

Урок 94. Классы и объекты (обзор)

Класс представляет собой *составной (структурированный)* тип, состоящий из **полей** (переменных), **методов** (процедур и функций) и **свойств**.

Переменные типа класс называются **объектами** или **экземплярами** класса. В языке PascalABC.NET классы являются *ссылочными* типами. Это значит, что переменная типа класс хранит в действительности ссылку на объект.

1. Описание классов

Описание класса имеет вид:

```
type
    имя_класса = class
        секция1
        секция2
        ...
    end;
```

Каждая секция имеет вид:

```
модификатор доступа
    описания полей
    объявления или описания методов и описания свойств
```

Модификатор доступа в первой секции может отсутствовать, при этом подразумевается модификатор `internal`. Все описания и объявления внутри класса образуют **тело класса**.

2. Модификаторы доступа

<code>public</code> (открытый)	можно обратиться из любого места программы
<code>private</code> (закрытый)	доступны только внутри методов этого класса
<code>protected</code> (защищённый)	доступны внутри методов этого класса и всех его подклассов
<code>internal</code> (внутренний)	доступны внутри сборки (термин .NET, сборка — это множество файлов, необходимых для генерации .exe или .dll-файла)

3. Описание методов. Конструкторы

Методы могут описываться как внутри, так и вне класса. При описании метода внутри класса его имя предваряется именем класса с последующей точкой.

Объекты создаются с помощью специальных методов, называемых **конструкторами**.

Конструктор представляет собой функцию, создающую объект в динамической памяти, инициализирующую его поля и возвращающую ссылку на созданный объект. При описании конструктора вместо ключевого слова `function` используется ключевое слово `constructor`. Кроме того, для конструктора не указывается тип возвращаемого значения.

В PascalABC.NET конструктор всегда должен иметь имя `Create`. При описании конструктора внутри класса можно опускать его имя.

В силу особенностей реализации вызовов конструкторов в .NET в PascalABC.NET всегда создается конструктор без параметров (независимо от того, определен ли другой конструктор). Этот конструктор инициализирует все поля нулевыми значениями (строковые поля — пустыми строками, логические — значением `False`).

4. Пример описания класса

```
type
  Person = class
  private
    fName: string;
    fAge: integer;
  public
    constructor Create(Name: string; Age: integer);
  begin
    fName := Name;
    fAge := Age;
  end;
  procedure Print;
  property Name: string read fName;
  property Age: integer read fAge;
end;

procedure Person.Print;
begin
  Writeln('$Имя: {Name} Возраст: {Age}');
end;
```