

Вспомним классификацию ПО по способу использования

---

# ОСНОВЫ ООП

# Цели обучения

---

- Управлять файлами проекта объектно-ориентированного программирования
- Использовать объектно-ориентированный язык программирования для решения простых задач

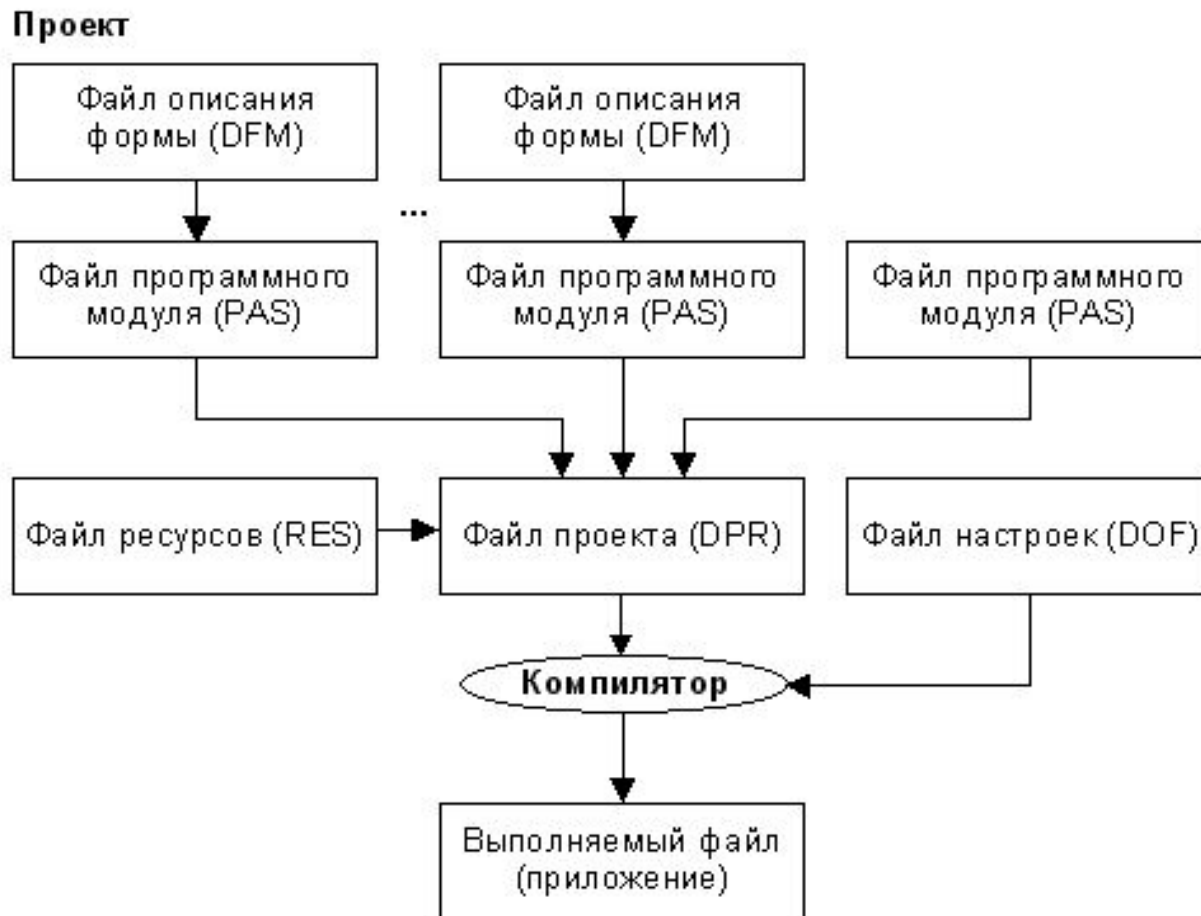
# Управлять файлами проекта объектно-ориентированного программирования

- создавать папку для сохранения файлов проекта
- сохранять файлы проекта, созданные в среде ООП
- управлять файлами проекта, запускает проект из внешнего носителя и копирует его на диск

