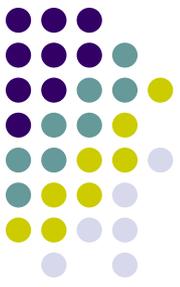


Библиотека VCL в Delphi





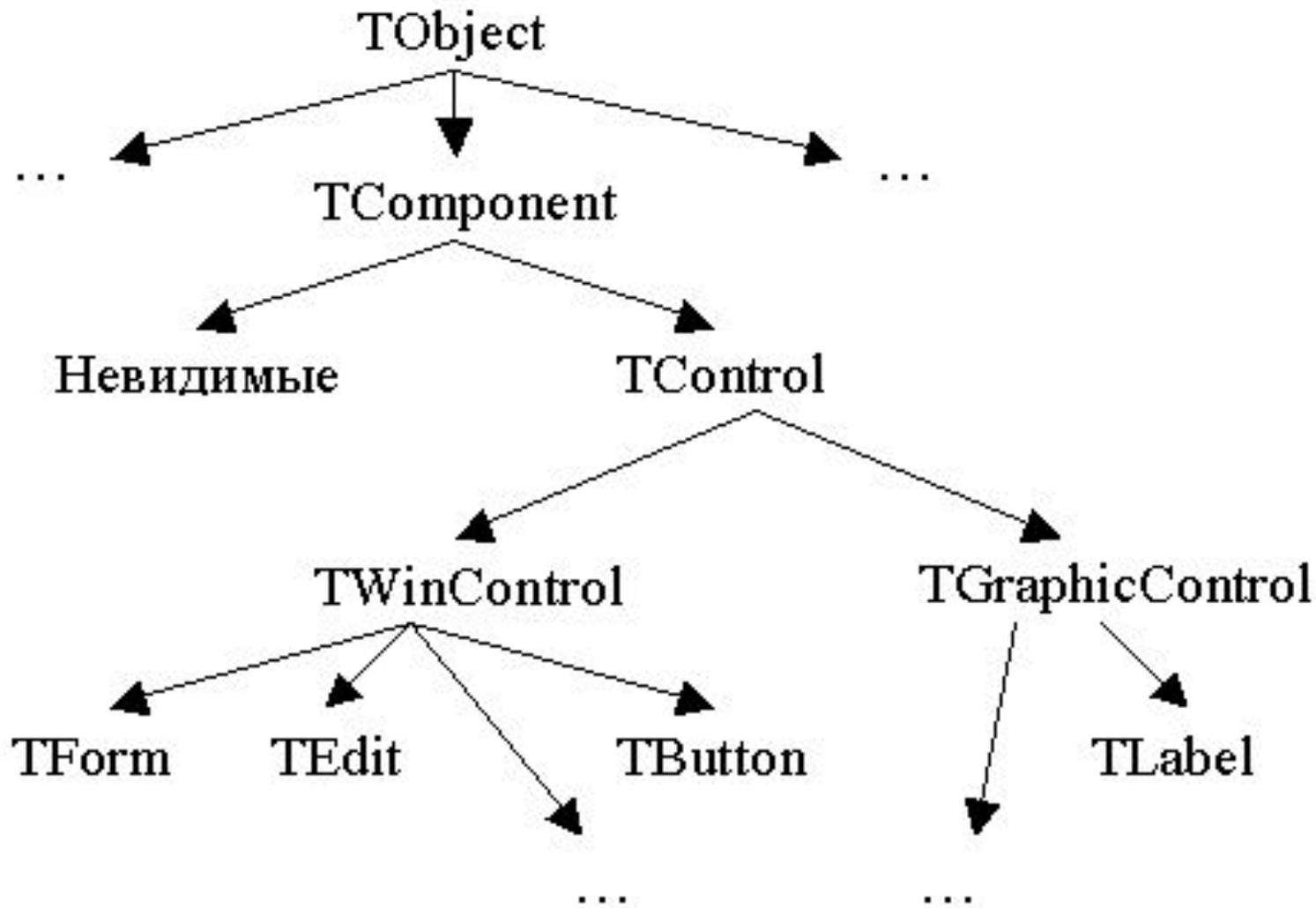
Визуальный компонент - объект,
большинство свойства которого доступны
и видимы еще на этапе разработки.



