








ЛЕКЦИЯ 4

ВЗАИМОДЕЙСТВИЕ С ОС

-  **Обект Приложение (App)**
-  **Обект Екран (Screen)**
-  **Обект Контейнер (Clipboard)**
-  **Обект Принтер (Printer)**
-  **Полезни възможности на ВБ**
-  **Обработка на грешки**
-  **Работа с дискови файлове**

ПРИЛОЖЕНИЕ (App)

Представя изпълняваната програма.
Свойствата само се четат и по-важни са:

- EXENAME** – име на програмата/проекта;
- Path** – път към програмата;
- Title** – заглавен низ на програмата;
- PrevInstance** – изпълнява ли се паралелно друго копие на програмата (True/False);
- Major** и **Minor** – версия на програмата;
- Revision** – реализация на програмата;
- ThreadID** – идентификатор, използваем в процедури и функции от ядрото на ОС.

ЕКРАН (Screen)

Представя екрана на монитора:

ActiveForm – указател към активната форма;

ActiveControl – указател към елемента на ГПИ, който е фокусиран (**и при форми**);

FontCount – брой на достъпните шрифтове;

Fonts – масив (**от 0**) с имена на шрифтовете;

TwipsPerPixelX , **TwipsPerPixelY** – размер на пиксел от екрана **в twips**.

MousePointer – код на показалеца;

MouseIcon – потребителски показалец (**99**);

Height, Width – размери на екрана **в twips**.

КОНТЕЙНЕР (Clipboard)

Няма свойства, а само методи за достъп:

Clear – изтрива съдържанието;

GetFormat(<формат>) – показва запомнени ли са данни с посочения формат (**Да/Не**);

GetData(<формат>) – прочита графичното изображение с посочения формат;

GetText(<формат>) – прочита текст;

SetData <графика>[, <формат>] – записва графично изображение;

SetText <низ>[, <формат>] – записва текст.

ФОРМАТ НА ДАННИТЕ

Системният контейнер съхранява едновременно по един комплект данни от следните формати:

vbCFBitmap (2) – битова карта (**.bmp**);

vbCFMetafile (3) – метафайл (**.wmf**);

vbCFDIB (8) – независима битова карта (**DIB**);

vbCFPalette (9) – цветова палитра;

vbCFLink (&HBF00) – информация за обмен;

vbCFRTF (&HBF01) – текст от вида **Rich Text**;

vbCFText (1) – обикновен текст (подразбиран).

ПРИНТЕР (Printer)

Във ВБ **Printer** означава две неща:

- ❶ **име на клас**, описващ характеристиките на драйвер на печатащо устройство;
- ❷ **име на обектова променлива**, която е указател към екземпляр от горния клас, описващ принтера, с който работи програмата (първоначално системния за ОС).

Всички налични принтери са членове на колекция **Printers** със свойство **Брой (Count)** и **индекси от 0**.

Чрез обектовата променлива и колекцията от принтери, **текущият принтер** на програмата и **неговите свойства** могат да бъдат **променени**.

Свойствата на **принтерите в колекцията са само за четене** и не могат да се променят.

СВОЙСТВА НА ПРИНТЕР

По-интересни свойства на принтерите са:

Page – номер на текущата страница.

Copies – брой отпечатвани копия.

DriverName – име на драйвера.

DeviceName – име (модел) на устройството.

Zoom – мащабиране на печата в проценти.

ColorMode – как ще работи цветен принтер:

VbPRCMMonochrome (1) – монохромно (сиво);

VbPRCMColor (2) – пълна цветова гама.

TrackDefault – мени ли се променлива **Printer** при корекция на системния принтер (**Да/Не**).

ОТПЕЧАТВАНЕ НА ТЕКСТ

Свойства при отпечатване на текст:

FontName – име на използвания шрифт.

FontSize – размер на шрифта (в пункта).

Font – очертания на използвания шрифт:

.**FontBold** – получерно (Да/Не);

.**FontItalic** – наклонено (Да/Не);

.**FontStrikethru** – зачертано (Да/Не);

.**FontUnderline** – подчертано (Да/Не);

FontCount – брой достъпни шрифтове.

Fonts – масив с имената на достъпните за принтера шрифтове.

ИЗБОР НА ХАРТИЯ

Свойствата за хартиения лист са:

PaperSize – код за размер на листа: A4, Letter.

Height и **Width** – физ. размери на листа в **twips**.

PaperBin – подразбиран източник на хартия.

