

# Записи

Для работы с группой различных по типу данных, в системе программирования Turbo Pascal применяют тип данных "запись".

Понятие записи мы рассмотрим на примере списка владельцев машин:

N	Владелец	Марка
1.	Сурин О. О.	ВАЗ - 2102
2.	Куличёв В. А.	BMW
3.	Яшин К. Е.	ОКА

**Каждая строка в этой ведомости состоит из**

**отдельных элементов - данных разного типа:**

**а) порядковый номер - целое число;**

**б) Фамилия И.О. - символьная строка;**

**в) марки машины - символьная строка**

**Эти данные можно объединить в одну группу и считать записью. Запись в целом и отдельные её элементы (поля) обозначаются именами.**

**Введём например следующие обозначения:**

*Vladel* - имя всей записи;

*N* - порядковый номер;

*NAME* - фамилия И.О.;

*Ident* – марка автомобиля.

Обращение к элементу (полю) записи в программе выполняется с помощью уточнённого (составного) имени.

Уточнённое имя содержит

**Имя записи** и **Имя элемента**

A stylized illustration of a stack of books in various colors (red, yellow, blue) with a pink and orange pen resting on top. The books are stacked in a slightly offset manner, creating a sense of depth. The pen is positioned diagonally across the top of the books.

и записывается в следующем виде:

**<имя записи>. <имя поля записи>**

# Декларация (объявление) записи

*Объявление записи можно производить в разделе объявления переменных VAR или с использованием раздела объявления типов пользователя TYPE.*

```
VAR <имя записи>: RECORD  
<имя поля 1>: тип;  
<имя поля 2>: тип;  
    ...  
<имя поля n>: тип  
END;
```

# Декларация (объявление) записи

***TYPE***

*<имя типа> = RECORD*

*<имя поля 1>: тип;*

*<имя поля 2>: тип;*

*...*

*<имя поля n>: тип*

***END;***

***VAR***

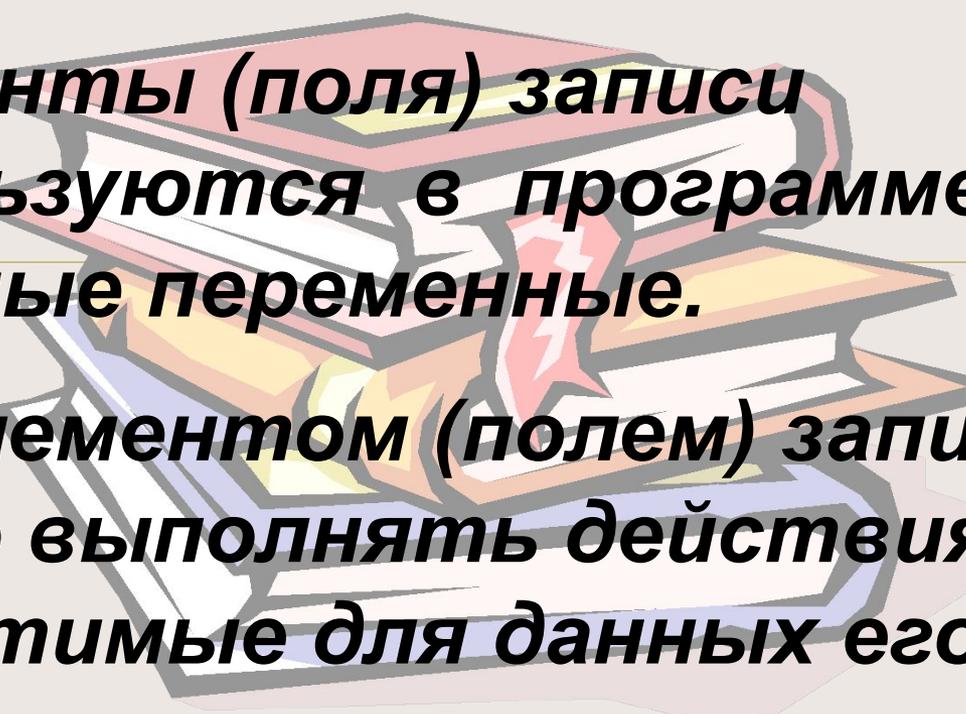
*<имя записи>: <имя типа>;*

# Примеры:

## Объявление записи

```
VAR VLADEL: record  
    N: Integer;  
    NAME: String[25];  
    IDENT: String[15];  
END;
```

```
TYPE VLADEL = record  
    N: Integer;  
    NAME: String[25];  
    RB: String[15];  
END;  
VAR  
    R1, R2: VLADEL;
```



