

ИЗПОЛЗВАНЕ НА VISUAL BASIC FOR APPLICATIONS ПРИ РЕШАВАНЕТО НА ПРАКТИЧЕСКИ ЗАДАЧИ В ОБУЧЕНИЕТО ПО ИНФОРМАЦИОННИ ТЕХНОЛОГИИ

Асен Рахнев, Коста Гъров, Стефка Анева

Резюме. Настоящата разработка е посветена на използването на Visual Basic for Applications при изучаване на модула „Интегриране на дейности и документи“ в профилираната подготовка по информационни технологии в горните класове на средното училище. Разглежда се практическа задача, в която са реализирани основните етапи на приложение на Visual Basic for Applications в MS Excel. Материалът е предназначен за учители по информационни технологии, преподаващи в 11. и 12. клас на СОУ.

Keywords: Information Technology, training, Visual Basic for Applications, macro.

I. Използване на Visual Basic for Applications в профилираното обучение по информационни технологии в средното училище

В учебната програма по информационни технологии (ИТ) за профилирана подготовка в 11. клас е включен за изучаване модула „Интегриране на дейности и документи“ в рамките на 100 учебни часа¹. В тази програма се изисква учениците да се запознаят с предназначението на вградените средства за програмиране, чрез които могат да бъдат разширени функционалните възможности на потребителските продукти с общо предназначение, автоматизирани редица дейности по обработка на документи и опростени процедурите за манипулиране с данни във всички приложения на MS Office. В резултат от обучението, учениците ще разберат и осмислят предназначението и общата структура на макродефинициите, ще формират базови познания за обектните модели на отделните приложения от MS Office и практически, и приложни умения за създаване и редактиране на кратки програми, инсталиране и изтриване на макродефиниции.

Поради недостига на учебни помагала и методически разработки за организация на учебния процес и методика за провеждане на профилирано обучение по информационни технологии за темата „Разширяване на възможностите на потребителски продукти с общо предназначение чрез вградените средства за програмиране“, тук предлагаме една реализация на поставените цели с използване на **Visual Basic for**

Applications (VBA). Други подобни разработки са представени в (Гъров & Анева, 2006), (Рахнев, Гъров & Анева, 2012), (Роман, 2004) и (Роман, 1999).

VBA е обектно ориентиран език, базиран на Microsoft Visual Basic и е мощен инструмент за разработка на приложения на базата на програмния пакет Microsoft Office. В помощ на разработчиците са не само основните конструкции на езика, но и мощна обектно-ориентирана среда. Чрез използване на обектите на VBA може в значителна степен да се опростят процедурите за манипулиране с данни и документи.

Комплектът Microsoft Office съдържа повече от дванадесет различни обектни модела за отделните Office приложения и отнасящите се към тях продукти. Това включва обектни модели за Word, Excel, Access, DAO (Data Access Objects), Outlook, PowerPoint, Binder, Graph, Forms, VBA, Visual Basic, ASP (Active Server Pages) и др.

Всяко главно приложение на VBA (Word, Excel, Access, PowerPoint и др.) допълва основния VBA език, като му предоставя обектен модел за работа с обектите, специфични за съответното приложение.

Например, VBA за Excel включва обектния модел Excel, който работи с такива обекти като работни книги, работни листове, клетки, редове, колони, области, таблици, осев таблици и др.

VBA за Word включва обектния модел Word, който работи с такива обекти като документи, шаблони, параграфи, шрифтове, колонтитули, таблици и др.

VBA на Access включва два обектни модела – Access обектен модел и DAO обектен модел, които позволяват на програмистите да работят с такива обекти като таблици с данни, заявки, форми и отчети.

С помощта на VBA могат да се създават **кратки описателни програми**, с които могат да се управляват **автоматично** често повтарящи се действия, манипулации с обекти и действия, ръчното изпълнение на които предизвиква известна сложност. В практиката често се използва и понятието **макрос**, което може да се дефинира като **именувана** последователност от команди и инструкции, групирани като **една команда**, с цел автоматично изпълнение. Но същевременно VBA е и много мощен език за програмиране, с който могат да се създават реални програми във всички приложения на MS Office, благодарение на които ще се автоматизира решението и изпълнението на много сложни задачи от реалния живот и ежедневие.

Със следващата практическа задача, ще покажем приложенията на VBA в MS Office.

II. Практическа задача, реализираща връзката между Visual Basic for Applications и MS Excel

Постановка на задачата: В работен лист от електронна таблица е проектирана база от данни, която организира и съхранява необходимата информация, поддър-

жаща регистър за любителски риболовни билети. В последователни колони на работния лист е съхранена информация за риболовния билет, данни за лицето, на което се издава билета и данни за негов законен представител, в случаите когато лицето е под 14-годишна възраст.