# Использовать объектно-ориентированный язык программирования для решения простых задач

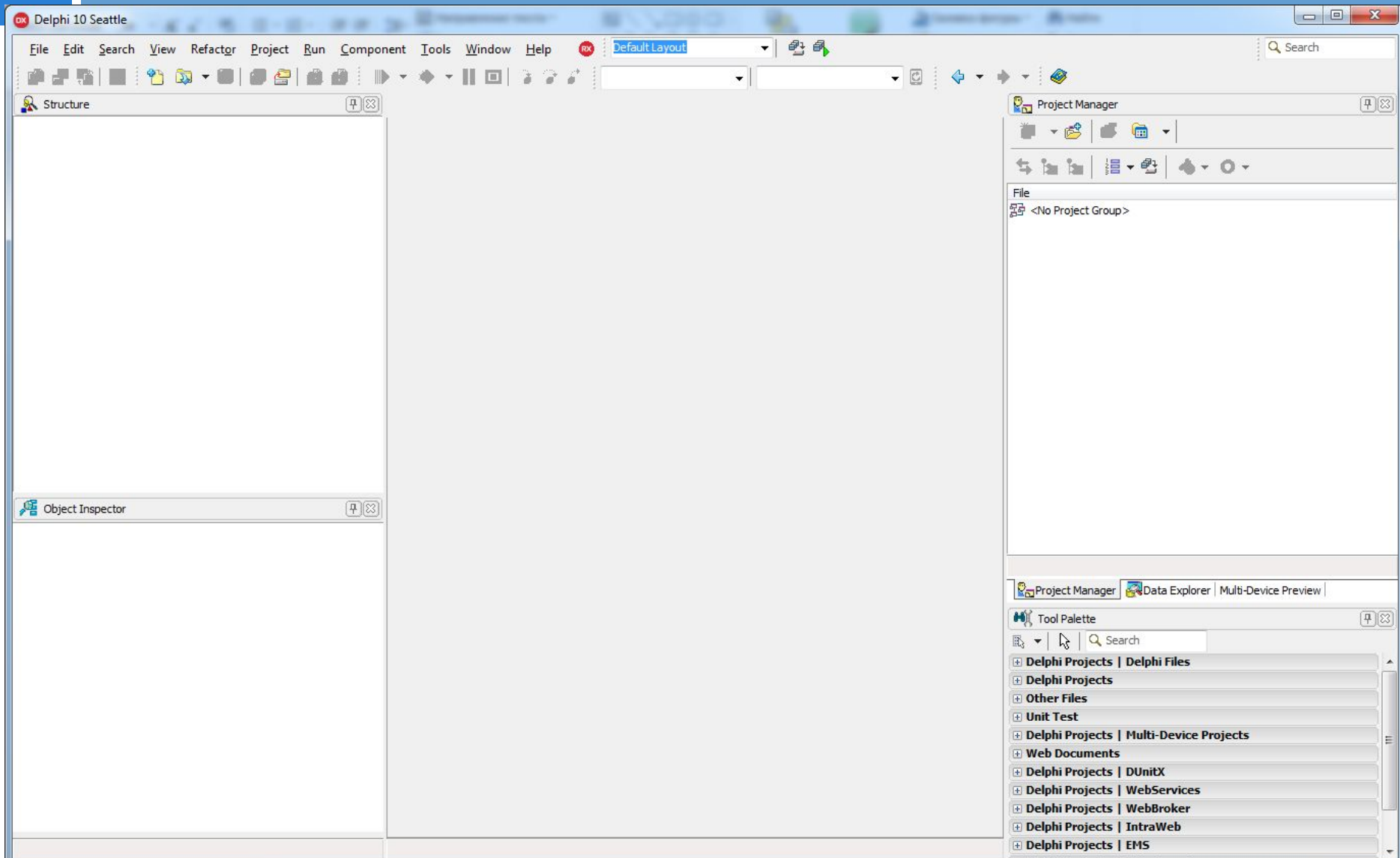
- указывает окна среды ООП, их расположение и назначение
- различает свойства компонентов (видимость, доступность, положение на форме, размеры, надпись, шрифт)
- конструирует интерфейс программы, используя основные компоненты объектно-ориентированной среды (кнопка, метка, текстовое поле, компоненты вставки графических примитивов)
- устанавливает свойства компонентов с помощью инспектора объектов
- изменяет свойства компонентов программным путем.
- запускает приложение на выполнение и проверяет правильность результата

# Проект в объектно-ориентированной среде – это набор файлов приложения.



# Запустим DELPHI

## Пуск-Embarcadero RAD Studio 10 Seattle



New ▶ VCL Forms Application - Delphi

Open... Multi-Device Application - Delphi

Open Project... Ctrl+F11 Package - Delphi

Open From Version Control...

