

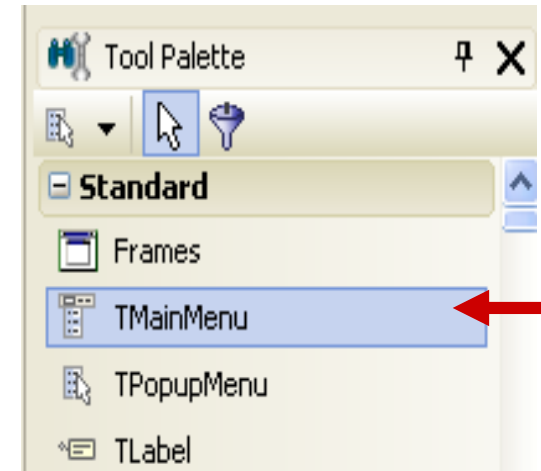
# «Проектирование главного и локального меню»

1. Проектирование главного меню.
2. Проектирование локального меню.
- 3 Проектирование панели инструментов.
4. Проектирование строки состояния.

# 1. Проектирование главного меню.

- Отображение в форме *главного меню* (обычно это строка под заголовком окна) обеспечивает компонент **MainMenu**:

- Компонент **MainMenu** является невидимым – такие компоненты на стадии проектирования отображаются на форме в виде пиктограммы. Их действие проявляется при выполнении программы.



# 1. Проектирование главного меню.

- Основное свойство компонента `MainMenu`: *Items* – Оно обеспечивает доступ к пунктам меню.
- Редактирование пунктов меню выполняется в специальном окне *Дизайнера Меню (Menu Designer)*.
- Вызов Дизайнера Меню осуществляется из локального меню компонента `MainMenu`, или двойным щелчком по значку компонента на форме.

# 1. Проектирование главного меню.

Дерево просмотра объект... [X]

Инспектор объектов

OpenItem TMenuItem

Properties | Events

<b>Action</b>	
AutoCheck	False
AutoHotkeys	maParent
AutoLineReduction	maParent
Bitmap	(None)
Break	mbNone
Caption	Открыть...
Checked	False
Default	False
Enabled	True
GroupIndex	0
HelpContext	0
Hint	Подсказка в с
ImageIndex	-1
Name	OpenItem
RadioItem	False
Shortcut	F3
SubMenuItem	
Tag	0
Visible	True

All shown

PictureForm.MainMenu

Файл Вид

Открыть...	F3
Сохранить как...	F2
Закреть	F4
Выход F10	

О программе.

- Пункты меню можно добавить, удалить, отредактировать.

# 1. Проектирование главного меню.

- Пункты меню являются компонентами **MenuItem** внутри компонента **MainMenu**.

- Их основные свойства:

**Caption** - Текст пункта меню.

**Checked** - Если равно True, то пункт меню содержит метку: “птичку” или жирную точку в зависимости от значения свойства **RadioItem**.

