

ЛЕКЦИЯ 3

ОБЕКТИ НА ГПИ

 **Общи постановки**

 **Основни свойства**

 **Основни събития**

 **Работа с форми**

 **Основни обекти на ГПИ**

 **Менюта, графика и анимация**

 **Допълнителни средства**

ОБЩИ ПОСТАНОВКИ

- ❶ Формите са **основен елемент на ГПИ**, който съответства на понятието **прозорец**.
- ❷ Формите **носят** върху себе си останалите **компоненти** (елементи, **Controls**) **на ГПИ**.
- ❸ Формите **се проектират** от потребителя и **имат** **статут на** новосъздаден от него **клас** от обекти.
- ❹ Елементите **на ГПИ**, които носи дадена форма, са **екземпляри на** предварително създадени (**готови**) **класове** от обекти.
- ❺ Подобно на формите **някои елементи на ГПИ** имат възможност да **носят върху себе си други** елементи на ГПИ.
- ❻ **Елементите на ГПИ се рисуват при проектиране!**

СВОЙСТВА НА ОБЕКТИТЕ

- ❶ Всеки клас от обекти има специфични **свойства** и **методи**, чрез които става **използването** на неговите **екземпляри**.
- ❷ **Екземпляр** от даден клас се цитира чрез име или чрез **указател** към него (**обектова променлива**).
- ❸ **Свойствата** дават възможност за **управление** на екземпляра и **сведения** за неговото състояние.
- ❹ Някои от **свойствата** са достъпни само по време на **проектиране**, други – само **при изпълнение**, трети – и в **двете фази**.
- ❺ Някои **свойства** са достъпни само **за четене**, други – само **за запис**, а трети – **и за двете**.

МЕТОДИ НА ОБЕКТИТЕ

- ❶ **Всеки клас** от обекти **има** специфични **методи**, чрез които става възможно да се използват неговите екземпляри.
- ❷ Част от **методите** се **нуждаят от** определен брой **фактически параметри**, чрез което те **приличат на процедурите** и функциите в обичайното процедурно програмиране.
- ❸ Всички **екземпляри** от определен клас от обекти са в състояние на **реагират на** набор от специфични за класа **събития**.

ФАЗА ПРОЕКТИРАНЕ

- ❶ В този етап **чрез свойство Име ((name))** се задават имената на формите и на носените от тях **компоненти на ГПИ**.
- ❷ **Две форми** в един и същи проект не могат да бъдат **с еднакви имена**.
- ❸ **Два елемента** на ГПИ, носени от една форма, също не могат да имат **еднакви имена**, освен когато **чрез различни числови стойности на свойство Index** формират **(едномерен)** масив.

ФАЗА ИЗПЪЛНЕНИЕ

- ❶ Една **форма** може да бъде въведена в **ОП** чрез:
 - изпълнение на **оператор Load**;
 - прилагане на **метод Show** към тази форма;
 - **достъп до** произволно **свойство** на тази форма;
 - **достъп до свойство на** носен от формата **елемент**.
- ❷ **Видимата (графична) част на форма се извежда** от ОП **чрез** оператор **Unload**.
- ❸ **Unload не извежда кода** на формата **от ОП!**
- ❹ **Чрез оператори Load и Unload един масив** от компоненти на ГПИ **може да промени броя** на своите елементи.
- ❺ **При извеждане на форма от ОП добавените с Load към нейните масиви елементи се губят!**

ОСНОВНИ СВОЙСТВА

Почти **всички класове** от обекти на ГПИ **имат важни едноименни свойства:**

- ❶ Име ((**name**)) и Индекс (**Index**).
- ❷ Видимост (**Visible**) – **True** или **False**.
- ❸ Достъпност = разрешеност (**Enabled**).
- ❹ Изглед (**Appearance**) – 2 D или 3 D.
- ❺ Шрифт, вкл. име, очертание и цвят (**Font**).
- ❻ Ред на клавиш Tab (**TabIndex** и **TabStop**).
- ❼ Местоположение (**Top, Height, Left, Width**).

