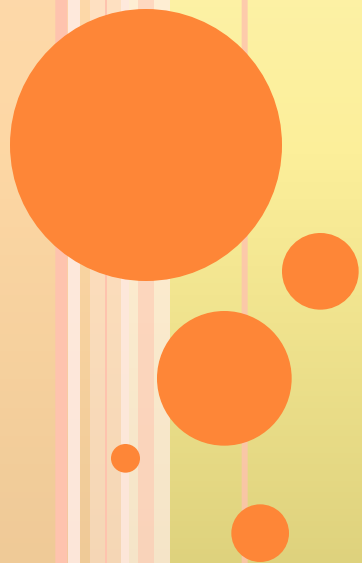


# **ВВЕДЕНИЕ В ПРОГРАММИРОВАНИЕ В СРЕДЕ BORLAND DELPHI**



# НЕСКОЛЬКО СЛОВ О СРЕДЕ DELPHI

Delphi — это среда быстрой разработки, в которой в качестве языка программирования используется язык Delphi. Язык Delphi — строго типизированный объектно-ориентированный язык, в основе которого лежит хорошо знакомый программистам Object Pascal.

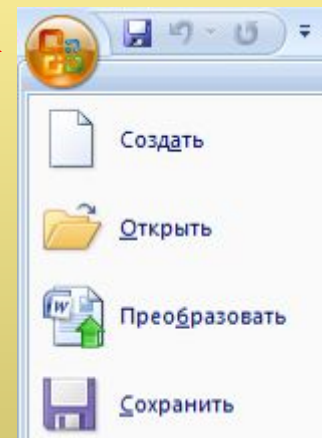
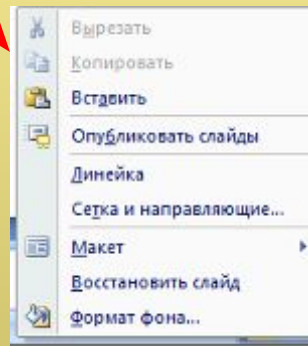
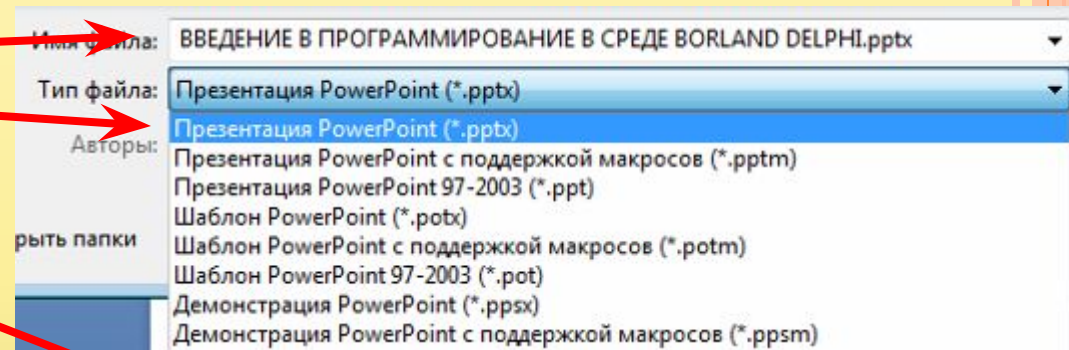
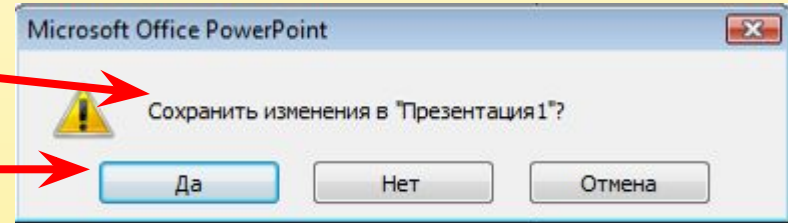
## **Что значит «объектно-ориентированный язык»?**

Это означает, что мы будем программировать работу объектов, т.е. будем ориентироваться на то, какие объекты в программе можно использовать, какие у этих объектов свойства и что эти объекты могут делать (или что можно делать с объектами).



# ОБЪЕКТЫ – ЭТО ЧТО?

- Это сообщения и кнопки
- Окна ввода и раскрывающиеся списки
- Главные меню и контекстные



# ОБЪЕКТЫ – ЭТО ЧТО?

**Объект – это основная единица в объектно-ориентированном программировании.**

**Объект имеет *свойства, методы*, определяющие его поведение и реагирует на *события*.**

**Если говорить образно, то объекты – это «существительные», свойства объекта – «прилагательные», а методы объекта – это «глаголы».**



# ОБЪЕКТЫ – ЭТО ЧТО?

Его  
свойст  
ва  
и  
методы

The screenshot shows the Object Inspector for a **Button1** control. It has two tabs: **Properties** and **Events**. The **Properties** tab is active, displaying a list of properties and their values. The properties are grouped into sections: **Locale**, **Localizable**, **Constraints**, **Font**, **Miscellaneous**, and **Visual**.