**Enabled** - Определяет, доступен ли пункт меню пользователю.

# 1. Проектирование главного меню.

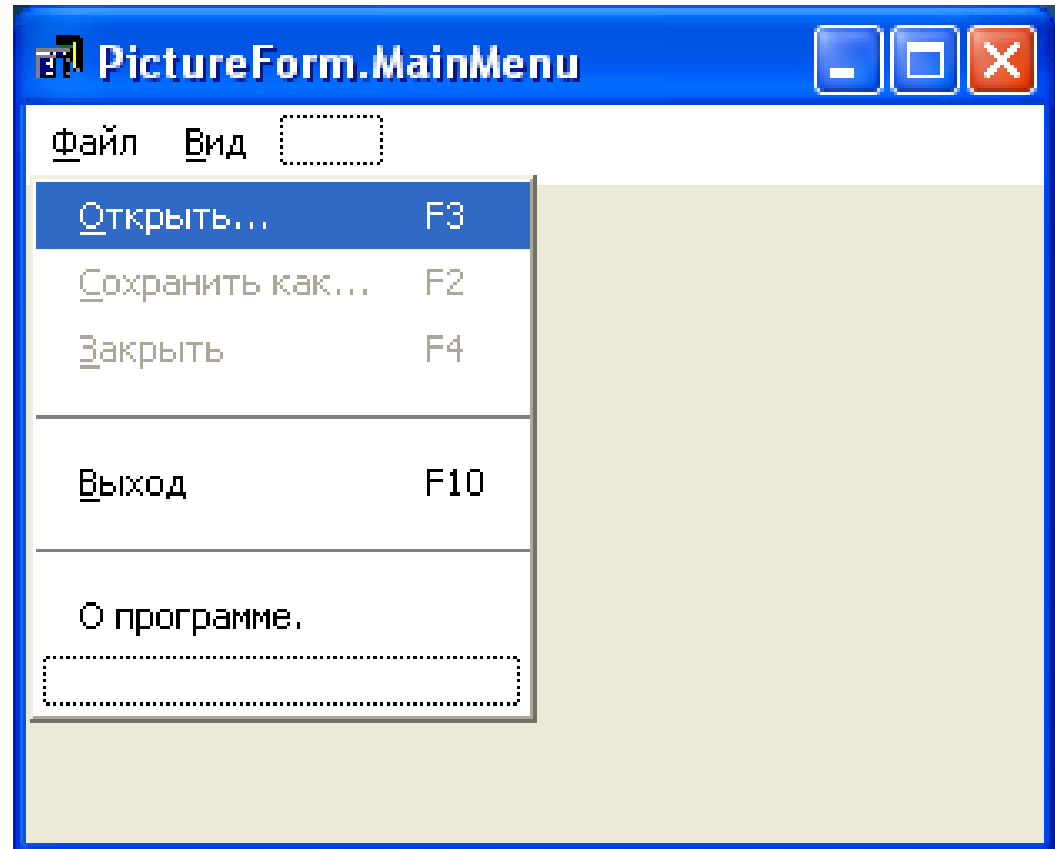
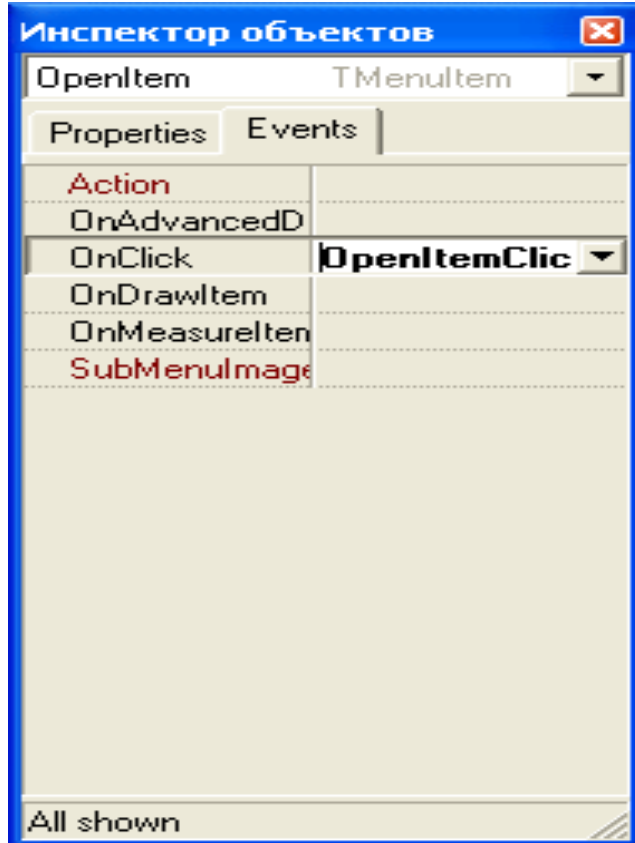
**Radioltem** - Если равно True, то метка пункта имеет вид жирной точки и несколько пунктов с одинаковым значением свойства **GroupIndex** работают как зависимые переключатели - установка у одного пункта свойства **Checked** в True снимает метку с другого пункта.