Основна **мерна единица** на ВБ е единицата
Twip = 1/44 инча (≈ 0.577 милиметра).

ДРУГИ СВОЙСТВА

Други общи свойства са:

- Показалец на мишката (**MousePointer**);
- Потребителски показалец (**MouseIcon**);
- Потребителски белег (**Tag**);
- Подсещащ надпис (**ToolTipText**);
- Контекст при помощ (**WhatsThisHelpID**).

Често срещани свойства са:

- Цветове на текста и фона (**ForeColor** и **BackColor**);
- Надпис на елемента (**Caption** или **Text**);
- Подравняване (**Alignment**);
- Пренос и оразмеряване (**WordWrap** и **AutoSize**).

ОСНОВНИ СЪБИТИЯ

Почти **всички класове** от обекти на ГПИ **реагират** на следните **събития**:

- ❶ Щракване с мишката (**Click** и **DbClick**).
- ❷ Деактивиране (**LostFocus**).
- ❸ Активиране = фокусиране (**GotFocus**).
- ❹ Действия с мишката (**MouseDown**, **MouseUp**, **MouseMove**, **DragOver** и **DragDrop**).
- ❺ Действия с клавиатурата (**KeyDown**, **KeyPress** и **KeyUp**).

ВХОДНИ ЕЛЕМЕНТИ

Елементите на ГПИ, чрез които става въвеждане, имат специфични особености като:

- ① Събитие промяна (Change);**
- ② Свойство предизвиква проверка на предишния фокусиран елемент (CausesValidation);**
- ③ Събитие потвърждаване (Validate);**
- ④ Събитие нова позиция на плъзгача (Scroll) за елементи, които имат такъв.**

ЕДИНИЧЕН КОМПОНЕНТ И МАСИВ ОТ КОМПОНЕНТИ

Компоненти на ГПИ се разполагат върху формите **само по време на проектиране**. Единичните компоненти имат **уникални имена** (**<клас>№**), **празен низ** ("") като **Index** и:

- ❶ не могат да се премахват от формата по време на изпълнение;
- ❷ не могат да се добавят върху формата по време на изпълнение;
- ❸ всеки има **собствена събитийна процедура** **<име>_<събитие>**.

Записването на **число** в свойство **Index** формира **едномерен масив** от компоненти, който се характеризира с това, че:

- ❶ не се изисква индексите да бъдат **последователни цели числа**;
- ❷ по време на изпълнение с оператор **Load** в него може да **се добави**, а с **Unload** от него може да **се премахне елемент** с избран индекс;
- ❸ **всички компоненти** на ГПИ имат една **обща събитийна процедура**, с един допълнителен **параметър** – **индекс на елемента** на масива, **който** е регистрирал и **реагира на съответното събитие**.

СВОЙСТВА НА ФОРМА

Като главен елемент, съответстващ на понятието **прозорец** в ГПИ, **формите имат някои специфични свойства:**

- ❶ **Дъщерна форма (MDIChild)** – да/не;
- ❷ **Обработка на клавишите (KeyPreview)** – да/не;
- ❸ **Стил на рамката (BorderStyle);**
- ❹ **Елементи и място на прозореца (ControlBox, MaxButton, MinButton, Moveable, StartUpPosition, WindowState, ShowInTaskBar и др.);**
- ❺ **Икона (Icon);**
- ❻ **Управление на графичните методи [рисуването] (ClipControls, DrawMode, DrawStyle, DrawWidth, FillColor и FillStyle).**

СЪБИТИЯ НА ФОРМА

- ❶ Инициализация (**Initialize**) – създаване на нов екземпляр от конкретна форма;
- ❷ Завършване (**Terminate**) – премахване на последния указател към даден екземпляр;
- ❸ Въвеждане в ОП (**Load**);
- ❹ Заявка за извеждане (**QueryUnload**) – МДИ, деца с право на анулиране на искането;
- ❺ Извеждане от ОП (**Unload**) – деца, МДИ;
- ❻ Активиране на прозореца (**Activate**);
- ❼ Деактивиране на прозореца (**Deactivate**);
- ❽ Промяна на размера (**Resize**);
- ❾ Опресняване (**Refresh** и **Paint**).