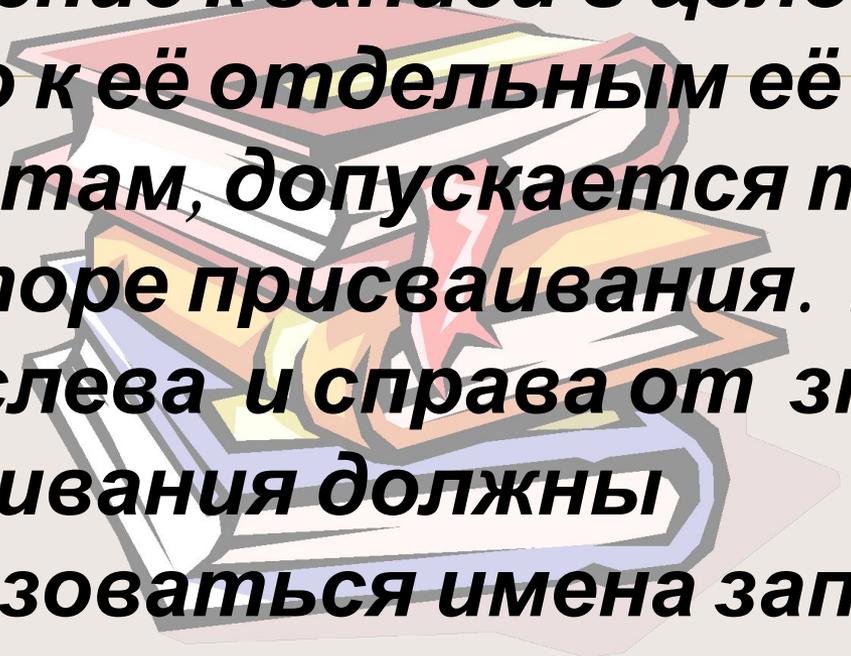
**Элементы (поля) записи  
используются в программе как  
обычные переменные.**

**Над элементом (полем) записи  
можно выполнять действия,  
допустимые для данных его  
типа.**

**Например:**

*R1.N := 2;*

*R2.Name := Сидоров П.П.;*

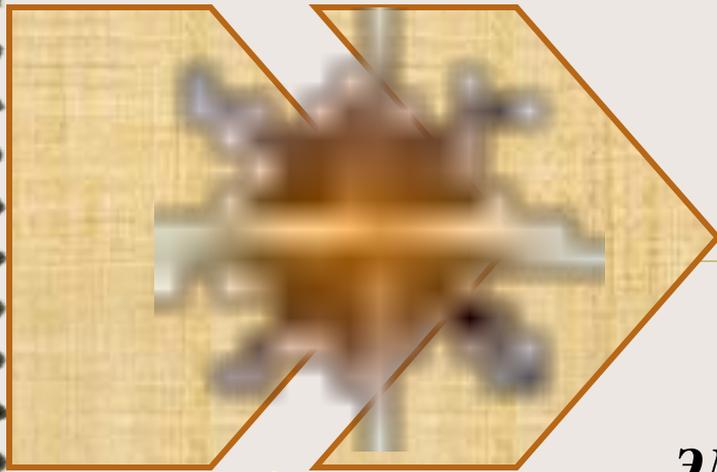


**Обращение к записи в целом, а не только к её отдельным её элементам, допускается только в операторе присваивания. При этом слева и справа от знака присваивания должны использоваться имена записей одинакового типа.**