**Visible** - Определяет, виден ли пункт меню пользователю.

**ShortCut** - Комбинация клавиш для выполнения команды, не открывая меню.

# 1. Проектирование главного меню.

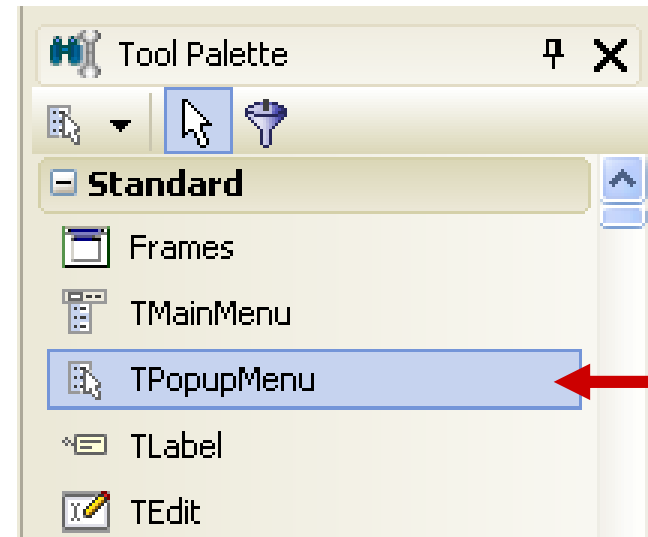
- Обработка команд меню заключается в написании обработчика события *OnClick* для каждого пункта меню.





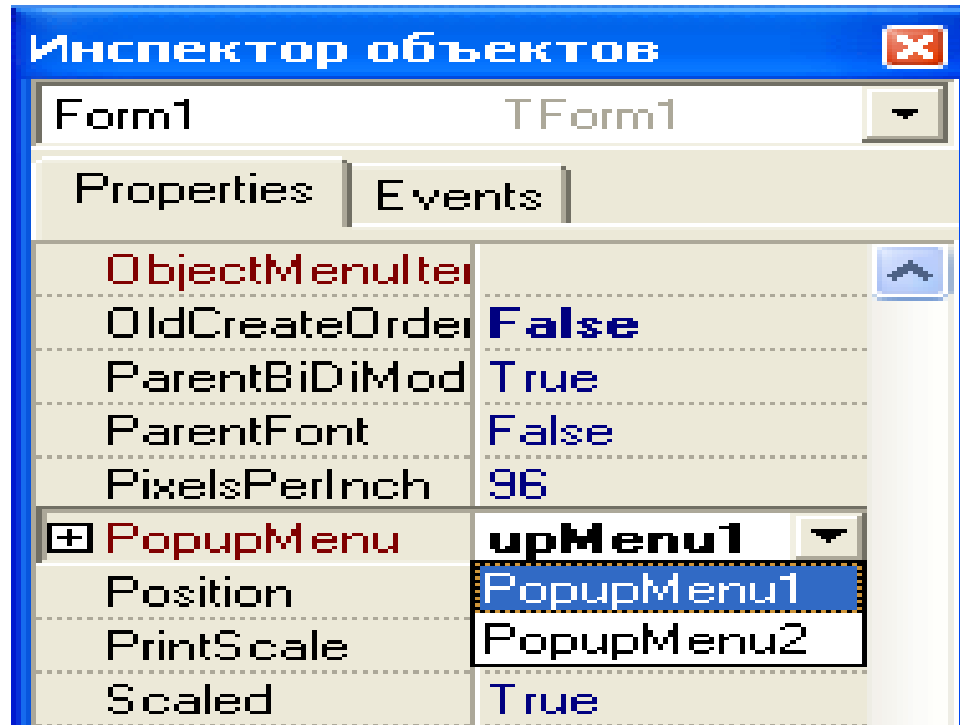
## 2. Проектирование локального меню.