Reopen ▶ VCL Form - Delphi

Save Ctrl+S Multi-Device Form - Delphi

Save As... Unit - Delphi

Save Project As...

Save All Shift+Ctrl+S Other...

Close

Close All

Use Unit... Alt+F11

Print...

Exit

Customize...



- СРЕДА DELPHI -система визуального ООП для создания приложений, удовлетворяющих стандартам Windows
- интегрированная среда разработки - Integrated Development Environment (IDE), позволяющая создавать, компилировать, тестировать и редактировать проект в единой среде программирования;
- Средство ускоренной разработки программ Rapid Application Development – RAD).
- **(ООП) – это методика разработки программ, в основе которой лежит понятие объекта, как некоторой структуры, описывающей объект реального мира и его поведение.**

Structure

Form1

Unit1

Form1

Project1.dproj - Project Manager

- File
- ProjectGroup1
  - Project1.exe
    - Build Configurations (Debug)
    - Target Platforms (Win32)
    - Unit1.pas

Object Inspector

Form1 TForm1

Properties Events

Search

- Action
- ActiveControl
- Align allNone
- AlignWithMargins  False
- AlphaBlend  False
- AlphaBlendValue 255
- Anchors [akLeft,akTop]
- AutoScroll  False
- AutoSize  False
- BIDiMode bdLeftToRight
- BorderIcons [biSystemMenu,biMinimize,biMaximize]
- BorderStyle bsSizeable
- BorderWidth 0

Bind Visually...

All shown

Tool Palette

- Data Access
- Standard
- FireDAC
- FireDAC UI
- FireDAC Links
- FireDAC Services
- FireDAC ETL
- Dialogs
- System
- Vista Dialogs
- Additional

Structure Unit1

- Form1

Form1

Object Inspector

Form1 TForm1

Properties Events

Search

>> Action	
ActiveControl	allNone
Align	
AlignWithMargins	<input type="checkbox"/> False
AlphaBlend	<input type="checkbox"/> False
AlphaBlendValue	255
↳ Anchors	[akLeft,akTop]
AutoScroll	<input type="checkbox"/> False
AutoSize	<input type="checkbox"/> False
BIDiMode	bd.LeftToRight
↳ BorderIcons	[biSystemMenu,biMinimize,biMaximize]
BorderStyle	bsSizeable
BorderWidth	0
Bind Visually...	