Orientation – ориентация на печата:

vbPRORPortrait (1) – портретна;

vbPRORLandscape (2) – пейзажна.

Duplex – управление на двустранно печатане:

VbPRDPSimplex (1) – едностранно;

VbPRDPHorizontal (2) – двустранно по хоризонтал;

VbPRDPVertical (3) – двустранно по вертикала.

ПЕЧАТ НА ГРАФИКА

Свойствата за отпечатване на графика са:

CurrentX, CurrentY – текущи координати при методите за печат на графика.

DrawMode, DrawStyle, DrawWidth – параметри на графичните методи.

FillColor и **FillStyle** – цвят и стил на запълване.

PrintQuality – качество на отпечатайка:

vbPRPQDraft (-1) – чернова;

vbPRPQLow (-2) – ниска разрешаваща способност;

vbPRPQMedium (-3) – средна разрешаваща способност;

vbPRPQHigh (-4) – висока разрешаваща способност;

$n > 0$ – n точки на инч (dpi).

МЕТОДИ НА ПРИНТЕР

Отпечатването на текст и графика става
чрез следните методи:

EndDoc – край на отпечатването на документ.

KillDoc – анулиране на отпечатването.

NewPage – нова страница (увеличава **Page**).

TextHeight(<низ>) – изчислява височината;

TextWidth(<низ>) – изчислява широчината;

Print – отпечатване на текст.

Circle, Line, PaintPicture, PSet – отпечатване
на графики, рисунки и единични точки.

ВЪЗМОЖНОСТИ НА ВБ

Някои **полезни оператори** на ВБ са:

Beep – генерира звуков сигнал.

SendKeys <низ>[,<изчакване>] – имитира въвеждането на низ от клавиатурата.

DoEvents – освобождава ЦП.

AppActivate <заглавие>[,<изчакване>] – активира програма по заглавие на нейния прозорец.

Някои **полезни функции** на ВБ са:

Shell(<път>[,<прозорец>]) – стартира изпълнението на програма от посочен файл.

Format(<израз>[,<фмт>[,...]]) – преобразува в текст стойността на израза по зададен формат.

ОБРАБОТКА НА ГРЕШКИ

Всяка процедура може да определи **как да бъдат обработени** възникващите при нейното изпълнение **грешки**.

Това става с изпълнение на операторите:

On Error GoTo <етикет> – назначаване на участък за обработка на грешки, който започва от оператора с <етикет>.

On Error Resume Next – игнориране на грешките, които възникват.

On Error GoTo 0 – отмяна на назначения участък за обработка на грешки.

РЕАКЦИЯ ПРИ ГРЕШКА

При възникване на грешка стандартната реакция на ВБ е следната:

- ① Сведенията за възникналата грешка се записват в свойствата на системния обект Err.**
- ② В списъка с активирани процедури от текущата назад се търси такава, за която е назначен участък за обработка на грешки или грешките са забранени.**
- ③ При откриване на назначен участък за грешки той се активира чрез изпълнение на обявения оператор.**
- ④ При забрана на грешките управлението се предава на оператора, следващ този, който е породил грешката.**
- ⑤ В останалите случаи се показва подходящо съобщение и изпълнението на програмата се спира без Unload.**
- ⑥ Грешките в активен участък за обработка се обслужват от по-ранното процедурно равнище.**

КРАЙ НА ОБРАБОТКАТА

Активиран участък за обработка на грешки **завършва** активността си **чрез оператори**:

Resume [0] – изпълнението на оператора, породил грешката, се повтаря.

Resume Next – изпълнението продължава с оператора, следващ породилия грешка.

Resume <етикет> – изпълнението продължава с посочения чрез <етикет> оператор в същата процедура.

Изпълнението на оператор Resume, вън от активиран участък за обработка на грешки, е забранено и поражда грешка.

СВЕДЕНИЯ ЗА ГРЕШКИ

Свойства на стандартния обект **Err**:

Number – специфичен номер на грешката (подразбирано свойство на обекта).

Description – текстово описание.

LastDLLError – последна грешка, попълнена при изпълнение на подпрограма от DLL.

Source – име на клас (приложение), който е генерирал (открил) грешката.

Стандартна функция **Error**[(**<номер>**)] връща текста за грешката с посочения номер.

Стандартна функция **Err** (**VBA**) връща номера на реда, породил грешката.

ОЩЕ ЗА ГРЕШКИТЕ

Обект **Err** има два метода:

- ① **Clear** – явно нулиране на свойствата му.
- ② **Raise** <номер> [,<източник> [,<текст>]] – генерира грешка с посочения номер.

Забележки:

- ① Метод **Clear** се прилага неявно към **Err** при изпълнение на оператори **Resume**, **On Error**, **Exit/End Sub/Function** и при вход в процедура/функция.
- ② Оператор **Error** <номер> е еквивалентен на прилагане на метод **Raise** <номер>.

ДИСКОВИ ФАЙЛОВЕ

За дълготрайно съхранение на данни ВБ осигурява възможности за работа с **дискови файлове от два типа**: текстови и двоични.

Преди да се **обменят данни** с дисков файл той трябва **да бъде открит (отворен)** за работа чрез посочване на неговото име.

След приключване на работата с дисков файл той трябва **да бъде закрит (затворен)**.

Дисковите файлове се цитират с **целочислен идентификационен номер**, избран от програмиста и представян чрез числов израз, чиято стойност се окръглява до цяло.

ОТВАРЯНЕ НА ФАЙЛ

Open <път> **For** [<режим>] [**Access** <достъп>]
[<блокиране>] **As** [#]<номер>
[**Len**=<дължина>]:

- ❶ <път> задава името на файла;
- ❷ <режим> определя начина на работа: **Append**, **Binary**, **Input**, **Output** или **Random**;
- ❸ <достъп> определя разрешените операции: **Read**, **Write** или **Read Write**;
- ❹ <блокиране> определя колективния достъп: **Shared**, **Lock Read**, **Lock Write** или **Lock Read Write**;
- ❺ <номер> е идентификатор на отворения файл;
- ❻ <дължина> е размер на буфера в ОП (без **Binary**).

Файловете, с режимите **Append** и **Output**, не допускат повторно отваряне с друг идентификатор.

ЧЕТЕНЕ ОТ ФАЙЛ

Line Input #<номер>, <променлива>
прочита един ред до CR (13) [+ LF (10)].

Input #<номер>, <списък от променливи>
прочита специално оформени чрез , " и #
стойности за посочените променливи.

Функция **Input**(<брой знаци>, [#]<номер>)
чете посочения брой знаци, вкл. , " # и CR.

Функция **InputB**(<брой байта>, [#]<номер>)
чете посочения брой байтове.

Get [#]<номер>, [<начало>], <променлива>
прочита данни във вътрешен (ОП) формат.

ЗАПИС ВЪВ ФАЙЛ

Put [#]<номер>, [<начало>], <променлива>
запис на данни без преобразуване на ОП .

Print #<номер>, [<списък>] извежда
посоченото в списъка, който съдържа
последователности от вида:
[**Spc**(*n*) | **Tab**(*n*)] [<израз>] [, | ;] .

Write #номер, [списък от изрази] извежда
стойностите на изразите чрез специално
форматиране със , " и #.

Seek(<номер>) връща позицията във файла.

Seek [#]<номер>, <позиция> позиционира.

ДРУГИ ВЪЗМОЖНОСТИ

Close [[#]<номер> [, [#]<номер>]...]

затваря посочените номера или всички отворени за работа файлове.

Reset затваря всички отворени файлове.

EOF(<номер>) връща истина при достигане на края на посочения файл.

LOF(<номер>) връща броя на байтовете в отворен файл.

FileLen(<път>) връща броя на байтовете (евентуално до отварянето на файла).

РЕГИСТРАЦИЯ

За съхранение на специфични данни между две изпълнения ОС Windows предоставя специална регистратура – **Windows Registry**. Стойностите за регистрираните чрез програми на ВБ данни се съхраняват чрез елементите **HKEY_CURRENT_USER\Software\VB and VBA Program Settings\програма\секция\ключ** .

За работа с регистратурата на Windows ВБ предоставя следните оператори и функции:

- SaveSetting** програма, секция, ключ, стойност ;
- DeleteSetting** програма, секция [, ключ] ;
- GetSetting**(програма, секция, ключ [, липса]) ;
- GetAllSettings**(програма, секция) – масив $n \times 2$.

**БЛАГОДАРЯ ВИ
ЗА ВНИМАНИЕТО!**

**БЪДЕТЕ С МЕН И
В ПОСЛЕДНАТА ЛЕКЦИЯ,
КОЯТО ЩЕ НИ ОТВЕДЕ
В НЕВЕРОЯТНИЯ СВЯТ НА
ИЗПОЛЗВАНЕТО
НА БАЗИ ОТ ДАННИ**