- Отображение в форме *локального меню* (это меню, которое появляется при нажатии правой кнопки на каком-либо компоненте) обеспечивает компонент **PopUpMenu**:
- Проектирование локального меню осуществляется аналогично главному, при помощи дизайнера меню.



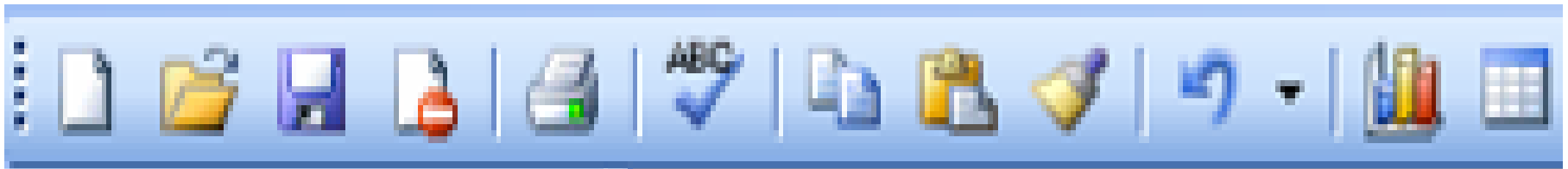
## 2. Проектирование локального меню.

- Компонент, который может обладать локальным меню, имеет свойство **PopupMenu**, значением этого свойства является имя одного, из имеющихся компонентов **PopupMenu**.



### 3. Проектирование панели инструментов.

- Панель инструментов (**toolbar**) — это расположенная снизу строки главного меню панель с командными кнопками, выпадающими списками и другими управляющими элементами. Кнопки инструментальной панели служат ускорителями для наиболее часто используемых команд меню.



### 3. Проектирование панели инструментов.

- В Delphi панель инструментов строится из нескольких компонентов. Панель делается из компонента Panel (раздел Standard), кнопки-инструменты — из компонентов SpeedButton (раздел Additional), рельефная канавка между инструментальной панелью и меню — из компонента Bevel (раздел Additional).

# 3. Проектирование панели инструментов.

## • Основные свойства компонента SpeedButton:

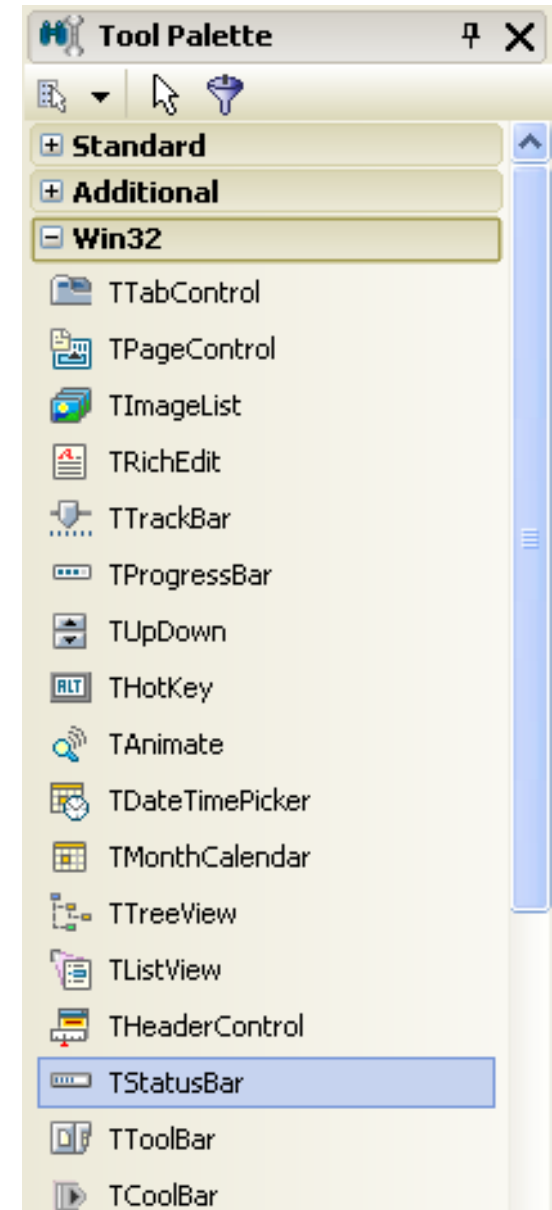
Свойство	Описание
AllowAllUp	Если равно True, то все кнопки одной группы могут находиться в отжатом состоянии.
GroupIndex	Задаёт группу кнопок, работающих согласованно.
Down	Если равно True, то кнопка нажата. Иначе кнопка отжата.
Caption	Текст на кнопке.
Glyph	Картинка на кнопке.
Layout	Положение картинки относительно текста.
Margin	Расстояние от границы кнопки до картинки. Если оно равно -1, то картинка вместе с текстом центрируются на кнопке.
NumGlyphs	Количество образов кнопки на картинке Glyph. Delphi отображает один из образов в зависимости от состояния кнопки.
Spacing	Расстояние от картинки до текста. Если оно равно -1, то текст центрируется между картинкой и границей кнопки.

