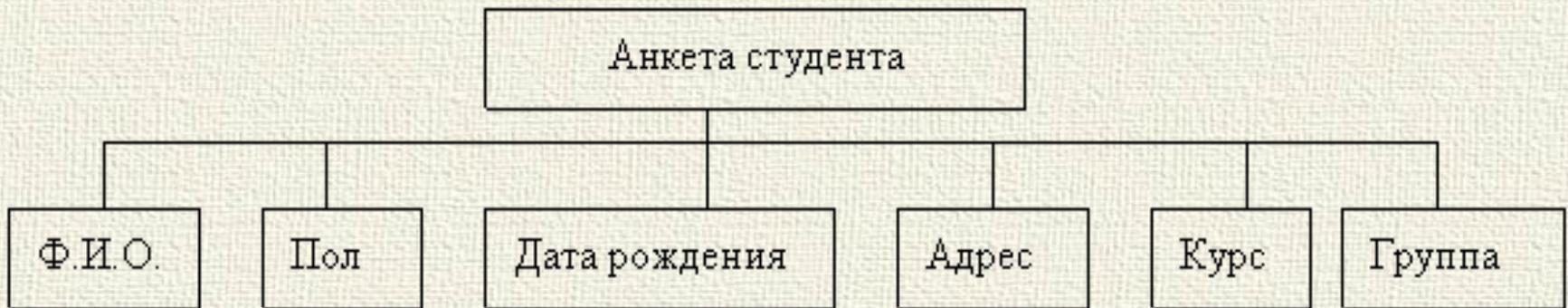


**Записи**

# Что это?

Запись – упорядоченная последовательность данных разного типа, объединённых одним именем.



Записи состоят из полей. Все операции совершаются с полями записи (с записью целиком действия не совершаются). Поле записи состоит из имени и типа поля. В качестве типа поля можно использовать любые простые и сложные типы (исключая файлы, и включая массивы и записи).

# Описание записей

Type

```
Имя_записи = Record
```

```
поле1: тип;
```

```
поле2: тип; ...
```

```
полеN: тип;
```

```
end;
```

Type

```
oly = record
```

```
strana: string[20];
```

```
medaly: integer;
```

```
god: 1900..2020;
```

```
leto_zima: boolean;
```

```
end;
```

Var

```
a: oly;
```

# Обращение к полям

К полю записи можно обратиться, используя составное имя

<имя переменной>.<имя поля>

a.strana; a.god - обращение к полю

Действия с полями.

```
a.letto_zima:=true;
```

```
writeln(a.strana);
```

```
readln(a.medaly);
```

# Оператор присоединения

Оператор присоединения позволяет упростить обращение к элементу записи. Имя записи выносится в заголовок оператора присоединения, а в блоке используются только имена элементов записи.

```
with <имя записи> do  
begin  
операторы, содержащие  
имена элементов записи  
end;
```

```
with a do  
begin  
strana:='Россия';  
medaly:=295;  
god:=2014;  
leto_zima:=false;  
end;
```

# Массив записей

Чаще всего используются не единичные записи, а массив записей, представляющий собой базу данных.

Type

```
oly = record  
  strana: string[20];  
  medaly: integer;  
  god: 1900..2020;  
  leto_zima: boolean;
```

end;

Var

```
a: array [1..10] of oly;
```

Обращение к полям происходит также, как и для обычной записи, с учетом того, что каждая запись является элементом массива:

Действия с полями.

```
a[1].leto_zima:=true;  
writeln(a[2].strana);  
readln(a[1].medaly);
```

**Файлы**

# Что это?

Файл – поименованная область на носителе информации, содержащая данные определенного типа. Предназначены для долговременного хранения данных.

Типы файлов в Pascal:

- 1) Текстовые – хранят данные любого вида в текстовом формате.
- 2) Типизированные – хранят последовательность данных определенного типа. Доступ к данным осуществляется по их порядковым номерам.
- 3) Нетипизированные – хранят данные любого типа.

# Описание файлов

Взаимодействие Pascal с файлами происходит при помощи файловых переменных (Ф.п.).

var

f1: text; {текстовый файл}

f2: file of char; {типизированный файл}

f3: file; {нетипизированный файл}

...

f1, f2, f3 – файловые переменные.

Обращение к файлам внутри программы происходит только через файловые переменные.

И важно правильно связать файловую переменную с конкретным файлом.

# Процедуры и функции

П **Assign(Ф.п., 'Имя файла')** – связывание файла и файловой переменной.

П **Reset(Ф.п.)** – открытие файла для чтения. Если файл не существует выдается сообщение об ошибке.

П **Rewrite(Ф.п.)** – открытие файла для перезаписи. При открытии все данные из файла удаляются, если файл не существует, он создается автоматически.

П **Append(Ф.п.)** – открытие файла для дозаписи (только для текстовых файлов). При открытии данные из файла не удаляются, если файл не существует выдается сообщение об ошибке.

П **Close(Ф.п.)** – закрытие файла.

# Процедуры и функции

**П Rename(Ф.п., 'Новое имя')** – переименовывает файл. Файл должен быть закрыт.

**П Erase(Ф.п.)** – удаляет файл. Файл должен быть закрыт.

**Ф EOF(Ф.п.)** – возвращает значение «истина», если достигнут конец файла.

**Ф EOLN(Ф.п.)** – возвращает значение «истина», если достигнут конец строки. Только для текстовых файлов.

While not EOF(f) do

# Текстовые файлы

Запись в файл

П **Write(Ф.п., список данных)** – запись данных в одной строке.

П **Writeln(Ф.п., список данных)** – запись данных построчно.

Например: `writeln(f, 'Кривенс!!!');`

Считывание из файла

П **Read(Ф.п., список переменных)** – считывает данные одной строки файла в указанные переменные.

П **Readln(Ф.п., список переменных)** – считывает данные из файла построчно в указанные переменные.

Например: `readln(f, s); writeln(s);`

# Типизированные файлы

**П Write(Ф.п., список данных)** — записывает компоненты в файл.

**П Read(Ф.п., список переменных)** — считывает компоненты из файла в указанные переменные.

**П Seek(Ф.п., номер)** — перемещает указатель в файле к элементу, с указанным номером.

**Ф FileSize(Ф.п.)** — возвращает количество компонент файла.

**Ф FilePos(Ф.п.)** — возвращает порядковый номер текущего компонента файла.