-Данните за риболовен билет включват следната информация (*Таблица № 1*):

Данни за риболовен билет				
Тип ползвател на билета	Тип билет	Номер на билет	Дата на издаване	Орган издал билета

Таблица № 1

- Възможните стойности за полето *Тип ползвател на билета* са:
 - платен;
 - за деца (под 14 години);
 - навършили 65 г. мъже;
 - навършили 60 г. жени;
 - инвалиди.
 - Възможните стойности за полето *Тип билет* са: годишен; полугодие; месечен и седмичен.

Данните за лицето, на което се издава риболовен билет включват следната информация (*Таблица № 2*):

Данни на лицето		
име	ЕГН	адрес

Таблица № 2

Данните на законния представител на лицето, в случаите на издаване на безплатен риболовен билет на лице под 14-годишна възраст включват следната информация (*Таблица № 3*):

в случаите на издаване на безплатен билет на лице под 14-годишна възраст				
данни на законния представител на лицето			лична карта на законния представител	
име	ЕГН	адрес	номер	дата на издаване

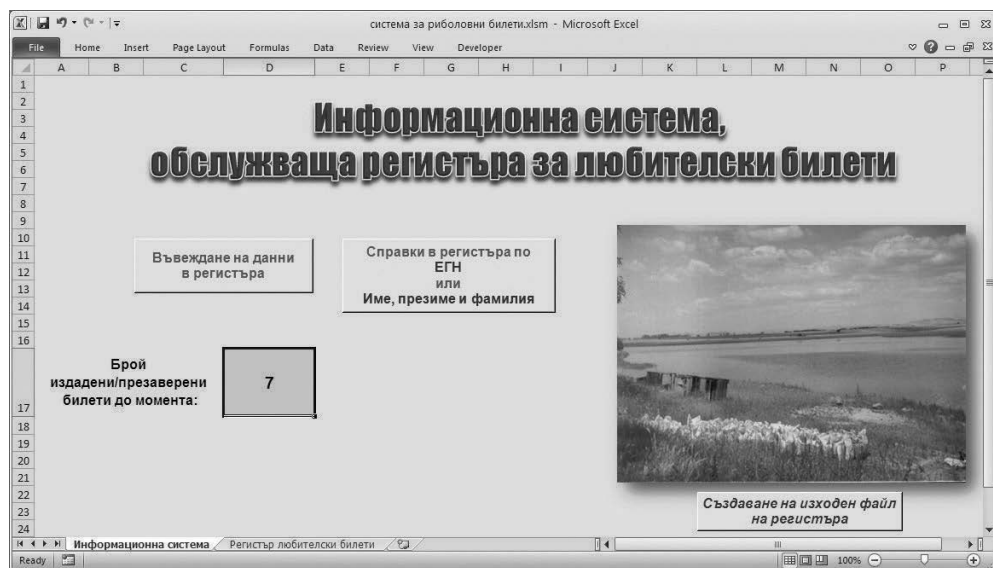
Таблица № 3

С помощта на MS Excel и VBA да се създаде информационна система (*фиг. 1*) за автоматизирано обслужване на регистъра за риболовни билети, с цел улесня-

ване на дейностите по въвеждането на информация, актуализиране и извършване на справки в него. Тъй като обема на въвежданата информация е от порядъка на няколко хиляди човека, то се налага извършване на справки и в стари регистри от минали години в случаи на презаверка на риболовни билети, като по този начин се улеснява значително дейността по въвеждане на информацията в новия регистър.

Системата трябва да предоставя следните функционални възможности:

- Въвеждане на нови данни в регистъра на любителските билети;
- Справка по ЕГН в стария регистър на любителски билети от миналата година при въвеждане на нови данни с цел по-бързо и автоматизирано попълване на необходимата информация за лицето, на което се издава билет;
- Текуща справка за введените данни в регистъра по ЕГН или Име, презиме и фамилия;
- Актуализация на вече попълнени данни;
- Създаване на външен файл във вид на електронна таблица, съдържащ само актуалният регистър с попълнени данни за любителските билети.



Фигура 1

Реализация на задачата:

Работната книга ще се състои от 3 работни листа: *Регистър любителски билети*, *Информационна система* и *Данни от минали години*.

В средата на Visual Basic Editor ще именуваме тези работни листове съответно с имена **reg**, **main** и **danni**. Тъй като листа **danni** ще съдържа подробна информация

от регистъра с любителски билети от минали години е удачно той да бъде скрит и да не може да се осъществява директен достъп до него в средата на MS Excel. За да се реализира това е необходимо в средата на Visual Basic Editor да се зададе стойност 2-xlSheetVeryHidden на свойство Visible в прозореца Properties Window.