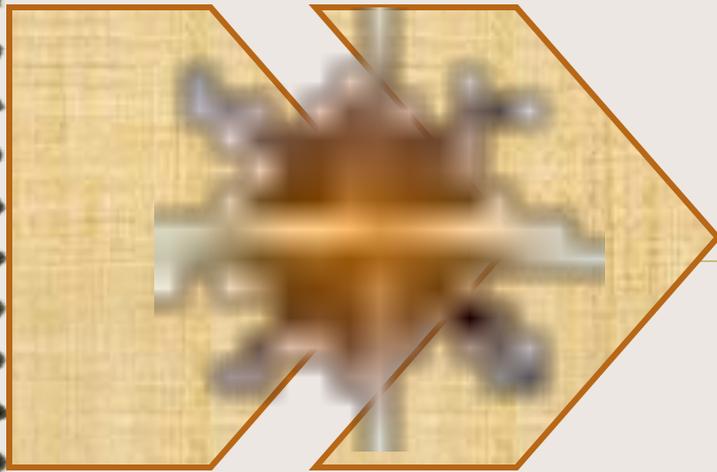
$R1 := R2;$



**Оператор  
присоединения**



**Обращение к  
элементам (полям) записи  
происходит с помощью уточнённого  
имени. Оператор присоединения  
позволяет упростить обращение к  
элементу (полю) записи. Имя записи  
выносится в заголовок оператора  
присоединения, а в блоке  
используются только имена  
элементов записи.**



# Общая форма

оператора  
присоединения

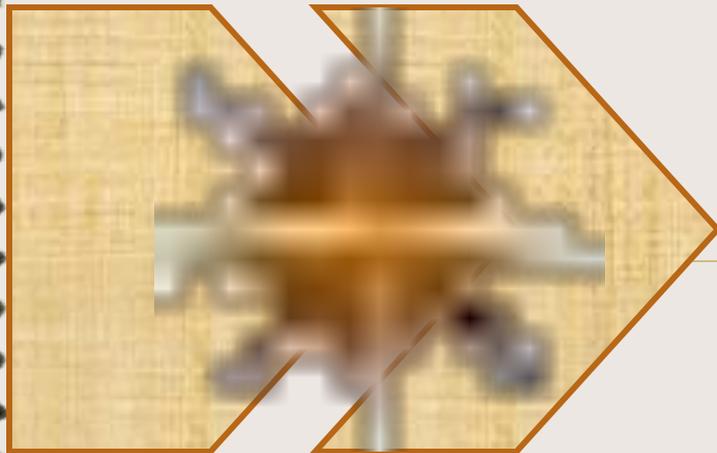
**WITH <имя записи> DO**

**Begin**

**{операторы, содержащие**

**ТОЛЬКО    имена элементов записи}**

**End;**



Например, для рассмотренной выше записи (списка учащихся) оператор присоединения может выглядеть так:

```
WITH R_EX1 DO
```

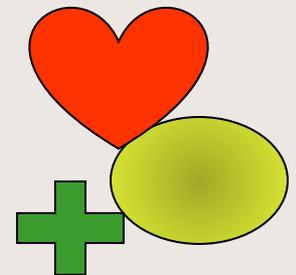
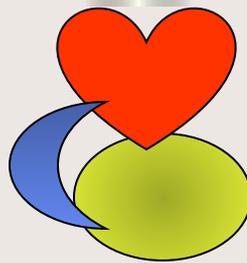
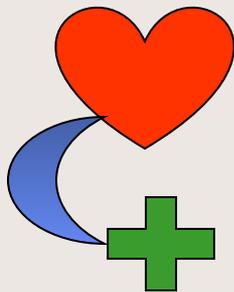
```
Begin
```

```
Write('Введите номер записи ');  
Readln(N);
```

```
Write('Введите Фамилию И.О. ');  
Readln(Name);
```

```
End;
```