All shown

Project1.dproj - Project Manager

File

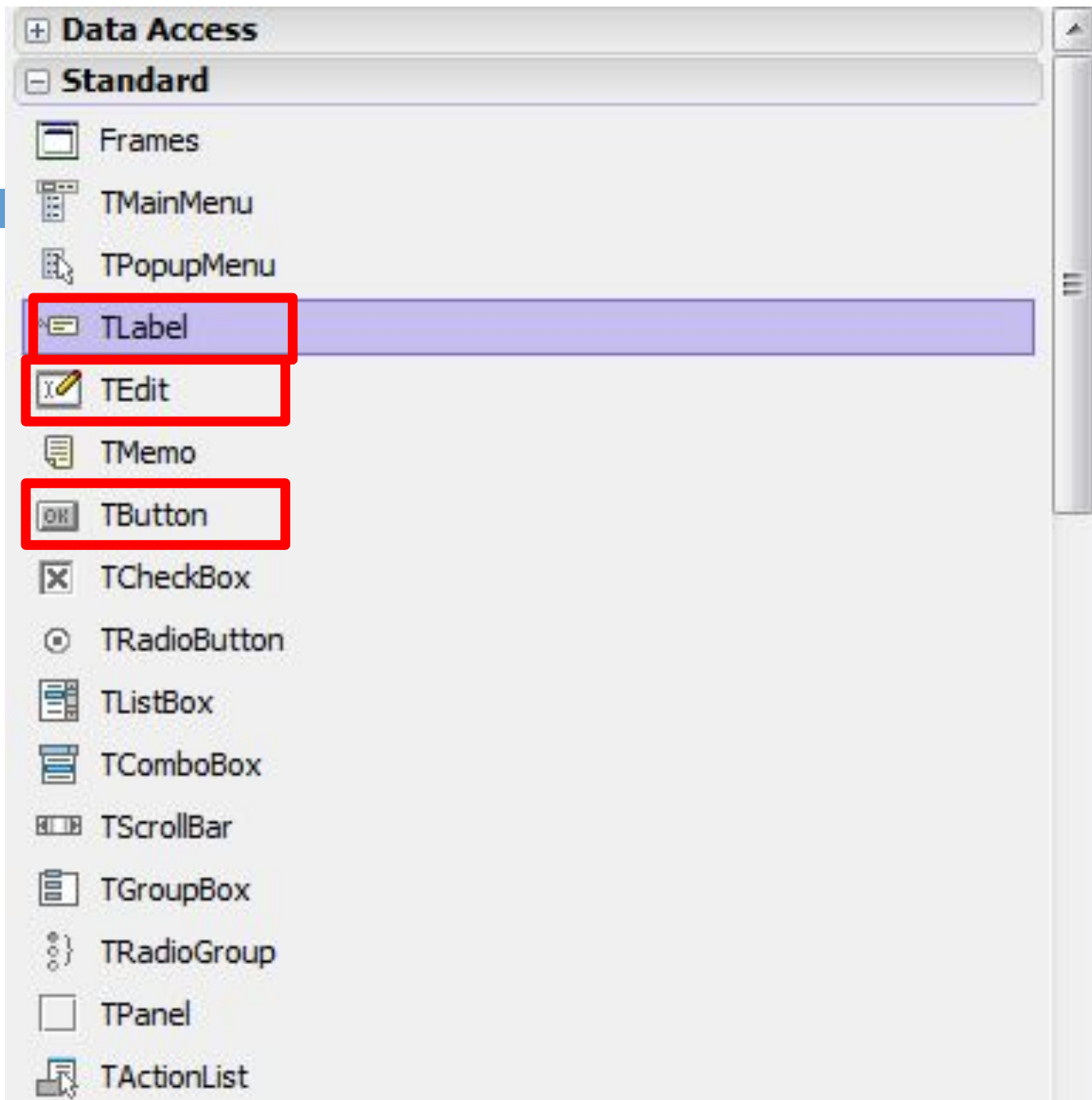
- ProjectGroup1
- Project1.exe
  - Build Configurations (Debug)
  - Target Platforms (Win32)

Project1.dproj - Project Ma... Data Explorer Multi-Device Preview

Tool Palette

Search

- Data Access
- Standard
  - Frames
  - TMainMenu
  - TPopupMenu
  - TLabel
  - TEdit
  - TMemo
  - TButton
  - TCheckBox
  - TRadioButton
  - TListBox
  - TComboBox
  - TScrollBar
  - TGroupBox
  - TRadioGroup
  - TPanel
  - TActionList
- FireDAC
  - FireDAC UI
  - FireDAC Links
  - FireDAC Services
  - FireDAC ETL