РАБОТА С ФОРМИ

Константата **Me** е указател към екземпляра, за който се изпълнява кодът на формата.

Глобалната колекция **Forms** съдържа всички екземпляри, които в момента са в ОП.

Forms има единствено свойство Брой (**Count**), но **индексите са от 0 до Брой – 1!**

Нов екземпляр на форма се създава чрез използване на **New** при дефиниране или присвояване на обектова променлива.

РАБОТА С ФОРМИ (прод.)

Форма се въвежда в ОП с оператор **Load** <име> и при достъп до нейните компоненти.

Въвеждането в ОП на дъщерна форма автоматично въвежда и МДИ-формата.

Графичната част на форма, но не и нейният код, се извежда от паметта с оператор **Unload** <име>.

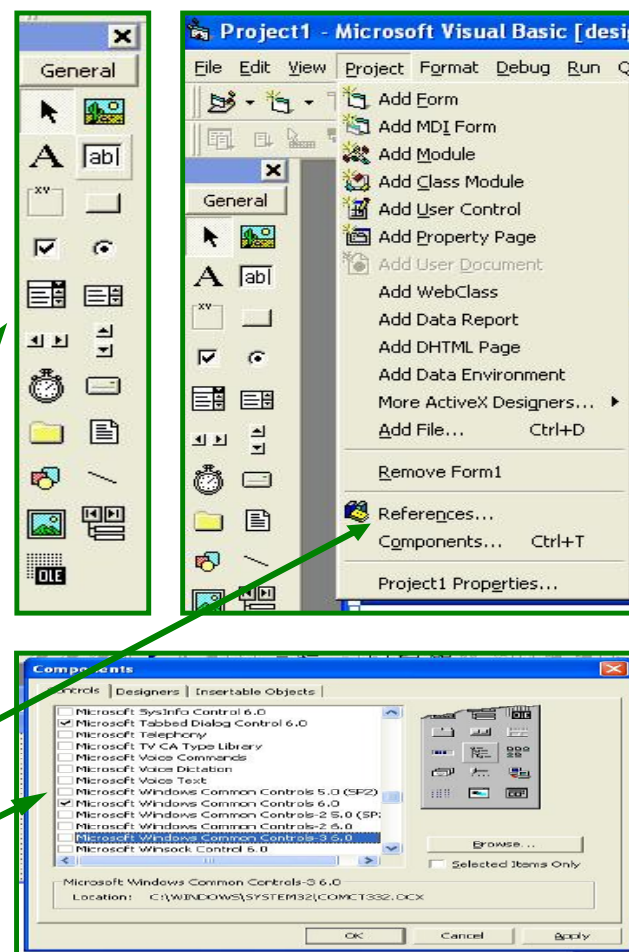
Формите се показват чрез метод **Show** [<стил>], където <стил> е **vbModal** (1) – последователно или **vbModeLess** (0) – паралелно (по подразбиране).

Записването на **True** в свойство **Visible** на форма е еквивалентно на прилагане на метод **Show**.

Форма се скрива чрез метод **Hide**, еквивалентен на записване на **False** в свойство **Visible**.

ОБЕКТИ НА ГПИ

Обектите на ГПИ се избират от палитрата с класове обекти на ВБ и се рисуват върху съответната форма. Палитрата съдържа само стандартните класове от обекти за ГПИ. В нея могат да бъдат добавяни както нови компоненти, така и класове, които са част от други програми.