# Записи с вариантами





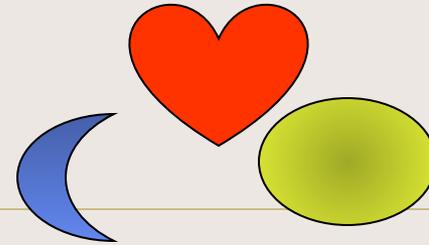
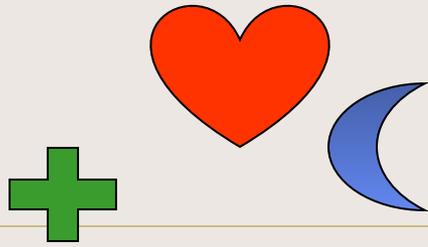
Особой разновидностью  
записей являются

"записи с вариантами",  
которые объявляются с

помощью  
зарезервированного

слова

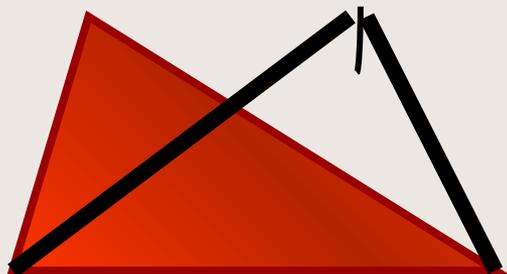
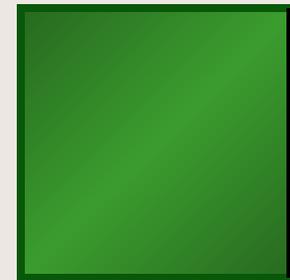
case



**С помощью записей с вариантами  
можно одновременно сохранять  
различные структуры данных,  
которые имеют  
*одну большую общую часть,*  
/  
**И**  
*небольшие части, различные в  
разных структурах.***

# Запись, хранящая параметры геометрических фигур.

Для квадрата задаётся одна сторона,



для треугольника – две стороны и угол между ними,

для окружности – её радиус.



Общей частью являются координаты текущей точки  $x, y$ .

**VAR**

**MS: Record**            {переменная типа запись}

**x: real;**    {текущая координата x}

**y: real;**    {текущая координата y}

**Case Fig: (Square, Triangle, Circle) of**  
{вариантная часть}

**Square: (side: real);**            {квадрат}

**Triangle: (s1, s2, angle: real);**    {треугольник}

**Circle: (Radius: real);**            {окружность}

**END;**

**VAR**

**MS: Record** {переменная типа запись}

**x: real;** {текущая координата x}

**y: real;** {текущая координата y}

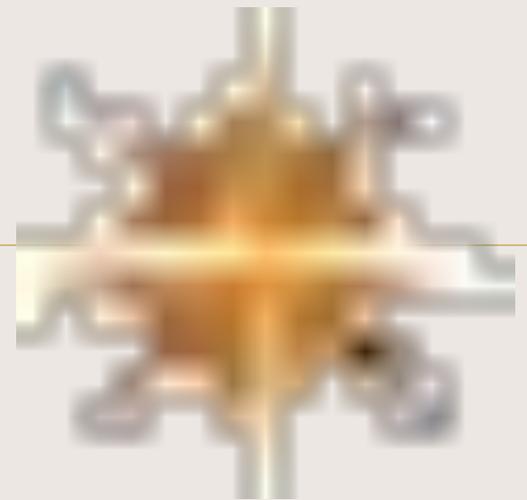
**Case Fig: (Square, Triangle, Circle) of**  
{вариантная часть}

**Square: (side: real);** {квадрат}

**Triangle: (s1, s2, angle: real);** {треугольник}

**Circle: (Radius: real);** {окружность}

**END;**



*Вариантная часть может быть только одна и должна располагаться в конце записи. Начинается она служебным словом case, за которым следует переменная выбора варианта.*



## VAR

MS: Record {переменная типа запись}

x: real; {текущая координата x}

y: real; {текущая координата y}

Case Fig: (Square, Triangle, Circle) of  
{вариантная часть}

Square: (side: real); {квадрат}

Triangle: (s1, s2, angle: real); {треугольник}

Circle: (Radius: real); {окружность}

END;

*За каждым возможным значением этой переменной через двоеточие в круглых скобках записываются поля данного варианта записи с указанием их типов. Круглые скобки необходимы, если даже для данного варианта записи поля отсутствуют.*