Property	Value
PopupMenu	
<b>Locale</b>	
BiDiMode	bdLeftToRight
ParentBiDiMode	true
<b>Localizable</b>	
BiDiMode	bdLeftToRight
Caption	<b>Кнопка</b>
Constraints	(TSizeConstraints)
Font	<b>(TFont)</b>
Height	<b>89</b>
Hint	
Left	<b>32</b>
ParentBiDiMode	true
ParentFont	<b>false</b>
Top	<b>32</b>
Width	<b>193</b>
<b>Miscellaneous</b>	
Cancel	false
Default	false
ModalResult	mrNone
Name	Button1
Tag	0
WordWrap	false
<b>Visual</b>	
Align	alNone

The screenshot shows the Design View of a **Form1** control. The form is a rectangular window with a blue title bar and standard Windows window controls. The main area is a grid with a light gray background. A button control is placed on the grid, with the text **Кнопка** centered on it. The button has a light gray background and a thin black border. Red arrows point from the text 'Его свойства и методы' to the 'Properties' tab and from the text 'объект' to the button on the form.

объект



# ОБЪЕКТ ЯВЛЯЕТСЯ ЭКЗЕМПЛЯРОМ ТОГО ИЛИ ИНОГО КЛАССА

- Ученик – экземпляр класса «Школьники»
- Ноутбук – экземпляр класса «Компьютеры»
- Земля – экземпляр класса «Планеты»

Приведите свои примеры.

У каждого класса есть свойства, например, класс «Школьники» имеет свойства: возраст детей, способности детей, школьная форма, наличие учебников.



# ПРИМЕР

Каждый из классов обладает специфическим набором свойств, методов и событий. Например, в приложении Word существует класс объектов «Документ»(Document), который обладает определенными наборами:

- Свойств: имя (Name), полное имя (FullName) и так далее;
- Методов: открыть документ (Open), напечатать документ (PrintOut), сохранить документ (Save);
- Событий: создание документа (Document\_New), закрытие документа (Document\_Close) и т.д.



# ОСНОВНЫЕ ПРИНЦИПЫ ОБЪЕКТНО – ОРИЕНТИРОВАННОГО ПРОГРАММИРОВАНИЯ

- Инкапсуляция
- Наследование
- Полиморфизм





# **ИНКАПСУЛЯЦИЯ -**

**единство внутри класса трех составляющих его сущностей: полей, методов и свойств.**

**Например, класс TForm – содержит (включает, инкапсулирует) в себе все необходимое для создания Windows – окна.**



# НАСЛЕДОВАНИЕ -

**процесс порождения объектов – потомков от объектов – родителей. При этом к имеющимся свойствам и методам объекта – родителя добавляются новые, характерные только для объекта – потомка класса. Удалить какие-либо элементы родителя в объекте – потомке нельзя. Но можно от потомка породить следующий объект и таким образом сформировать иерархию класса или дерево объектов.**



# **ПОЛИМОРФИЗМ -**

**возможность реализации одноименных, но алгоритмически разных методов в классе-родителе и в классе – потомке. То есть, метод родителя перекрывается (изменяется) в классе – потомке, в результате чего объект – потомок приобретает отличную от родителя функциональность.**



# ВИД ЭКРАНА ПОСЛЕ ЗАПУСКА DELPHI

The image shows the Borland Developer Studio 2006 interface with several components highlighted by red boxes and labels:

- Project1 - Borland Developer Studio 2006 - Unit1**: The main application window at the top.
- СПИСОК ОБЪЕКТОВ**: A red label pointing to the **Structure** window on the left, which shows a tree view of the project's components.
- ОКНО ФОРМЫ**: A red label pointing to the central **Form1** design window, which is a grid-based workspace for designing the application's form.
- СПИСОК ФАЙЛОВ ПРИЛОЖЕНИЯ**: A red label pointing to the **Project Manager** window on the right, which displays the project's file structure.
- ИНСПЕКТОР ОБЪЕКТОВ**: A red label pointing to the **Object Inspector** window on the left, which shows the properties and events of the selected object.
- ПАЛИТРА КОМПОНЕНТОВ**: A red label pointing to the **Tool Palette** window on the right, which contains a collection of standard Delphi components like TFrame, TMainMenu, TEdit, etc.
- ОКНО РЕДАКТОРА КОДА**: A red label pointing to the **Code Editor** window at the bottom, which is currently empty.

The interface includes a menu bar (File, Edit, Search, View, Refactor, Project, Run, Component, Tools, StarTeam, Window, Help), a toolbar, and a status bar at the bottom showing the current line (1: 1) and the active mode (Code/Design/History).

# ЗАДАНИЕ

- Открыть приложение Пуск\Программы\Borland Developer Studio 2006\Delphi for Microsoft Win32, найдите все элементы окна, описанные в предыдущем слайде