ПРЕНОС НА ПРОГРАМА

Връзката клас – екземпляр се осъществява **от ОС** динамично. Това означава, че **за коректното изпълнение** на програма на **ВБ** **от даден компютър**, **в него трябва да присъстват и класовете от обекти**, използвани в тази програма, **и изпълнителната система** на **ВБ** – **VBRUNxx.DLL**.

За пренос на програми системата за проектиране предлага **помощник, който** след анализ на даден проект, **подготвя** дистрибутивни **дискети или CD**.

Класовете от обекти **са файлове с разширение** **VBX, OX** или **DLL**, създадени чрез **ВБ** или друг език. Те **могат да бъдат и част от** паралелно работеща **програма**, например **MS Word, MS Excel** и др.

ПРОСТИ ОБЕКТИ НА ГПИ

Етикет (**Label**) – за показване на надписи;



Текстова кутия (**TextBox**) – за вход
на всякакви текстови данни;



Команден бутон (**CommandButton**) – за
реализиране на действия (команди);



Рисунка (**PictureBox**), която играе и роля
на групиращ носител за други елементи.



Изображение (**Image**) – използва по-малко
системни ресурси и се рисува по-бързо.



ЕЛЕМЕНТ ЗА ИЗБОР



Контролната кутия (**CheckBox**) осигурява възможност за избор на независими помежду си възможности от типа ДА/НЕ. За задаване и определяне на положението се използва подразбираното свойство **Value** със стойности **vbUnchecked** (0), **vbChecked** (1) и **vbGrayed** (2).

Чрез свойство **Стил** (**Style**) и свойства **Picture**, **DownPicture** и **DisabledPicture** при желание избраният елемент може да бъде реализиран като „потънал“, вместо с белег.

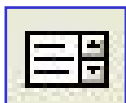
ЕЛЕМЕНТ ЗА ВЗАИМНО ИЗКЛЮЧВАЩ СЕ ИЗБОР



Изборният (радио) бутон (OptionButton**)**
дават възможност за избор на една
от няколко възможности, които са взаимно
изключващи се помежду си.

Няколко бутонa, които са разположени
върху един и същ носещ компонент –
форма (**Form**), рисунка (**PictureBox**) или
рамка (**Frame**), работят съвместно (**в група**).
Свойство **Value** е **True** или **False** и, както при
CheckBox, може да се реализира „потъване“.

ЕЛЕМЕНТИ ЗА ИЗБОР ОТ СПИСЪК



Списъчните (**ListBox**) и комбинираните (**ComboBox**) кутии осигуряват по-сложни възможности за избор.

Свойство **Стил (Style)** определя вида на елементите:

При **ComboBox** стойностите му са:

vbComboDropDown (0) – комбинация от **TextBox** за директен вход и разгъващ се списък за избор;

vbComboSample (1) – комбинация от **TextBox** за директен вход и видим списък за избор;

vbComboDropList (2) – избор от разгъващ се списък.

При **ListBox** стойностите са:

vbListBoxStandard (0) – показва се само текста;

vbListBoxCheckbox (1) – към текста има отметка.

ИЗБОР ОТ СПИСЪК (прод.)

Допълнителни свойства на списъците **са**:

Колони (**Columns**) [**LB**]: 0 – една с вертикален плъзгач,
 $n > 0$ – n колони с хоризонтален плъзгач;

Цели редове (**IntegralHeight**) – **True/False** (счпи);

Брой (**ListCount**) – недостъпно при проектиране (ндпп);

Списък (**List**) – масив от низове за избор (0 .. ListCount–1);

Данни (**ItemData**) – масив от **Long** (от 0 до ListCount–1);

Множествен избор (**MultiSelect**) [**станд. LB**]: **0** – забрана,
1 – прост (винаги), **2** – разширен (с Ctrl и Shift);

Сортиран (**Sort**) – ДА/НЕ (**True/False**);

Избран ред (**ListIndex**) – **-1** = няма такъв, 0 .. ListCount–1;

Индекс на новодобавения ред (**NewIndex**) – счпи, ндпп;

Индекс на първия видим ред (**TopIndex**) – ндпп.