В клетка D17 от листа „Информационна система“ трябва да се визуализира общият брой на въведените в регистъра данни за риболовни билети до момента с помощта на следната функция: MAX(‘Регистър любителски билети’!A:A).

За реализиране на описаните по-горе дейности по обслужване на регистъра за любителски билети с помощта на Visual Basic for Applications трябва да бъдат автоматизирани следните операции:

- Въвеждането на нови данни в работния лист *Регистър любителски билети* ще се осъществява с помощта на потребителска форма. За целта трябва да бъдат създадени процедурни макроси за визуализиране на формата за въвеждане на данни и съхраняване на въведената информация в работния лист *Регистър любителски билети*. Макроса за визуализиране на формата за въвеждане ще се стартира чрез бутон, разположен в работния лист *Информационна система*;
- За улеснение при попълването на данни във формата за въвеждане в случай на презаверка през текущата година на вече издаден риболовен билет, е необходимо да бъде създаден процедурен макрос за организиране на търсене на данни в стария регистър (работен лист *Данни от минали години*). Този макрос ще се стартира чрез команден бутон, разположен във формата за въвеждане на данни, като преди това е необходимо да се въведе ЕГН в съответното текстово поле на формата за въвеждане на данни. В резултат от изпълнението на макроса при намерени данни, необходимите данни за лицето, за което ще се презаверява билет ще бъдат автоматично попълнени в съответните полета на формата;
- Изборът на критерий за търсене (ЕГН или Име, презиме и фамилия) при осъществяване на справка за въведени данни в регистъра ще се реализира с помощта на потребителска форма. В нея ще бъдат въведени и необходимите данни за търсене. В случая, трябва да бъдат създадени процедурни макроси за визуализиране на формата, за реализиране на търсенето и представяне на получения резултат от търсенето. Процедурният макрос за визуализиране на формата за търсене ще се стартира чрез бутон, разположен в работния лист *Информационна система*. Процедурният макрос за реализиране на търсенето ще се стартира с помощта на команден бутон, разположен върху формата за избор на критерий и въвеждане на данни за търсене. Крайният резултат от справката за намерения запис в регистъра при търсенето ще се визуализира в потребителска форма;
- За актуализация на въведени данни в регистъра ще се използват отново формата за търсене на запис по избран критерий и макросите за визуализиране на формата

- за търсене, за реализиране на търсенето и представяне на получения резултат във форма за справка и актуализация на данните. Трябва да бъде създаден процедурен макрос, с помощта на който ще се осъществява актуализацията на данните за даден запис в регистъра. Този макрос ще се стартира с помощта на команден бутон, разположен във формата за справка за въведени данни, която се визуализира при представяне на крайния резултат от търсенето;
- При създаването на нов файл във вид на електронна таблица, съдържащ единствено данните от работен лист *Регистър любителски билети* е необходимо да бъде създаден процедурен макрос. Той ще се стартира чрез бутон, разположен в работния лист *Информационна система*.

Създаване на потребителските форми за информационната система

Към информационната система с помощта на Visual Basic for Applications трябва да бъдат създадени следните форми:

1. Форма за въвеждане на нови данни в регистъра (frmadd) – данни за риболовния билет и лични данни на лицето:

Фигура 2

Формата съдържа следните елементи на графичния потребителски интерфейс (ГПИ) (Фиг. 2): две рамки, етикети за визуализиране на текст върху формата, две групи избираеми бутони за определяне на ползвател и тип на билета, седем текстови полета и три командни бутона. В режим на проектиране към двете групи избираеми бутони трябва да бъдат извършени някои допълнителни настройки, определящи начални стойности по подразбиране за типа и ползвателя на билета.

2. Форма за въвеждане на лични данни за законен представител, в случаите, когато се издава риболовен билет на деца под 14 години (frmpredstavitel).

Данни на законния представител

Лични данни на законния представител

Име, презиме и фамилия

ЕГН

Адрес

Лична карта №

Дата на издаване на документа

Съхрани данните на представителя

Фигура 3

Формата съдържа следните елементи на ГПИ (Фиг. 3): една рамка, етикети за визуализиране на текст върху формата, пет текстови полета и един команден бутон.

Забележка:

В случаите при извършване на справки в регистъра за дете (под 14 години), на което е издаден риболовен билет ще се използва същата форма с някои малки корекции за визуализиране и актуализация на вече въведените данни за законен представител, т.е. формата ще изглежда по следния начин (Фиг. 4):

Данни на законния представител

Лични данни на законния представител

Име, презиме и фамилия

ЕГН

Адрес

Лична карта №

Дата на издаване на документа

Актуализация на данните на представителя

Фигура 4

3. Форма за търсене в регистъра по ЕГН или по Име, презиме и фамилия (frmspravka).