Во всех системах визуального программирования принята концепция **двойного инструментария (Two Way Tools)**, которая позволяет менять свойства компонентов:

- в процессе визуального конструирования с помощью мыши и клавиш (**Design Time**);
- в процессе работы приложения с помощью операторов языка программирования (**Run Time**).

VCL - Visual Component Library (визуальная библиотека компонентов)





- **TObject** - объекты. В Object Pascal все классы являются потомками этого класса.
- **TComponent** - компоненты. Все элементы VCL являются потомками этого класса.
- **TControl** - элементы управления. Одинаково выглядят в Design Time и Run Time.



- **TWinControl** - оконные элементы управления. Могут получать фокус ввода.
- **TGraphicControl** - графические элементы управления. Не имеют фокуса и уникального дескриптора, не получают сообщений.



Свойства (Properties) хранят данные
определенного типа

Свойства класса TComponent



- **Name**: `string`; ИМЯ КОМПОНЕНТА
- **Tag**: `integer`; МОЖЕТ ИСПОЛЬЗОВАТЬСЯ ПРОГРАММИСТОМ ПО СВОЕМУ УСМОТРЕНИЮ



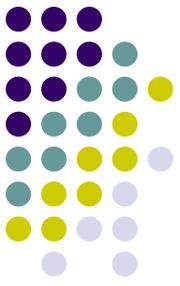
Свойства класса TControl

- **Caption, Text**: `string`; - надпись на компоненте
- **Left, Top**: `integer`; координаты верхнего левого угла компонента
- **Height, Width**: `integer`; высота и ширина компонента
- **ClientHeight, ClientWidth**: `integer`; высота и ширина части компонента, доступной для пользователя



Свойства класса TControl

- **Visible**: `boolean`; ВИДИМОСТЬ КОМПОНЕНТА
- **Enabled**: `boolean`; ДОСТУПНОСТЬ КОМПОНЕНТА
- **AutoSize**: `boolean`; КОМПОНЕНТ принимает размер в зависимости от его содержимого.
- **Color**: `TColor=integer`; ЦВЕТ компонента (`$000000` - `$FFFFFF` (BGR) `clBlack,clRed,...`)



Свойства класса TControl

- **Align**: TAlign = (alNone, alTop, alBottom, alLeft, alRight, alClient); **выравнивание компонента**
- **Alignment**: TAlignment = (taLeftJustify, taRightJustify, taCenter); **выравнивание содержимого компонента**

Свойства класса TControl

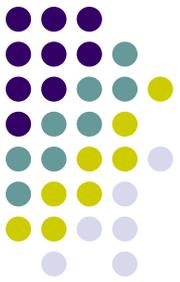


- **Hint**: `string`; всплывающая подсказка
- **ShowHint**: `boolean`; показать подсказку
- **Cursor**: `TCursor=integer`;
-`crDefault`, `crHourGlass`, `crHandPoint`,
...: курсор



Свойства класса TControl

- **Font**: TFont шрифт
 - **Font.Color**: integer цвет шрифта
 - **Font.Size**: integer размер шрифта в пунктах
 - **Font.Name**: string название шрифта
 - **Font.Style**: TFontStyles = set of TFontStyle, где TFontStyle = (fsBold, fsItalic, fsUnderline, fsStrikeOut); стиль шрифта (жирный, курсив, подчеркнутый, зачеркнутый)



Методы (Methods) - процедуры и функции,
связанные с компонентами.