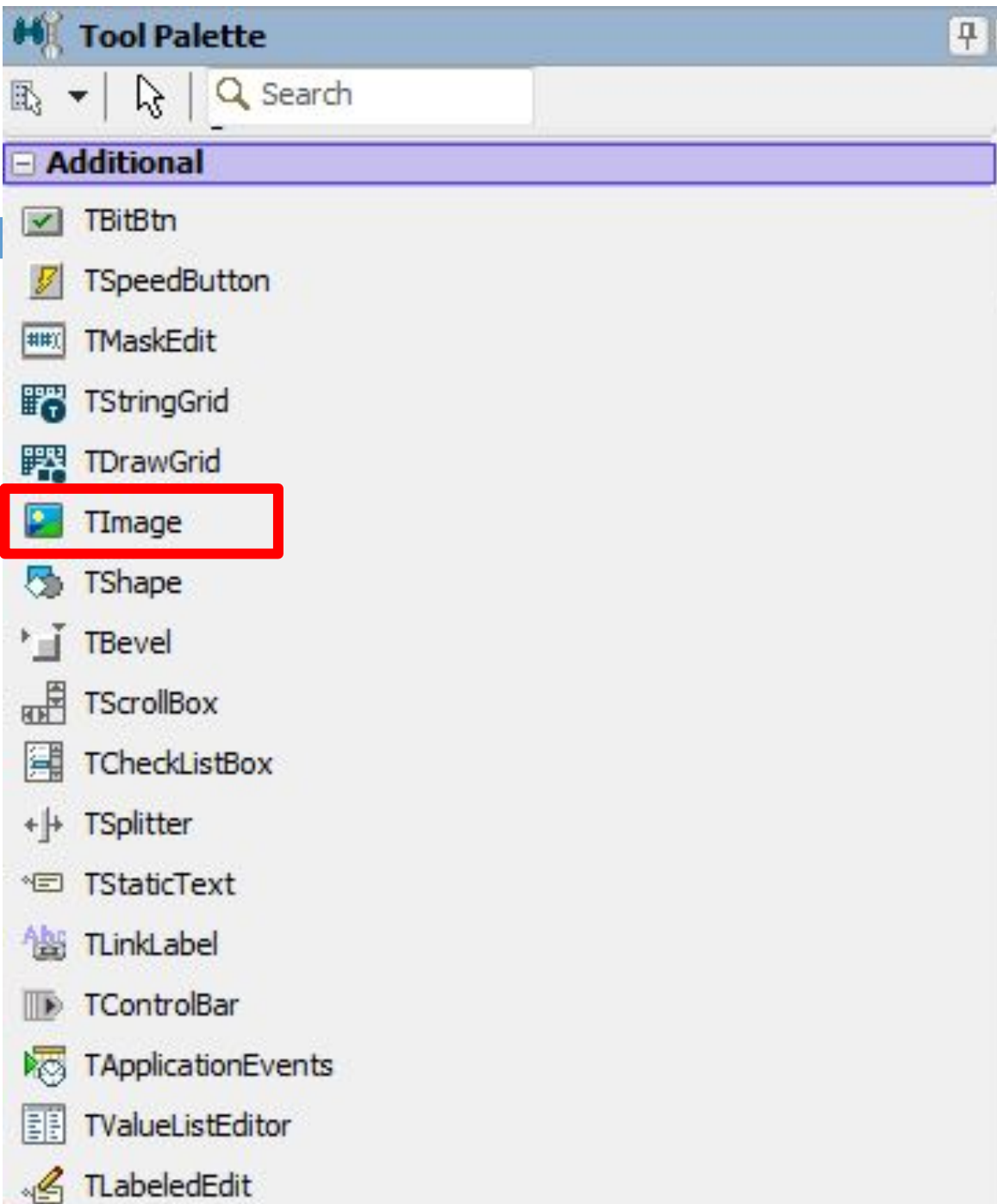


>>	Caption	Button1
	Enabled	<input checked="" type="checkbox"/> True
-	Font	(TFont) ...
	Charset	DEFAULT_CHARSET
	Color	■ clWindowText
	Height	-11
	Name	Tahoma
	Orientation	0
	Pitch	fpDefault
	Quality	fqDefault
	Size	8
	Height	25
	Left	40
	Top	96
>>	Visible	<input checked="" type="checkbox"/> True ▼
	Width	75

# Напишем программу

---

- По нажатию на кнопку размер текста в Label будет увеличиваться






Object Inspector

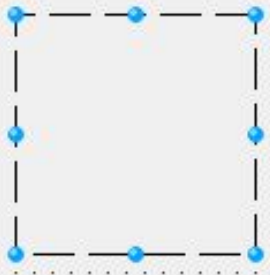
Image1 TImage

Properties Events


Search

+	LiveBindings	LiveBindings
+	LiveBindings Designer	LiveBindings Designer
+	Margins	(TMargins)
	Name	Image1
	ParentCustomHint	<input checked="" type="checkbox"/> True
	ParentShowHint	<input checked="" type="checkbox"/> True
>>	Picture	(None) 
	PopupMenu	
	Proportional	<input type="checkbox"/> False
	ShowHint	<input type="checkbox"/> False
	Stretch	<input type="checkbox"/> False
	Tag	0
	Top	152

Bind Visually...