Справка

Търсене по ЕГН

Търсене по име

☒ Търсене по ЕГН

☐ Търсене по име

Търсене

Фигура 5

Формата съдържа следните елементи на ГПИ (Фиг. 5): етикети за визуализиране на текст върху формата, два избираеми бутона, две текстови полета и един команден бутон. В режим на проектиране към елементите на ГПИ върху формата трябва да бъдат извършени някои допълнителни настройки, така че при визуализиране на формата да бъде избран критерий за търсене по ЕГН. Съответното текстово поле за въвеждане на данни за търсене по ЕГН трябва да бъде разрешено, а второто текстово поле – не.

4. Форма за визуализация на въведената информация за търсен запис в регистъра и актуализация на данните (fmrshow)

Справка за въведените данни

Пореден номер на вписване: 4

Тип ползвател на билета:

- ☐ 1. Платен
- ☒ 2. За деца (под 14 г.)
- ☐ 3. Навършили 65 г. мъже
- ☐ 4. Навършили 60 г. жени
- ☐ 5. Инвалиди

Тип билет:

- ☒ 1. Годишен
- ☐ 2. Полугодие
- ☐ 3. Месечен
- ☐ 4. Седмичен

Данни за билета

Номер на билета:

Дата на издаване или презаверяване на билета:

Лични данни на лицето

Име, презиме и фамилия:

Адрес:

Орган, издал билета:

ЕГН:

Покажи данните на законния представител

Актуализация

Изход

Фигура 6

Формата съдържа следните елементи на ГПИ (Фиг. 6): две рамки, етикети за визуализиране на текст върху формата, две групи избираеми бутона за определяне на ползвател и тип на билета, седем текстови полета и три командни бутона. В режим на проектиране към елементите на ГПИ върху формата трябва да бъдат извършени някои допълнителни настройки, така че при визуализиране на формата за справка бутонът „Покажи данните на законния представител“ да се визуализира само в случай, че ползвателя на билета е лице под 14 годишна възраст.

Създаване на процедурните макроси към информационната система
Към информационната система да се реализират следните макроси:

Задача 1: Да се създаде процедурен макрос с име add, който визуализира формата за въвеждане на данни.

```
Public Sub add()  
    Dim m As Integer  
    frmadd.txtnomer.Text = main.Range("D17").Value + 1  
    frmadd.txttime.Text = ""  
    frmadd.txttegn.Text = ""  
    frmadd.txtadres.Text = ""  
    frmadd.optgod.Value = True  
    frmadd.optplaten.Value = True  
    frmadd.txtnumbilet.Text = ""  
    Dim today As Variant  
    today = Now  
    frmadd.txtdata.Text = Format(today, "dd.mm.yyyy г.")  
    frmadd.txtorgan.Text = ""  
    frmadd.Show  
End Sub
```

В работният лист „*Информационна система*“ да се създаде команден бутон с име „Въвеждане на данни в регистъра“ и да се свърже с макроса **add**.

Задача 2: Да се създаде процедурен макрос с име spravka, който визуализира формата за търсене, като по подразбиране е избрано търсенето да се извършва по ЕГН.

```
Public Sub spravka()  
    frmspravka.OptionButton1.Value = True  
    frmspravka.Show  
End Sub
```

В работният лист „*Информационна система*“ да се създаде команден бутон с име „Справки в регистъра по ЕГН или Име, презиме и фамилия“ и да се свърже с макроса **spravka**.

Задача 3: Да се създаде процедурен макрос с име create_newbook, който създава нова работна книга – изходен файл, съдържащ само работния лист „Регистър любителски билети“.

```
Public Sub create_newbook()  
    Dim varpath As String  
    If MsgBox("Желаете ли да създадете външен файл, съдържащ  
    въведената информация в регистъра?", vbYesNo) = vbYes
```

```
Then
    reg.Copy
    varpath = ThisWorkbook.Path
    ActiveWorkbook.SaveAs varpath & "/registyr.xlsx"
    ActiveWorkbook.Close
End If
End Sub
```

В работният лист „*Информационна система*“ да се създаде команден бутон с име „Създаване на изходен файл на регистъра“ и да се свърже с макроса **create_newbook**.