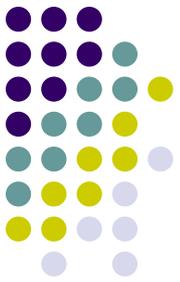
Методы класса TControl



- **Hide**; скрыть компонент
- **Show**; показать компонент

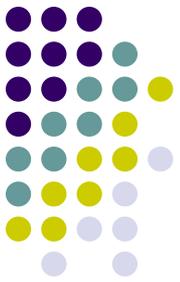
Метод класса TWinControl

- `SetFocus`; получить фокус ввода



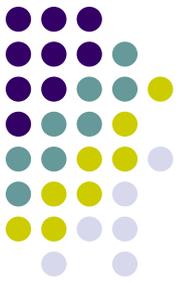
Методы класса TForm

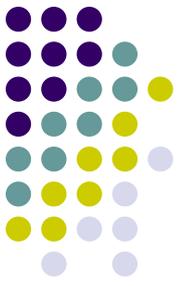
- `Close`; закрыть форму



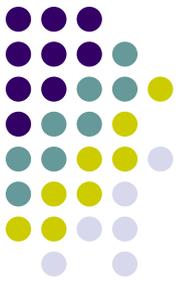
Методы класса TEdit

- `Clear`; ОЧИСТИТЬ



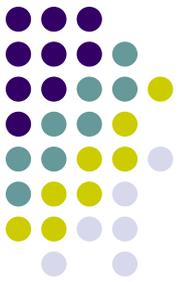


- С каждым компонентом связан определенный список **событий (Events)**, на которые он способен реагировать.
- Для каждого события, связанного с компонентом, можно написать специальную процедуру - **обработчик события (Events Handler)**.
- Обработчик каждого события относится к определенному классу, который *определяется набором передаваемых в обработчик параметров.*



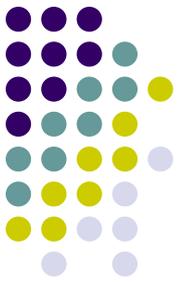
События класса TControl

- **OnClick**: TNotifyEvent
(Sender: Object) **ЩЕЛЧОК МЫШИ**
- **OnDb1Click**: TNotifyEvent
(Sender: Object) **ДВОЙНОЙ ЩЕЛЧОК
МЫШИ**



Оператор приведения типа

- `(Sender as Класс) .Свойство;`
или
- `Класс (Sender) .Свойство;`



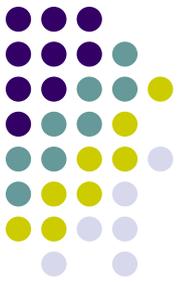
События класса TControl

- **OnMouseMove** :

```
TMouseEvent (Sender: TObject;  
Button: TMouseButton; Shift:  
TShiftState; X, Y: Integer)  
перемещение мыши
```

- **OnMouseDown** : TMouseEvent нажата
кнопка мыши

- **OnMouseUp** : TMouseEvent отпущена
кнопка мыши

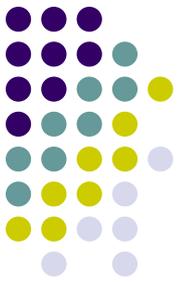


События класса TControl

Параметры событий:

- **Button:** TMouseButton = (mbLeft, mbRight, mbMiddle) – нажатая кнопка мыши;
- **Shift:** TShiftState = set of (ssShift, ssAlt, ssControl, ssLeft, ssRight, ssMiddle, ssDouble) – управляющие клавиши;
- **X, Y:** integer; - координаты мыши на компоненте.

События класса TWinControl



- **OnEnter**: TNotifyEvent; получить фокус ввода
- **OnExit**: TNotifyEvent; потерять фокус ввода

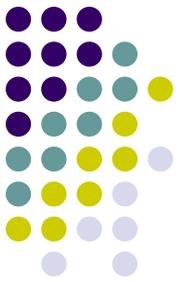
События класса TWinControl

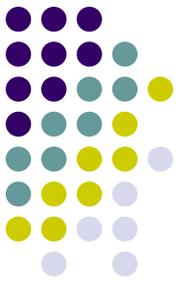


- **OnKeyPress**: TKeyPressEvent (Sender: TObject; **var** Key: Char); нажата клавиша
- **OnKeyDown**: TKeyEvent (Sender: TObject; **var** Key: Word; Shift: TShiftState); прижата клавиша
- **OnKeyUp**: TKeyEvent (Sender: TObject; var Key: Word; Shift: TShiftState); отпущена клавиша

Событие для TEdit

- **OnChange**; изменение содержимого





События для TForm

- **OnCreate**; создание формы
- **OnClose**; закрытие формы
- **OnActivate**; форма становится активной