МЕТОДИ НА СПИСЪЦИТЕ

Текстовете, които ще бъдат предложени за избор в списък, могат да бъдат записани в свойство **List** още при проектирането.

При изпълнение те могат да бъдат променени с прилагане на следните методи:

- ❶ добавяне: **AddItem** <низ>[, <индекс>];
- ❷ изтриване: **RemoveItem** <индекс>;
- ❸ нулиране: **Clear**.

Списъците се цитират чрез своето име или чрез обектова променлива, която сочи към конкретния екземпляр.

МЕНЮТА

Менютата са важен елемент на формите.
Те се проектират със специален редактор.
Допустима е йерархия до 4-то равнище.



Знак амперсанд (&) в заглавието на меню
определя следващата буква като бърз
клавиш за неговото активиране.

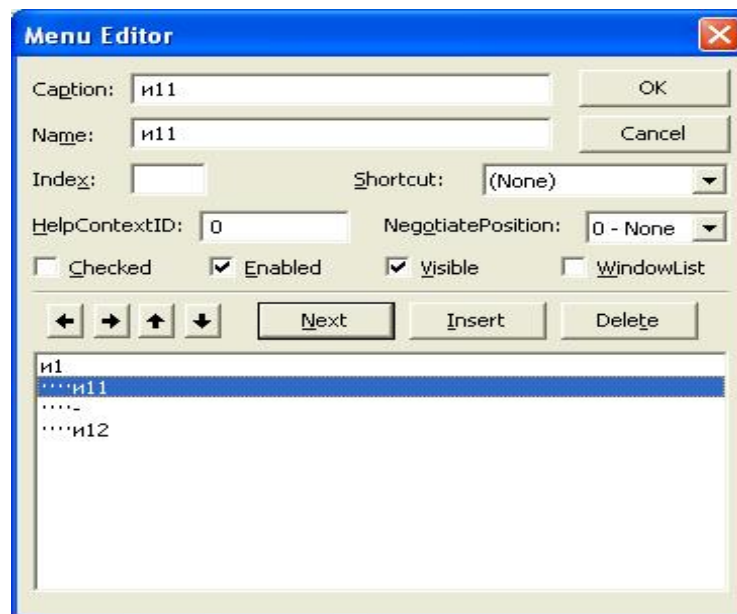
Знак минус (-) като заглавие единствено
на подменю реализира разделителна ивица.

Менютата разпознават само събитие **Click**.

Формите имат метод **PopupMenu**, чрез който
се показват контекстни менюта.

С оператори **Load** и **Unload** в масив от менюта
може да се добавят и изтриват елементи.

РЕДАКТОР НА МЕНЮТА



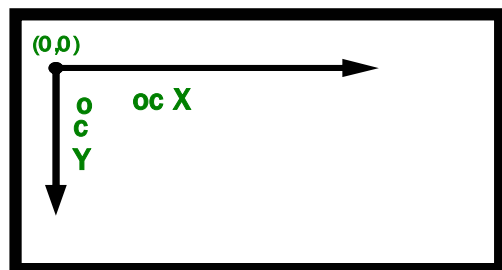
Редовете на контекстно меню се подготвят с редактора като подменюта на невидимо меню, което се показва с метод **PopupMenu** при събитие **MouseDown** или **MouseUp**.

ГРАФИКА И АНИМАЦИЯ

При проектирането на форма чрез елементи Отческа (**Line**) и Фигура (**Shape**) може да бъдат реализирани и графични компоненти.

Файлове с изображения се разполагат върху форма чрез елементите Рисунка (**PictureBox**) и Изображение (**Image**).