Задача 4: Да се създаде процедурен макрос с име **spravka_oldreg**, който по въведен ЕГН във формата за въвеждане на нови данни извършва търсене в листа „Данни от минали години“. При намерен запис в стария регистър във формата за въвеждане на нови данни в съответните полета се попълва автоматично името и адреса на лицето и номера на издадения билет на търсения човек.

```
Public Sub spravka_oldreg()
    Dim search As String
    Dim num, i As Integer
    Dim flag As Boolean
    search = frmadd.txtegn.Text
    Dim m As Double
    m = Val(search)
    flag = False
    If search <> "" Then
        If (m <= 1000000000) Or (m >= 9999999999#) Then
            MsgBox „Неправилно въведено ЕГН!“
            Exit Sub
        Else
            danni.Range(„P1“).Formula = „=MAX(,Данни от минали години`!A:A)“
            num = danni.Range(„P1“).Value
            For i = 1 To num
                pom = danni.Cells(3 + i, 8).Value
                If pom = search Then
                    flag = True
                    frmadd.txttime.Text = danni.Cells(3 + i, 7).Value
                    frmadd.txtadres.Text = danni.Cells(3 + i, 9).Value
                End If
            Next i
        End If
    End If
End Sub
```

```
        frmadd.txtnumbilet.Text = dann1.Cells(3 + i,
4).Value
        main.Select
    End If
Next i
    If flag = False Then MsgBox „Няма данни в стария
регистър!“
End If
Else
    MsgBox „Въведете ЕГН и след това натиснете бутона
Справка!“
End If
main.Select
End Sub
```

Задача 5: Да се създаде процедурен макрос с име save, който при въведени коректни данни във формата за въвеждане съхранява въведената информация в регистъра. Проверките за коректност включват: ЕГН – 10 цифри, номер на билет - числова стойност, не се допуска запис в регистъра при празни полета.

```
Public Sub save()
    Dim num As Integer
    main.Select
    num = 4 + main.Range(„D17“).Value
    Dim m As Double
    m = Val(frmadd.txtegn.Text)
    ,Проверка за некоректно въведен номер на билет
    If Val(frmadd.txtnumbilet.Text) = 0 Or _
        IsNumeric(frmadd.txtnumbilet.Text) = False Then
        MsgBox „Грешен номер на билет!“
        Exit Sub
    End If
    ,Проверка за непопълнени данни във формуляра
    If frmadd.txttime.Text = „“ Or frmadd.txtadres.Text = „“
    Or _
        frmadd.txtorgan.Text = „“ Then
        MsgBox „Не сте попълнили всички данни във
формуляра!“
        Exit Sub
    End If
```

```
,Проверка за некоректно въведен ЕГН
If (m <= 1000000000) Or (m >= 9999999999#) Then
    MsgBox „Неправилно въведено ЕГН!“
    Exit Sub
Else
    ,Въвеждане на попълнените данни в регистъра
    reg.Cells(num, 1).Value = frmadd.txtnomer.Text
    If frmadd.optplaten.Value = True Then reg.Cells(num,
2).Value = _
        „1. Платен“
    If frmadd.optdeca.Value = True Then reg.Cells(num,
2).Value = _
        „2. За деца (под 14 г.)“
    If frmadd.opt65man.Value = True Then reg.Cells(num,
2).Value = _
        „3. Навършили 65 г. мъже“
    If frmadd.opt60woman.Value = True Then reg.Cells(num,
2).Value = _
        „4. Навършили 60 г. жени“
    If frmadd.optinvalidi.Value = True Then reg.
Cells(num, 2).Value = _
        „5. Инвалиди“
    If frmadd.optgod.Value = True Then reg.Cells(num,
3).Value = _
        „1. Годишен“
    If frmadd.optpolugodie.Value = True Then reg.
Cells(num, 3).Value = _
        „2. Полугодие“
    If frmadd.optmesechen.Value = True Then reg.
Cells(num, 3).Value = _
        „3. Месечен“
    If frmadd.optsedmichen.Value = True Then reg.
Cells(num, 3).Value = _
        „4. Седмичен“
    reg.Cells(num, 4).Value = Val(frmadd.txtnumbilet.
Text)
    Dim l As Integer
    Dim s As String
```

```
l = Len(frmadd.txtdata.Text)
If Right(frmadd.txtdata.Text, 2) = „r.“ Then
    s = Left(frmadd.txtdata.Text, l - 2)
    reg.Cells(num, 5).Value = DateValue(s)
Else
    reg.Cells(num, 5).Value = DateValue(frmadd.txtdata.
Text)
End If
reg.Cells(num, 6).Value = frmadd.txtorgan.Text
reg.Cells(num, 7).Value = frmadd.txttime.Text
reg.Cells(num, 8).Value = frmadd.txttegn.Text
reg.Cells(num, 9).Value = frmadd.txtadres.Text
,Подготовка за въвеждане на нови данни във формата
frmadd.txtnomer.Text = main.Range(„D17“).Value + 1
frmadd.txttime.Text = „“
frmadd.txttegn.Text = „“
frmadd.txtadres.Text = „“
frmadd.txtorgan.Text = „“
frmadd.optplaten.Value = True
frmadd.optgod.Value = True
frmadd.txtnumbilet.Text = „“
Dim today As Variant
today = Now
frmadd.txtdata.Text = Format(today, „dd.mm.yyyy r.“)
main.Select
End If
End Sub
```

Задача 6: Да се създаде процедурен макрос с име search, който по въведени данни и избран критерий за търсене (ЕГН или Име, презиме и фамилия) намира желанния запис в регистъра.

```
Public Sub search()
    Dim s1, s2, pom As String
    Dim num, i As Integer
    Dim f As Boolean
    s1 = frmspravka.txtsearchegn.Text
    s2 = frmspravka.txtsearchime.Text
    main.Select
    num = main.Range(“D17”).Value
```

```
If frmspravka.OptionButton1.Value = True And s1 <> ""
Then
    Dim m As Double
    m = Val(s1)
    If (m <= 1000000000) Or (m >= 9999999999#) And _
        frmspravka.OptionButton1.Value = True Then
        MsgBox "Неправилно въведено ЕГН!"
        Exit Sub
    Else
        f = False
        For i = 1 To num
            pom = reg.Cells(3 + i, 8).Value
            If pom = s1 Then
                reg.Select
                Rows(i + 3).Select
                f = True
                Unload frmspravka
                showsearch_result (i + 3)
            End If
        Next i
    End If
End If
If frmspravka.OptionButton2.Value = True And s2 <> ""
Then
    f = False
    For i = 1 To num
        pom = reg.Cells(3 + i, 7).Value
        If pom = s2 Then
            reg.Select
            Rows(i + 3).Select
            f = True
            Unload frmspravka
            showsearch_result (i + 3)
        End If
    Next i
End If
If f = False Then
    MsgBox "Не е намерен запис по зададения критерий за търсене!"
```

```
Unload frmspravka
End If
main.Select
End Sub
```

Задача 7: Да се създаде процедура с име showsearch_result и формален параметър k, която да се извиква от процедурата search и да визуализира в потребителска форма данните за намерения запис в регистъра. Параметърът k представлява номера на реда в таблицата, на който се намира търсения запис. В общата част на модула запишете: Dim isoptdeca As Boolean

```
Sub showsearch_result(k As Integer)
    isoptdeca = False
    frmshow.txtnomer.Text = reg.Cells(k, 1).Value
    If reg.Cells(k, 2).Value = „1. Платен“ Then frmshow.
optplaten.Value = True
    If reg.Cells(k, 2).Value = „2. За деца (под 14 г.)“ Then
        frmshow.optdeca.Value = True
        frmshow.cmdshow.Visible = True
        isoptdeca = True
    End If
    If reg.Cells(k, 2).Value = „3. Навършили 65 г. мъже“
Then _
        frmshow.opt65man.Value = True
    If reg.Cells(k, 2).Value = „4. Навършили 60 г. жени“
Then _
        frmshow.opt60woman.Value = True
    If reg.Cells(k, 2).Value = „5. Инвалиди“ Then _
        frmshow.optinvalidi.Value = True
    If reg.Cells(k, 3).Value = „1. Годишен“ Then frmshow.
optgod.Value = True
    If reg.Cells(k, 3).Value = „2. Полугодие“ Then _
        frmshow.optpolugodie.Value = True
    If reg.Cells(k, 3).Value = „3. Месечен“ Then _
        frmshow.optmesechen.Value = True
    If reg.Cells(k, 3).Value = „4. Седмичен“ Then _
        frmshow.optsedmichen.Value = True
    frmshow.txtnumbilet.Text = reg.Cells(k, 4).Value
    frmshow.txtdata.Text = reg.Cells(k, 5).Value
    frmshow.txtorgan.Text = reg.Cells(k, 6).Value
```

```
frmshow.txttime.Text = reg.Cells(k, 7).Value
frmshow.txtegn.Text = reg.Cells(k, 8).Value
frmshow.txtadres.Text = reg.Cells(k, 9).Value
frmshow.Show
```

```
End Sub
```

Задача 8: Да се създаде процедура с име update, която при необходимост ще извършва актуализация на вече въведени данни в регистъра.

```
Public Sub update()
    Dim num As Integer
    num = 3 + Val(frmshow.txtnomer.Text)
    Dim m As Double
    m = Val(frmshow.txtegn.Text)
    'Проверка за некоректно въведен номер на билет
    If Val(frmshow.txtnumbilet.Text) = 0 Or _
        IsNumeric(frmshow.txtnumbilet.Text) = False Then
        MsgBox "Грешен номер на билет!"
        Exit Sub
    End If
    ,Проверка за непълнени данни във формуляра
    If frmshow.txttime.Text = "" Or frmshow.txtadres.Text =
"" Or _
        frmshow.txtorgan.Text = "" Then
        MsgBox "Не сте попълнили всички данни във формуляра!"
        Exit Sub
    End If
    ,Проверка за некоректно въведен ЕГН
    If (m <= 1000000000) Or (m >= 9999999999#) Then
        MsgBox "Неправилно въведено ЕГН!"
        Exit Sub
    Else
        If MsgBox("Желаете ли да съхраните актуализираните
данни?", vbYesNo) = _
            vbYes Then
            If frmshow.optplaten.Value = True Then reg.Cells(num,
2).Value = _
                "1. Платен"
            If frmshow.optdeca.Value = True Then reg.Cells(num,
2).Value = _
```

```
    „2. За деца (под 14 г.)“
    If frmshow.opt65man.Value = True Then reg.Cells(num,
2).Value = _
    „3. Навършили 65 г. мъже“
    If frmshow.opt60woman.Value = True Then reg.Cells(num,
2).Value = _
    „4. Навършили 60 г. жени“
    If frmshow.optinvalidi.Value = True Then reg.Cells(num,
2).Value = _
    „5. Инвалиди“
    If frmshow.optgod.Value = True Then reg.Cells(num,
3).Value = _
    „1. Годишен“
    If frmshow.optpolugodie.Value = True Then reg.
Cells(num, 3).Value = _
    „2. Полугодие“
    If frmshow.optmesechen.Value = True Then reg.Cells(num,
3).Value = _
    „3. Месечен“
    If frmshow.optsedmichen.Value = True Then reg.
Cells(num, 3).Value = _
    „4. Седмичен“
    reg.Cells(num, 4).Value = frmshow.txtnumbilet.Text
    Dim l As Integer
    Dim s As String
    l = Len(frmshow.txtdata.Text)
    If Right(frmshow.txtdata.Text, 2) = „г.“ Then
        s = Left(frmshow.txtdata.Text, l - 2)
        reg.Cells(num, 5).Value = DateValue(s)
    Else
        reg.Cells(num, 5).Value = DateValue(frmshow.txtdata.
Text)
    End If
    reg.Cells(num, 6).Value = frmshow.txtorgan.Text
    reg.Cells(num, 7).Value = frmshow.txttime.Text
    reg.Cells(num, 8).Value = frmshow.txttegn.Text
    reg.Cells(num, 9).Value = frmshow.txtadres.Text
    If isoptdeca = True And frmshow.optdeca.Value = False Then
```

```
reg.Cells(num, 10).Value = ""
reg.Cells(num, 11).Value = ""
reg.Cells(num, 12).Value = ""
reg.Cells(num, 13).Value = ""
reg.Cells(num, 14).Value = ""
End If
If isoptdeca = False And frmshow.optdeca.Value = True
Then
    MsgBox „При актуализацията не сте въвели данни за
    представителя!“
    frmpredstavitel.CommandButton1.Caption = _
    „Актуализация на данните на представителя“
    frmpredstavitel.Show
End If
End If
End If
frmshow.Hide
main.Select
End Sub
```

Добавяне на програмен код (събитийни процедури) към потребителските форми на информационната система

Към формата frmadd за въвеждане на данни добавете следния програмен

код:

```
Private Sub cmdspravka_oldreg_Click()
    spravka_oldreg
End Sub
Private Sub cmdexit_Click()
    frmadd.Hide
    main.Select
End Sub
Private Sub optdeca_Click()
    frmpredstavitel.prime.Text = ""
    frmpredstavitel.pregn.Text = ""
    frmpredstavitel.pradres.Text = ""
    frmpredstavitel.prlk.Text = ""
    frmpredstavitel.prdataizdavane.Text = ""
```

```
frmpredstavitel.CommandButton1.Caption = _  
    "Съхрани данните на представителя"  
frmpredstavitel.Show  
End Sub  
Private Sub cmdsave_Click()  
    Call save  
End Sub  
'Събитийна процедура за ограничаване на допустимите  
клавиши при въвеждане ,на стойност в полето txtegn (само  
цифри)  
Private Sub txtegn_KeyPress(ByVal KeyAscii As MSForms.  
ReturnInteger)  
    If KeyAscii < Asc(",0") Or KeyAscii > Asc(",9") Then  
        KeyAscii = 0  
    End Sub
```

