

# Турбо Паскаль

**Запуск интегрированной  
среды программирования**

# Задание

Запустите среду программирования и рассмотрите экран. Перед вами полоса меню, область окна и строка статуса. Нажмите клавишу F10 - теперь вам доступны все опции меню. Используя клавиши управления курсором, рассмотрите меню. С командами меню мы будем знакомиться постепенно. Нажмите клавишу Esc (вы вышли из меню). Перемещая курсор в окне, следите за строкой статуса. Какая информация отражается в этой строке?

Почти все, что вы видите и делаете в среде Турбо Паскаль, происходит в окнах.

# Окно

- **Окно** - это область экрана, которую можно перемещать, изменять в размере, перекрывать, закрывать и открывать.
- Интегрированная среда программирования Турбо Паскаль позволяет иметь любое количество открытых окон, но в любой момент времени активным может быть только одно.
- **Активное окно** – это окно, с которым вы в настоящий момент работаете.

# Общие горячие клавиши:

- **F1** - выводит окно подсказки;
- F2** - сохраняет файл активного окна;
- F3** - появление диалогового окна и возможность открыть файл;
- F4** - запускает программу до строки, на которой стоит курсор;
- F5** - масштабирует диалоговое окно;
- F6** - переходит к следующему открытому окну;
- F7** - запускает программу в режиме отладки с заходом внутрь процедур;
- F8** - запускает программу в режиме отладки, минуя вызов процедур;
- F9** - компилирование программы в текущем окне;
- F10** - возвращение в меню.

# Как войти в меню?

Всего есть три возможности:

- с помощью "мышки";
- с помощью клавиши F10;
- с помощью комбинации Alt+<выделенная буква>. О том, что мы в меню свидетельствует **курсор** - прямоугольник зеленого цвета.

С помощью клавиш управления курсором подсветите слово FILE и нажмите клавишу "Enter". Что вы видите?

## Появилась вертикальная таблица со списком команд, называемая выпадающим меню.

- **Open-F3** - открыть существующий файл (при активизации этой опции появляется окно со списком файлов, где можно выбрать необходимый),
- **New** - создать новый файл (очищает память редактора и переводит в режим создания нового файла, которому присваивается имя Noname.pas; имя можно изменить при записи файла на диск),
- **Save-F2** - сохранить файл (переписывает файл из памяти редактора на диск),
- **Save as** - сохранить с новым именем,
- **Save all** - сохранить все в окнах (записывает содержимое всех окон редактора в соответствующие файлы),
- **Change dir** - смена каталога (позволяет изменить установленный по умолчанию диск или каталог),
- **Print** - печать файла,
- **Get info** - выдача информации о текущем состоянии программы и используемой памяти,
- **DOS Shell** - выход в DOS без выгрузки из памяти (для возврата ввести команду exit),
- **Exit** - выход и выгрузка из памяти.

Программы на языке Паскаль имеют блочную структуру:

1. Блок типа PROGRAM - имеет имя, состоящее только из латинских букв и цифр. Его присутствие не обязательно, но рекомендуется записывать для быстрого распознавания нужной программы среди других листингов.
2. Программный блок, состоящий в общем случае из 7 разделов:
  - раздел описания модулей (uses);
  - раздел описания меток (label);
  - раздел описания констант (const);
  - раздел описания типов данных (type);
  - раздел описания переменных (var);
  - раздел описания процедур и функций;
  - раздел описания операторов.

Общая структура программы на языке Паскаль  
следующая:

```
Program ИМЯ..; {заголовок программы}  
  Uses ...; {раздел описания модулей}  
  Var ..; {раздел объявления переменных}  
  ...  
  Begin {начало исполнительной части  
    программы}  
    ... {последовательность  
      операторов}  
  End. {конец программы}
```



- Начнем знакомство с Паскалем с программы, которая складывает два числа и выводит сумму на экран.
- Откройте файл, в который Вы запишите эту программу. Для этого нажмите клавишу F10, чтобы выйти в главное меню, затем клавишами перемещения курсора выберите опцию File, а в выпавшем меню команду New.
- **Примечание.** Обратите внимание на оформление текста программы.

## Program Summa2;

{Задача. Вычислить сумму двух чисел и вывести на экран.

Решение. Иванов Петр, 10 А класс.}

- Var  
number1, {переменная для хранения первого числа}  
number2, {переменная для хранения второго числа}  
result {переменная для хранения результата вычисления}  
: integer;  
Begin {признак начала программы}  
number1 := 3; {присваиваем переменной number1 значение 3}  
number2 := 4; {присваиваем переменной number2 значение 4}  
{складываем значения переменных number1 и number2 и  
результат присваиваем переменной result }  
result := number1 + number2;  
Write (number1, '+', number2, '=', result); {вывод примера на  
экран}  
End. {признак конца программы}

# Задание

1. Найдите в этой программе заголовок, раздел описания переменных, признак начала программы, признак конца программы, тело программы, комментарий.
2. Что обозначает строчка:  
`number1, number2, result : integer;`
3. Как вы понимаете запись:  
`number1 := 3;`
4. Чему равно значение переменной `result` после выполнения оператора:  
`result := number1 + number2;`
5. Переведите с английского языка слово `Write`. Как вы думаете, что должен делать оператор с таким названием?
6. Поменяем местами второй и третий операторы. Будет ли программа работать? Почему?
7. Какой недостаток Вы видите у этой программы? Как нужно изменить условие задачи, чтобы решать подобные задачи с любыми числами. Подумайте, что должно измениться в теле нашей программы, чтобы выполнить эту задачу.

## А теперь подведем итог вашим размышлениям.

- Имя этой программы Summa2. Заметим, что требования к имени выполняются: оно отражает содержание программы, а также не содержит недопустимых символов.
- Далее идет специально выделенный комментарий, в котором вы должны записать подробно условие задачи и указать, кто написал эту программу и когда.
- Из разделов описаний имеется лишь один - раздел переменных. Он начинается со служебного слова Var. Мы описали три переменные: number1, number2, result. Все они переменные целого типа. Поэтому мы перечислили их через запятую, поставили двоеточие и указали тип переменных. Подобные объявления разделяются между собой точкой с запятой.
- После описательной части идет раздел операторов, начинающийся со служебного слова Begin, после которого идут операторы языка.
- Недостатком этой программы является то, что значения переменных постоянны. А нам нужно научиться писать такие программы, которые решают поставленные задачи в общем виде, т. е. для любых значений переменных. Для этого мы научимся запрашивать значения у пользователя, анализировать их и выдавать соответствующий результат.