При реализиране на графика във формата се използва **лява координатна система**:



ВБ 3



– Line



– Shape

PictureBox:



Image:



26 / 32

ГРАФИЧНИ ЕЛЕМЕНТИ

Координати на определящите отсечката точки са свойствата **X1**, **Y1**, **X2** и **Y2**.

Свойство Вид (**Shape**) определя фигурата:

vbShapeRectangle (**0**) – правоъгълник;

vbShapeSquare (**1**) – квадрат;

vbShapeOval (**2**) – елипса;

vbShapeCircle (**3**) – окръжност;

vbShapeRoundedRectangle (**4**) – правоъгълник със заоблени ъгли;

vbShapeRoundedSquare (**5**) – заоблен квадрат.

ГРАФИЧНИ МЕТОДИ

Формите (**Form**) и рисунките (**PictureBox**) имат методи за чертаене при изпълнение:

- Line** – отсечка и правоъгълник (вкл. запълнен);
- Circle** – дъга, елипса (вкл. запълнена), окръжност и кръг;
- Print** – изписване на текст само във форма;
- Cls** – анулиране на изчертаните графика и текст;
- PSet** – задава цвят на посочената точка;
- Point** – връща цвета на посочената точка;
- PaintPicture** – показване на файл с изображение.

С графичните методи са свързани свойства:

- CurrentX, CurrentY** – текущи координати;
- FillStyle** и **FillColor** – стил и цвят при чертаене;
- DrawWidth** – дебелина на линията;
- BackColor** – цвят на фона.

ПРОСТА АНИМАЦИЯ

Проста анимация се постигна по 3 начина:

- ❶ Създаване на невидими елементи **Image**, в които при проектиране в свойство Рисунок (**Picture**) се подготвят изображенията. При изпълнението тези свойства се копират в същото свойство на друг видим елемент;
- ❷ Промяна на мястото (**Top, Left**) на видим елемент;
- ❸ Показване и скриване (**Visible**) на елементи.

И в трите случая е важно да се измерва време.

За целта се ползва елемент Хронометър (**Timer**).



Този елемент е без видимо изображение и има свойство Интервал (**Interval**), определящо през колко милисекунди ще се генерира събитието **Timer** на разрешен (**Enabled = True**) хронометър.

ДОПЪЛНИТЕЛНИ ВЪЗМОЖНОСТИ НА ВБ

Стандартна функция **MsgBox** показва в свой прозорец кратко съобщение, желаните познати икони (**?**, **!**, **I**, **STOP**) и бутони (**OK**, **YES**, **CANCEL**, **NO**, **RETRY**, **IGNORE**). Тя връща сведения за избрания от потребителя бутон.

Стандартна функция **InputBox** показва в свой прозорец съобщение, текстова кутия с подразбиран текст и бутони (**OK** и **Cancel**), и връща въведения от потребителя текст.

СТАНДАРТНИ ДИАЛОЗИ

Познатите диалози за избор на шрифт, цвят, печатащо устройство и име на файл за четене или запис се реализират с прилагане на методи към елемент за **Общ диалог (CommonDialog)**.

Елементът **няма видимо изображение** във формата, **но прилагането на метод показва** отделен **прозорец**.

Методите, използващи различни свойствата, **са:**

ShowOpen – диалог за име на входен файл;

ShowSave – диалог за име на изходен файл;

ShowFont – диалог за избор на шрифт;

ShowColor – диалог за избор на цвят;

ShowPrinter – диалог за избор на печатащо устройство.

Елементът **не е част от стандартната палитра** с елементи на ГПИ и трябва да се добави към нея.



**БЛАГОДАРЯ ВИ
ЗА ВНИМАНИЕТО!**

**БЪДЕТЕ С МЕН И
В СЛЕДВАЩАТА ЛЕКЦИЯ,
КОЯТО ЩЕ НИ ОТВЕДЕ
В НЕВЕРОЯТНИЯ СВЯТ НА
ВЗАИМОДЕЙСТВИЕТО
С ОС WINDOWS**