Picture Editor



OK

Cancel

Help

Load... Save... Clear



Object Inspector

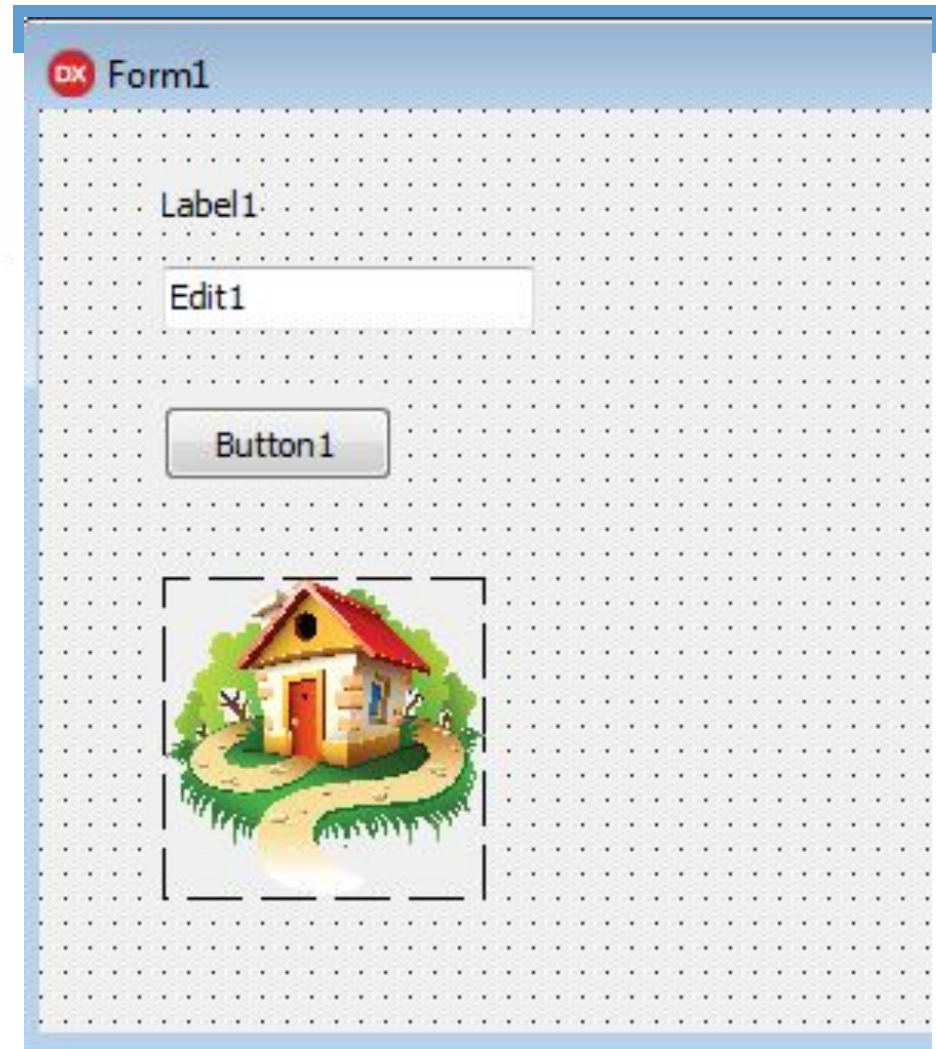
Image1 TImage

Properties | Events |

Search

+	LiveBindings	LiveBindings
+	LiveBindings Designer	LiveBindings Designer
+	Margins	(TMargins)
	Name	Image1
	ParentCustomHint	<input checked="" type="checkbox"/> True
	ParentShowHint	<input checked="" type="checkbox"/> True
>>	Picture	(None) ...
	PopupMenu	
	Proportional	<input type="checkbox"/> False
	ShowHint	<input type="checkbox"/> False
	Stretch	<input type="checkbox"/> False
	Tag	0
	Top	152

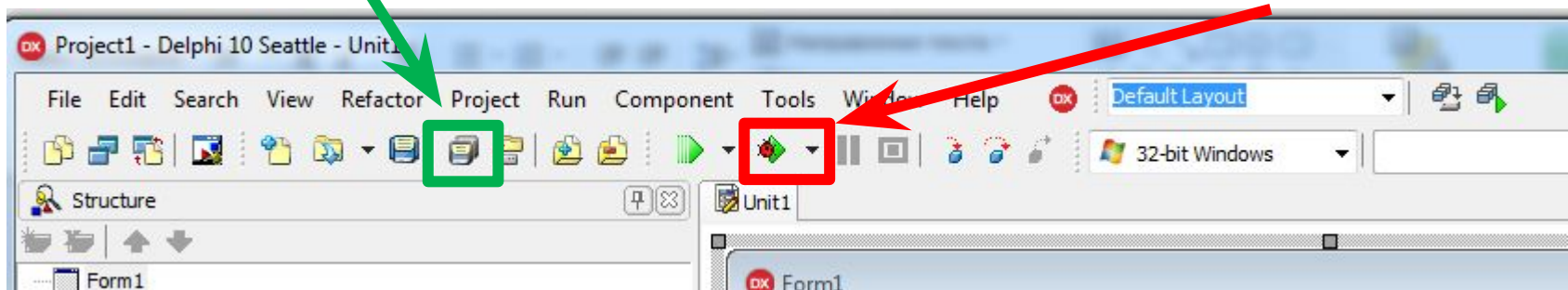
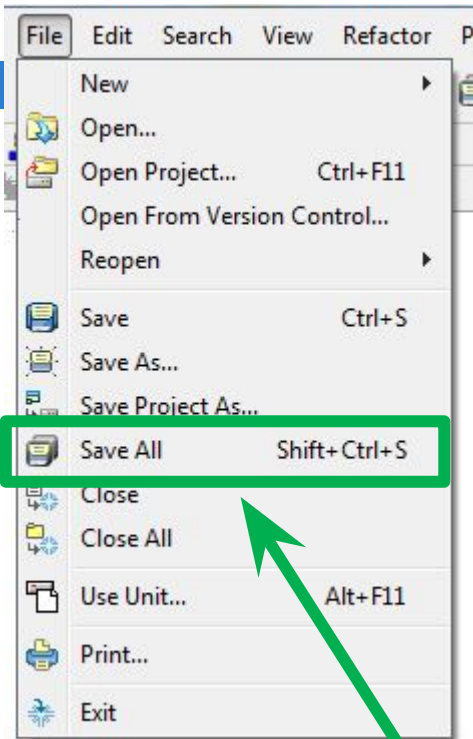
Bind Visually...



# Правило сохранения проекта:

1. Открыть среду Delphi
  2. В меню файл **File-save project AS**
  3. **Создать папку для проекта**
  4. Сохранить файл модуля Unit1.pas
  5. Сохранить файл проекта Project1.dpr
- Чтобы запустить проект на выполнение

нажмите F9 или



- главный файл проекта – **Project.dpr**;
- первый модуль программы (модуль формы) – **Unit.pas**;
- файл главной формы – **Unit.dfm**;
- файл описание ресурсов – **Project.res**;
- файл конфигурации проекта – **Project.cfg**;
- файл параметров проекта – **Project.dof**;
- файл **Project.dsk** – содержит информацию о состоянии рабочего пространства (экрана);
- дополнительные модули - **\*.pas**;
- откомпилированные модули проекта – **\*.dcu**;
- резервные копии файлов - **\*.~dp, \*.~pa, \*.~df**.



ООП, ОБЪЕКТ, ЕГО СВОЙСТВА,  
МЕТОДЫ И СОБЫТИЯ.  
КОМПОНЕНТЫ СРЕДЫ.  
ИНТЕГРИРОВАННАЯ СРЕДА  
РАЗРАБОТКИ.

# Цели обучения

- Знать определение объекта, свойств, методов, событий в объектно-ориентированном программировании
- Разрабатывать решение для проекта, используя, по крайней мере, два программных приложения (пакета) и объектно-ориентированный язык программирования
- Использовать логические операции в объектно-ориентированном языке программирования

# Знать определение объекта, свойств, методов, событий в объектно-ориентированном программировании

- объясняет понятия «объект», «свойства объекта», «метод», «событие» в объектно-ориентированном программировании
- разделяет понятия «метод» и «событие»
- приводит примеры «методов» и «событий» в объектно-ориентированном программировании

## Разрабатывать решение для проекта, используя, по крайней мере, два программных приложения (пакета) и объектно-ориентированный язык программирования

- использует константы/переменные при разработке приложения
- реализует алгоритмические структуры (линейные, разветвляющиеся) в среде программирования
- оценивает правильность работы программы с помощью тестирования, получаемых данных
- составляет тесты для проверки работоспособности своего приложения
- делает письменные комментарии при оценивании приложения одноклассников по определенному плану



# Использовать логические операции в объектно-ориентированном языке программирования

- определяет алгоритм задачи и верно строит логические отношения
- составляет условие задачи по программному коду, в котором использованы логические операции дизъюнкция, конъюнкция и инверсия, объектно-ориентированного языка программирования

- **Класс** – это сложная структура, «шаблон» включающая в себя описания данных, процедур и функций, которые могут быть выполнены над объектом

# В программе описание класса помещается в разделе описания ТИПОВ **Type**.

- **Type**
- TShcola=class
  - private
  - fname: string[15];
  - address: string[35];
  - public
  - procedure show;
- end;
- {TShcola – ИМЯ класса;  
fname, address – ИМЕНА полей;  
show – ИМЯ метода}

- **Объект** – это экземпляр класса, который имеет свои свойства, методы и события

# В программе описание объекта помещается в разделе описания переменных **var**

- **var**
- uchenik: TShcola;  
uchitel: TShcola;
- { TShcola=class  
private  
    fname: string[15];  
    address: string[35];  
    **constructor create;**  
public  
    procedure show;  
end;}

- **Метод** класса (процедуры и функции, объявление которых включено в описание класса) выполняет действия над объектами класса.
- **(то что можно делать с объектом)**
- Чтобы метод был выполнен, надо указать имя объекта и имя метода, отделив одно имя от другого точкой: `uchenik.show;`

**Событие** - действие которое происходит в объектом в данное время.

Простейшие события, события, связанные с мышкой (они есть практически у всех видимых объектов) или событие Click для кнопки TButton.

обработчик данного события:

```
procedure TForm1.FormClick(Sender: TObject);  
begin  
.....  
end;
```

- СРЕДА DELPHI -система визуального ООП для создания приложений, удовлетворяющих стандартам Windows
- интегрированная среда разработки - Integrated Development Environment (IDE), позволяющая создавать, компилировать, тестировать и редактировать проект в единой среде программирования;
- Средство ускоренной разработки программ Rapid Application Development – RAD).
- **(ООП) – это методика разработки программ, в основе которой лежит понятие объекта, как некоторой структуры, описывающей объект реального мира и его поведение.**



# Найдите соответствия

<b>Объект</b>	это функция или процедура, принадлежащая какому-то классу или объекту.
<b>Свойства</b>	это сущность, способная сохранять свое состояние (информацию) и обеспечивающая набор операций (поведение) для проверки и изменения этого состояния.
<b>Метод</b>	способ доступа к внутреннему состоянию объекта, имитирующий переменную некоторого типа.

# Принципы ООП

- ▣ Инкапсуляция – возможность объединения данных и операций в одно целое. Не бывает объекта без свойств и свойства без объекта.
- ▣ Button1.color:=clred;

# Принципы ООП

- ▣ **Наследование** – возможность создания новых производных классов на основе уже существующих.
- ▣ **у компонента** Button (кнопка) есть потомок Bitbtn(комбинированная кнопка), которая унаследовала все свойства кнопки Button , но и имеет собственные свойства, которых нет у ее предка Button обладает всеми свойствами

# Принципы ООП

- ▣ Полиморфизм – возможность модифицирования методов в производных классах.
- ▣ Одна и та же операция может выполняться по-разному для разных объектов.
- ▣ Например операция +(плюс) полиморфная. Цифры складывает, символы – склеивает, а множества объединяет.

# 2 звезды и пожелание

2 звезды=2 момента, в  
которых вы  
преуспели

Пожелание=что-то, что  
может быть  
улучшено