### 3. Проектирование панели инструментов.

- При нажатии кнопок возникают события **OnClick**, которые должны обрабатываться точно так же, как и команды меню, надо подключить к кнопкам уже существующие обработчики событий.
- Все визуальные компоненты Delphi могут иметь подсказки. Подсказки бывают двух видов: всплывающие подсказки и подсказки в строке состояния. Оба вида подсказок хранятся в значении одного и того же свойства **Hint**. Формат свойства следующий:  
*<всплывающая подсказка>|<подсказка в строке состояния>*
- Логическое свойство **ShowHint** вкл\откл отображение подсказок.

## 4. Проектирование строки состояния.

• Строка состояния — это панель в нижней части окна, предназначенная для вывода вспомогательной информации: параметров документа, подсказок к пунктам меню и т.д. В Delphi она организуется с помощью компонента StatusBar.



## 4. Проектирование строки состояния.

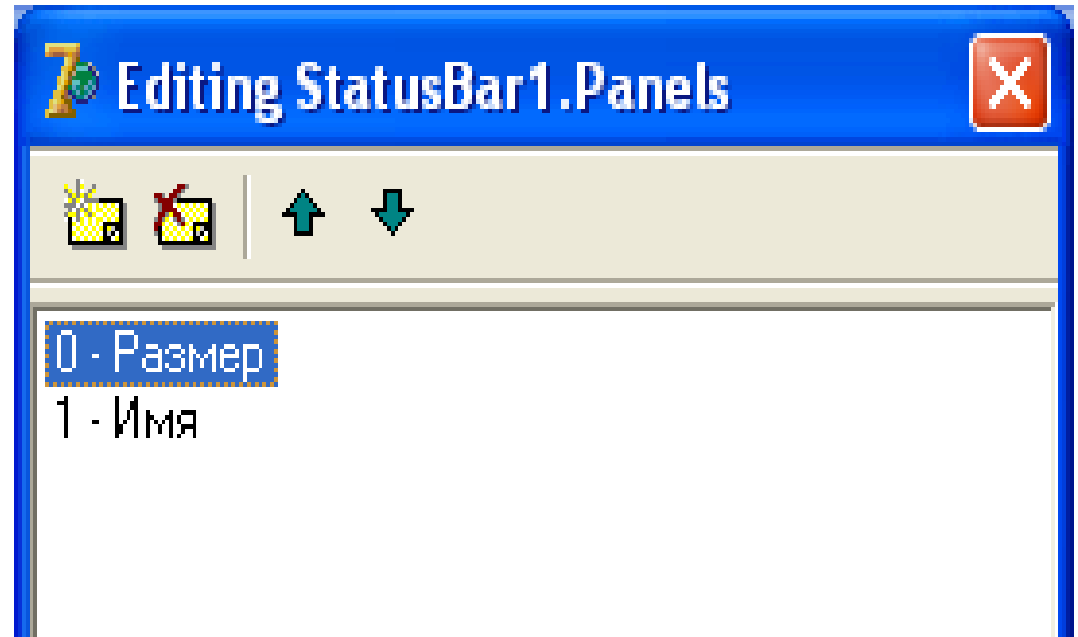
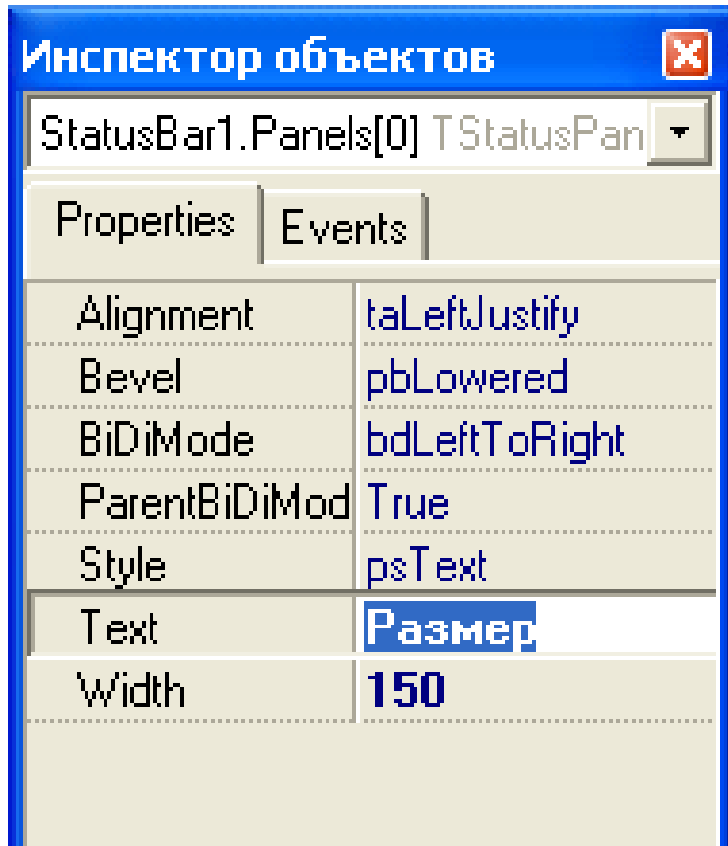
### • Основные свойства компонента StatusBar.

Свойство	Описание
Align	Способ выравнивания строки состояния в пределах владельца (формы или компонента).
Panels	Информационные панели, отображаемые на строке состояния.
SimplePanel	Если равно True, то вместо информационных панелей на строке состояния отображается одна простая строка текста, хранящаяся в свойстве SimpleText.
SimpleText	Строка текста, отображаемая в строке состояния, когда свойство SimplePanel равно True.
SizeGrip	Если равно True, то в правом нижнем углу строки состояния рисуется «гармошка», работающая как размерная рамка.



## 4. Проектирование строки состояния.

• Создание и редактирование информационных панелей (свойство Panels) выполняется в специальном редакторе:



## 4. Проектирование строки состояния.

- Объект `Panels` содержит массив `Items`, элементами которого являются объекты-панели. Каждая панель имеет свойство `Text`, в котором хранится отображаемый на панели текст. Итак, установка содержимого строки состояния может выглядеть так:

```
StatusBar1.Panels.Items[0].Text :=  
    Format('%d x %d', [Image1.Picture.Width,  
    Image1.Picture.Height]);  
StatusBar1.Panels.Items[1].Text :=  
    OpenFileDialog1.FileName;
```

## 4. Проектирование строки состояния.

- Пример:

