
- Раздел подключения модулей начинается со служебного слова **uses**, за которым следует список имен модулей, перечисляемых через запятую.
- Раздел описаний может включать разделы описания переменных, констант, типов, процедур и функций, которые следуют друг за другом в произвольном порядке.
- Раздел подключения модулей и раздел описаний могут отсутствовать.
- Операторы отделяются один от другого символом "точка с запятой".



- **program** имя программы;
- uses** раздел подключения модулей
- var** раздел описаний
- begin**  
операторы
- end.**

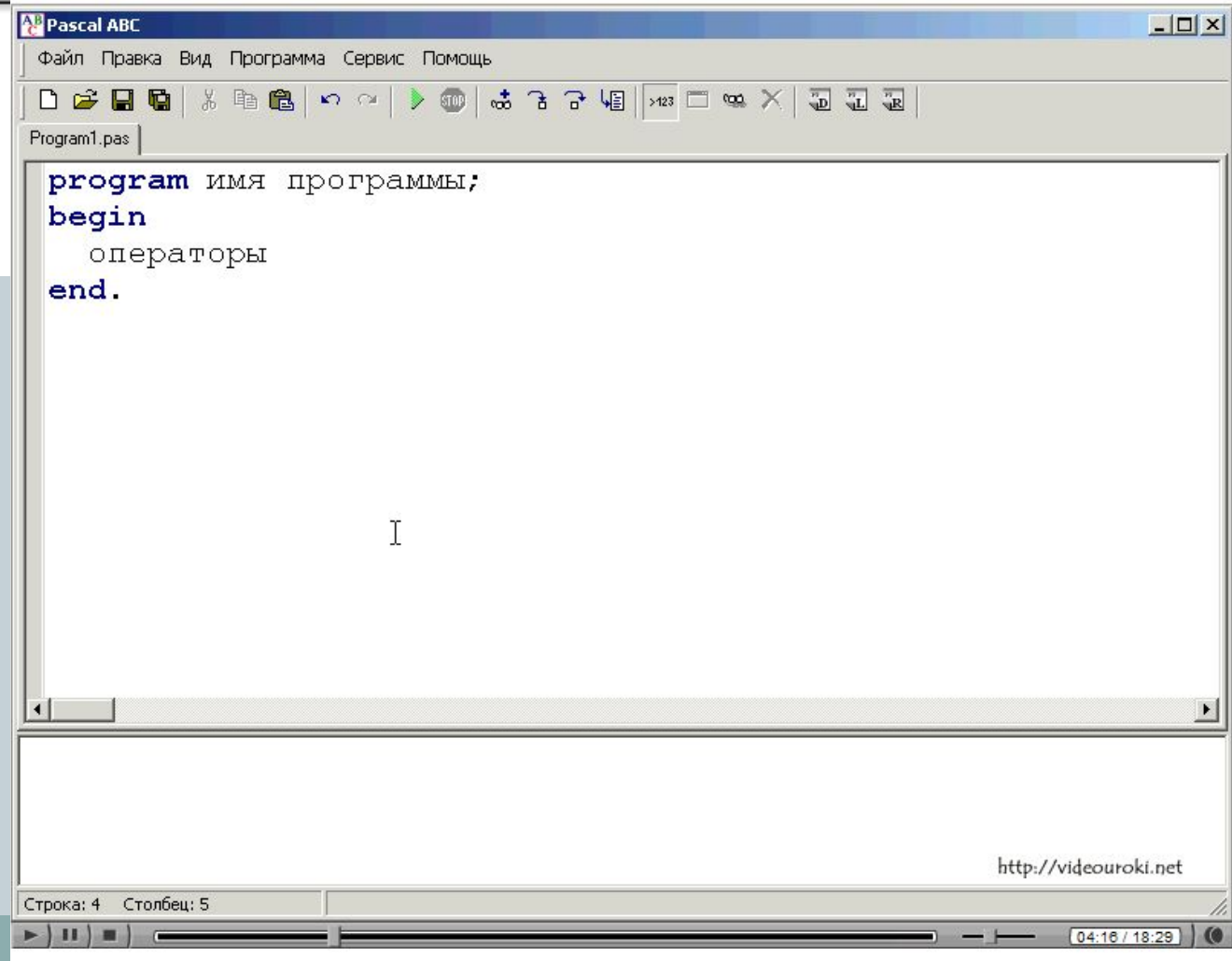
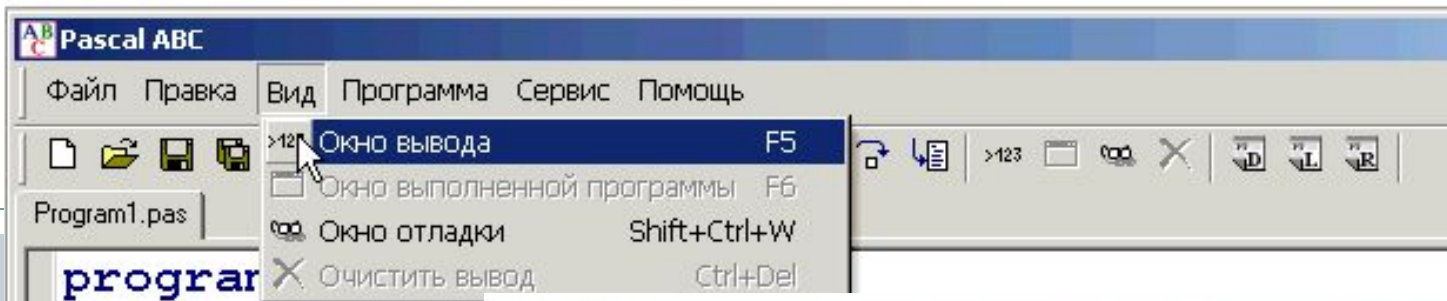


The screenshot shows a window titled "Pascal ABC" with a menu bar (Файл, Правка, Вид, Программа, Сервис, Помощь) and a toolbar. The main text area contains the following code:

```
program имя программы;  
begin  
    операторы  
end.
```

та;

- **Var** – описание;
- **Begin** – Начало;
- **End** – Конец.



**Pascal ABC**

Файл Правка Вид Программа Сервис Помощь

Program1.pas

```
program
begin
  операторы
end.
```

- >123 Окно вывода F5
- Окно выполненной программы F6
- Окно отладки Shift+Ctrl+W**
- Очистить вывод Ctrl+Del

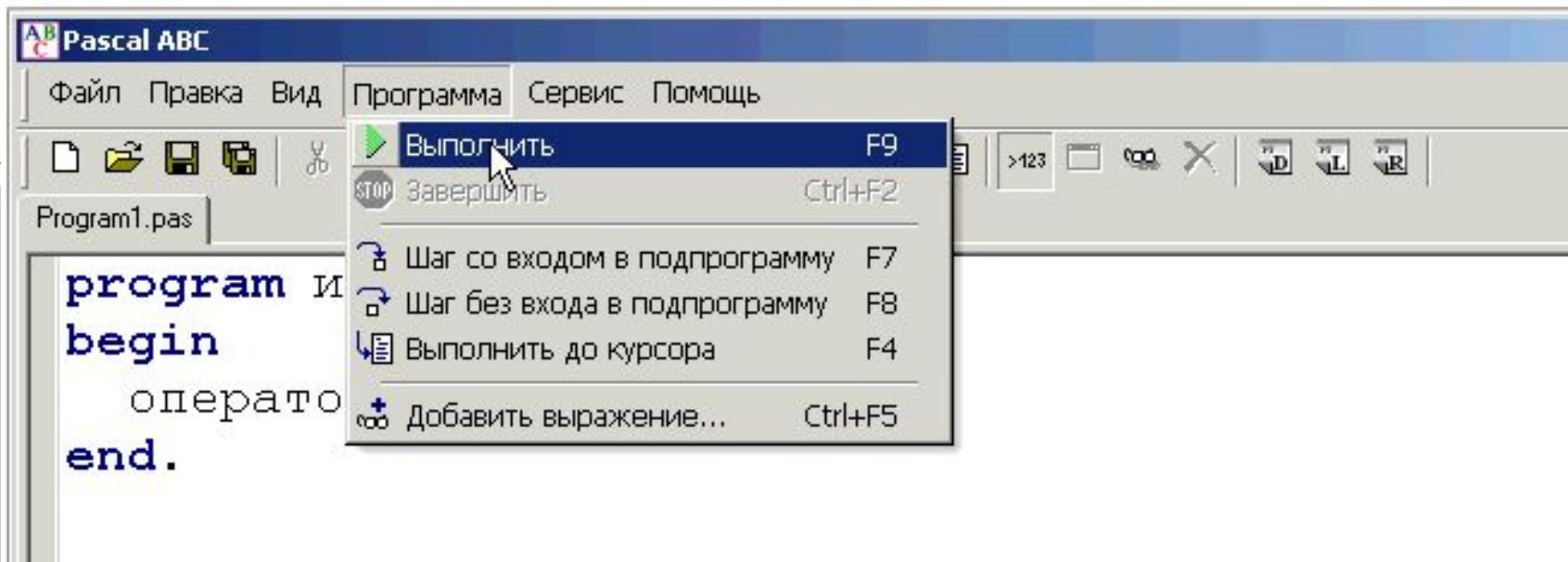
Окно отладки

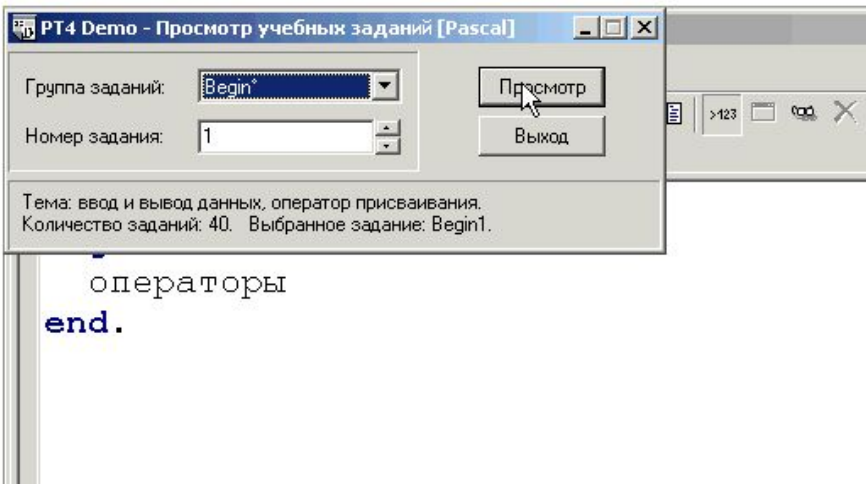
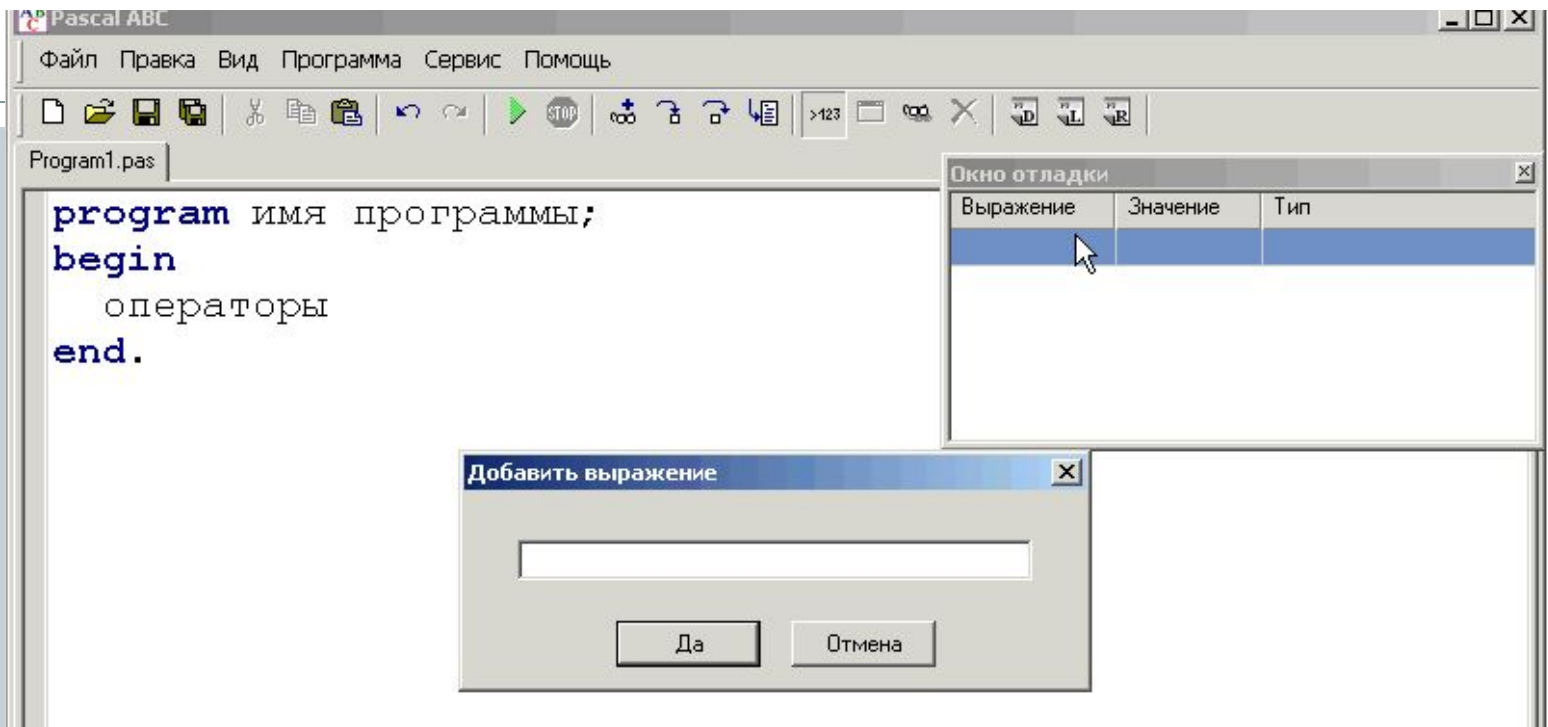
| Выражение | Значение | Тип |
|-----------|----------|-----|
|           |          |     |

Строка: 4 Столбец: 5

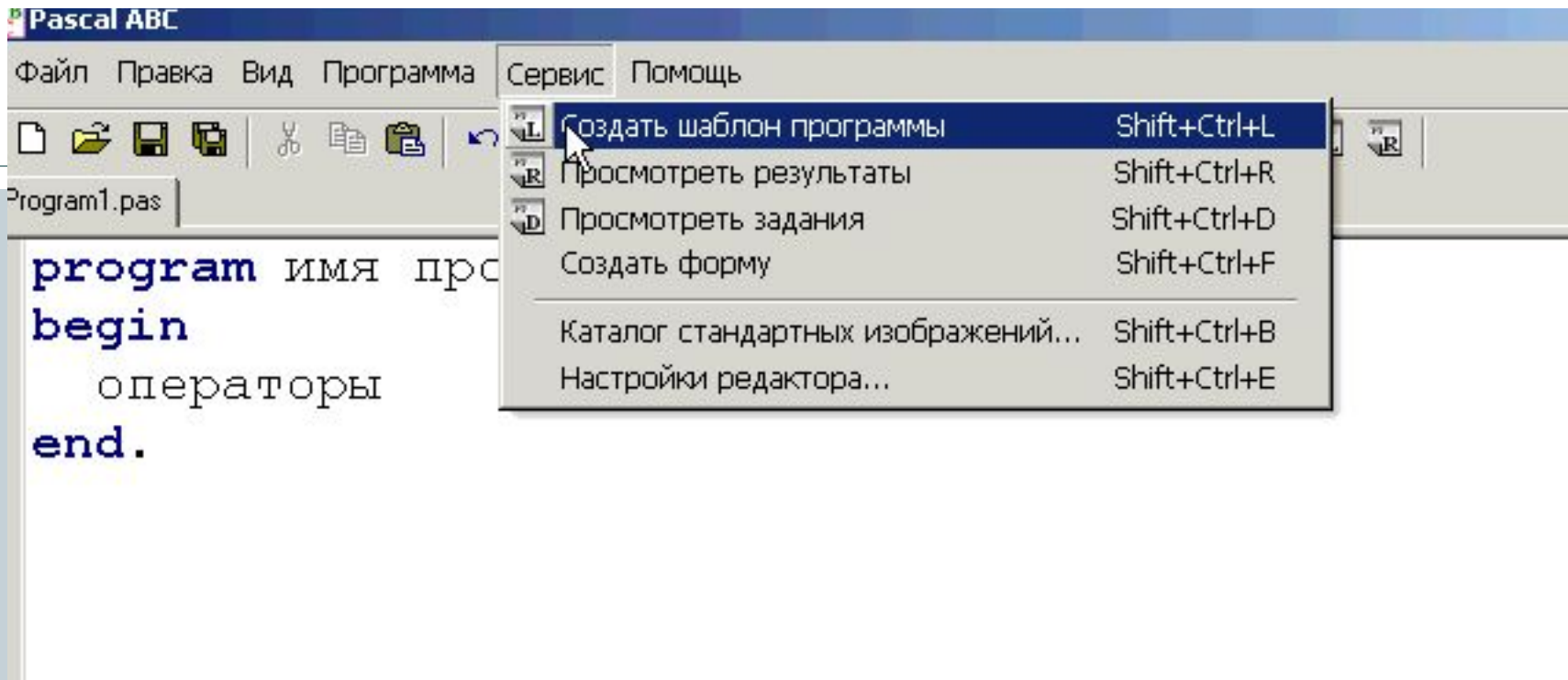
<http://videouroki.net>

04:44 / 18:29









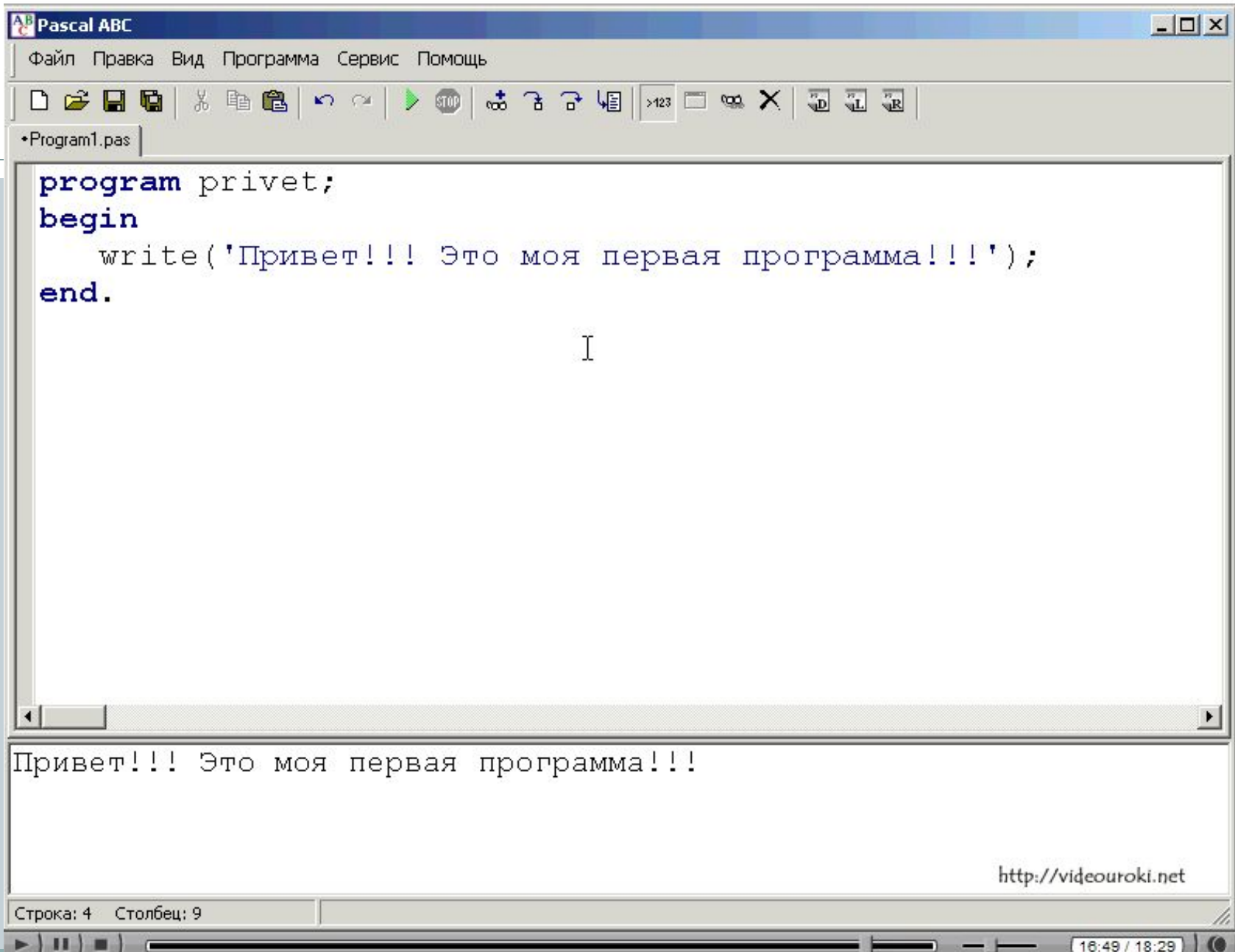
ABC Pascal ABC

Файл Правка Вид Программа Сервис Пом

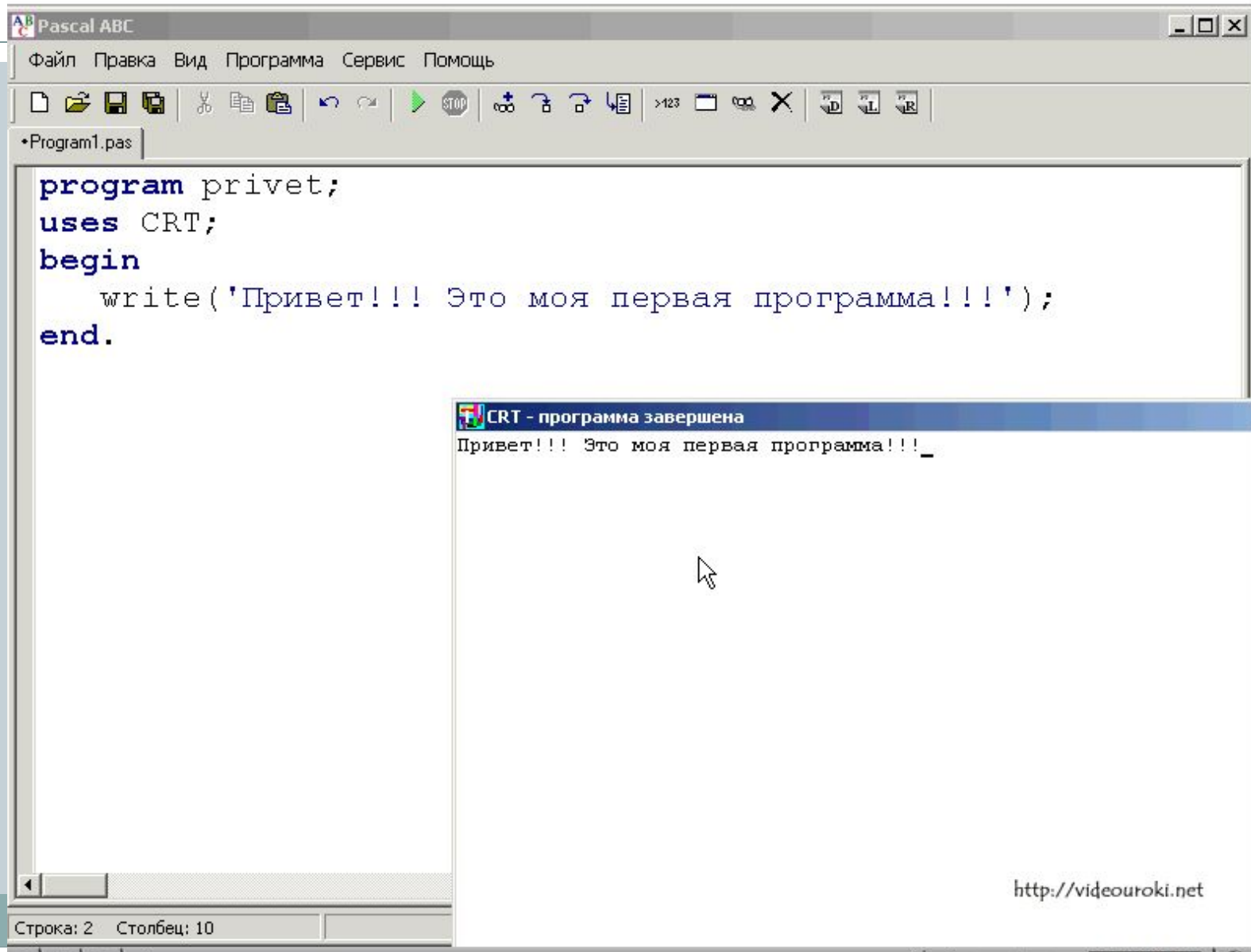


Program1.pas

```
program      (программа)
begin        (начало)
end          (конец)
```



## Команда uses – использование, откроется в отдельном окне





Напишем нашу первую программу:

Дадим нашей программе имя, должно быть написано латинскими буквами, и не должно начинаться с цифры.

Каждый оператор заканчивается - ;

Write - это команда вывода в окно просмотра.

# Задача 1.



- Выведем на экран приветствие: "Добрый день".
- Program prim1;

*{необязательный элемент программы  
Имя этой программы prim1 (заметим, что в имени программы не должно быть пробелов, оно должно начинаться с буквы, состоять только из латинских букв, цифр и некоторых символов, не допускается использование символов точки и запятой). Описательной части нет, а сразу идет раздел операторов, начинающийся со служебного слова begin в Turbo Pascal 7.0, после которого идет оператор языка}*

```
begin    {Выводим текст}  
    writeln('Добрый день');
```

*{В конце программы в Turbo Pascal 7.0 обязательно ставится оператор end.}*

```
end.
```



```
Program prim1;  
begin  
    writeln('Добрый день');  
  
end.
```

## Задача 2.

### Ввод с клавиатуры значения переменной N

```
program Inp;
  uses Crt;
  var
    N: integer;
begin
  ClrScr;
  write('Введите число с клавиатуры:');
  readln(N);  { Здесь программа приостановится и будет ожидать
ввода с клавиатуры. Наберите на клавиатуре число, например 153, и
нажмите клавишу Enter}
  writeln('Вы ввели число ', N);
  readln  { Это оператор пустого ввода. Здесь программа опять
приостановится и будет ожидать нажатия клавиши Enter. За это
время вы успеете просмотреть вывод на экране.}
end.
```





```
program Inp;  
  uses Crt;  
  var  
    N: integer;  
begin  
  ClrScr;  
  write('Введите число с клавиатуры:');  
  readln(N);  
  writeln('Вы ввели число ', N);  
  readln  
end.
```

# Расчет скорости тела при падении с башни



- Program Piza;  
const { *Это раздел описания констант. Он находится перед разделом var* }  
    G=9.8; { *Тип константы определяется автоматически, по форме записи числа. В данном случае из-за наличия десятичной точки это тип real* }  
var  
    V,H: real;  
begin  
    write('Введите высоту башни:');  
    readln(H);  
    V:=Sqrt(2\*G\*H);  
    writeln('Скорость падения',V:6:3):  
    { *Чтобы текст и число не "слиплись", после текста внутри апострофов добавлен пробел* }  
    readln  
end.

- Program Piza;  
const crt;  
G=9.8;  
var  
V,H,N:real;  
begin  
clrscr;  
write('Введите высоту башни:');  
readln(H);  
V:=Sqrt(2\*G\*H);  
writeln('Скорость падения',V:6:3);  
readln  
end.

```
uses crt;  
    const  
        G=9.8;  
    var  
        V,H, N:real;  
begin  
clrscr;  
    write('Введите высоту башни');  
    readln(H);  
    V:=Sqrt(2*G*H);  
    writeln('Скорость падения',V:6:3);  
    readln;  
end.
```

- crt, clrscr; - очистка экрана



# Система Pascal ABC

- Система **Pascal ABC** предназначена для обучения программированию на языке Паскаль и ориентирована на школьников и студентов младших курсов.
- По мнению авторов первоначальное обучение программированию должно проходить в достаточно простых и дружественных средах, в то же время эти среды должны быть близки к стандартным по возможностям языка программирования и иметь достаточно богатые и современные библиотеки стандартных подпрограмм.
- Язык Паскаль признан многими российскими преподавателями как один из лучших именно для начального обучения. Однако, среда Borland Pascal, ориентированная на MS DOS, устарела, а среда Borland Delphi с ее богатыми возможностями сложна для начинающего программиста. Так, попытка начинать обучение с написания событийной программы в Borland Delphi вызывает у обучаемого массу сложностей и приводит к ряду неправильно сформированных навыков.
- Система **Pascal ABC** основана на языке Delphi Pascal и призвана осуществить постепенный переход от простейших программ к модульному, объектно-ориентированному, событийному и компонентному программированию. Некоторые языковые конструкции в **Pascal ABC** допускают, наряду с основным, упрощенное использование, что позволяет использовать их на ранних этапах обучения. Например, в модулях может отсутствовать разделение на секцию интерфейса и секцию реализации. В этом случае модули устроены практически так же, как и основная программа, что позволяет приступить к их изучению параллельно с темой "Процедуры и функции". Тела методов можно определять непосредственно внутри классов (в стиле Java и C#), что позволяет создавать классы практически сразу после изучения записей, процедур и функций.
- Ряд модулей системы программирования **Pascal ABC** специально создавался для учебных целей:
- Модуль растровой графики GraphABC обходится без объектов, хотя его возможности практически совпадают с графическими возможностями Borland Delphi. Он доступен в несобытийных программах и позволяет легко создавать анимацию без мерцания.
- Модуль Events позволяет создавать простейшие событийные программы без использования объектов (события представляют собой обычные процедурные переменные).
- Модули Timers и Sounds позволяют создавать таймеры и звуки, которые также реализованы в процедурном стиле. Эти модули можно использовать даже в консольных программах.
- Модуль контейнерных классов Containers позволяет работать с основными структурами данных (динамические массивы, стеки, очереди, множества), реализованными в виде классов.
- Модуль векторной графики ABCObjects предназначен для быстрого изучения основ объектно-ориентированного программирования, а также позволяет создавать достаточно сложные игровые и обучающие программы.
- Модуль визуальных компонентов VCL позволяет создавать событийные приложения с главной формой в стиле Delphi. Классы VCL немного упрощены по сравнению с аналогичными классами Delphi. Имеется редактор форм и инспектор объектов. Технология восстановления формы по коду программы позволяет обойтись для приложения с главной формой одним файлом (!).
- В языке **Pascal ABC** имеются арифметические операции с типизированными указателями (в стиле языка C), а также тип complex, предназначенный для работы с комплексными числами.
- Компилятор **Pascal ABC** является компилятором переднего плана (front-end). Это означает, что он не генерирует исполняемый код в виде .exe-файла, а создает в результате компиляции дерево программы в памяти, которое затем выполняется с помощью встроенного интерпретатора. В итоге скорость работы программы примерно в 20 раз медленнее скорости работы этой же программы, откомпилированной в среде Borland Pascal, и в 50 раз медленнее этой программы, откомпилированной в среде Borland Delphi.
- В системе **Pascal ABC** ученик может выполнять так называемые **проверяемые задания**, обеспечивающие постановку задачи со случайными исходными данными, контроль операций ввода-вывода, проверку правильности решения, а также ведение протокола решения задач.
- Проверяемые задания реализованы в виде электронного задачника по программированию **Programming Taskbook**, содержащего 1000 задач по программированию разного уровня сложности (от простейших задач до задач на файлы, указатели и рекурсию) а также в виде исполнителей Робот и Чертежник, предназначенных для быстрого обучения основам программирования школьников младших и средних классов.
- В свободно распространяемую версию **Pascal ABC & Programming Taskbook Mini Edition** входит мини-версия электронного задачника (200 задач) и урезанный комплект задач для исполнителей Робот и Чертежник. Версия **Pascal ABC & Programming Taskbook Complete Edition** содержит полный комплект задач.