Към формата frmpredstavitel за въвеждане или актуализация на данни на законния представител добавете следния програмен код:

```
Private Sub CommandButton1_Click()  
    main.Select  
    If CommandButton1.Caption = "Съхрани данните на пред-  
ставителя" Then  
        num = 4 + main.Range("D17").Value  
    ElseIf CommandButton1.Caption = _  
        "Актуализация на данните на представителя" Then  
        num = 3 + frmshow.txtnomer.Text  
    End If  
    Dim m As Double  
    m = Val(pregn.Text)  
    If (m <= 1000000000) Or (m >= 9999999999#) Then  
        MsgBox "Неправилно въведено ЕГН!"  
    Exit Sub  
    Else  
        reg.Cells(num, 10).Value = prime.Text  
        reg.Cells(num, 11).Value = pregn.Text  
        reg.Cells(num, 12).Value = pradres.Text  
        reg.Cells(num, 13).Value = prlk.Text  
        Dim l As Integer
```

```
Dim s As String
l = Len(prdataizdavane.Text)
If Right(prdataizdavane.Text, 2) = „г.“ Then
s = Left(prdataizdavane.Text, l - 2)
reg.Cells(num, 14).Value = DateValue(s)
Else
If prdataizdavane.Text <> „“ Then _
reg.Cells(num, 14).Value = DateValue(prdataizdavane.
Text)
End If
frmpredstavitel.Hide
prime.Text = „“
pregn.Text = „“
pradres.Text = „“
prlk.Text = „“
prdataizdavane.Text = „“
End If
End Sub

Private Sub pregn_KeyPress(ByVal KeyAscii As MSForms.
ReturnInteger)
If KeyAscii < Asc(„0“) Or KeyAscii > Asc(„9“) Then
KeyAscii = 0
End Sub

Private Sub prlk_KeyPress(ByVal KeyAscii As MSForms.
ReturnInteger)
If KeyAscii < Asc(„0“) Or KeyAscii > Asc(„9“) Then
KeyAscii = 0
End Sub
```

Към формата frmspravka добавете следния програмен код:

```
Private Sub cmdsearch_Click()
search
End Sub

Private Sub OptionButton1_Click()
txtsearchime.Text = „“
txtsearchime.Enabled = False
txtsearchegn.Enabled = True
txtsearchime.BackColor = &H8000000F
```

```
txtsearchegn.BackColor = vbWhite
txtsearchegn.SetFocus
End Sub
Private Sub OptionButton2_Click()
    txtsearchegn.Text = ""
    txtsearchegn.Enabled = False
    txtsearchime.Enabled = True
    txtsearchegn.BackColor = &H8000000F
    txtsearchime.BackColor = vbWhite
    txtsearchime.SetFocus
End Sub
Private Sub txtsearchegn_KeyPress(ByVal KeyAscii As
MSForms.ReturnInteger)
    If KeyAscii < Asc(",0") Or KeyAscii > Asc(",9") Then
        KeyAscii = 0
    End Sub
```

Към формата frmshow добавете следния програмен код:

```
Private Sub cmdshow_Click()
    Dim num As Integer
    num = 3 + Val(txtnomer.Text)
    frmpredstavitel.prime.Text = reg.Cells(num, 10).Value
    frmpredstavitel.pregn.Text = reg.Cells(num, 11).Value
    frmpredstavitel.pradres.Text = reg.Cells(num, 12).Value
    frmpredstavitel.prlk.Text = reg.Cells(num, 13).Value
    frmpredstavitel.prdataizdavane.Text = reg.Cells(num,
14).Value
    frmpredstavitel.CommandButton1.Caption = _
        "Актуализация на данните на представителя"
    frmpredstavitel.Show
End Sub
Private Sub cmdexit_Click()
    frmshow.Hide
    main.Select
End Sub
Private Sub opt60woman_Click()
    cmdshow.Visible = False
End Sub
```

```
Private Sub opt65man_Click()  
    cmdshow.Visible = False  
End Sub  
Private Sub optdeca_Click()  
    cmdshow.Visible = True  
End Sub  
Private Sub optinvalidi_Click()  
    cmdshow.Visible = False  
End Sub  
Private Sub optplatten_Click()  
    cmdshow.Visible = False  
End Sub  
Private Sub cmdupdate_Click()  
    update  
End Sub
```

III. Заключение

Предложената в работата примерна технология за автоматизиране на дейностите при работа с документи и данни на Excel може да се приложи по аналогия и за автоматизиране на дейностите и разширяване на възможностите и за други приложения от Microsoft Office.

Описаната в статията **примерна практическа задача** лесно може да се **формулира** и в друг контекст, като се получат **различни варианти** на задания, имащи приложения в практиката. Така например може да се разглежда подобна задача, при която входните данни са медицинските картони на учениците от дадено училище или картотеките на футболните отбори, играещи в определена група и т.н. По този начин за **самостоятелна работа**, учениците могат да разработват задания, близки до техните интереси, а на най-подготвените може да се възложи **сами** да съставят условието на задачата, която трябва да решат. Разглежданата тематика ще бъде полезна в практиката на учителите по информационни технологии, които преподават предмета в профилираната подготовка.

БЕЛЕЖКИ

1. Учебни програми III част за задължителна и профилирана подготовка IX, X, XI и XII клас, Главна редакция на педагогическите издания към МОН, София, 2003 г.

ЛИТЕРАТУРА

- Гъров, К. & Анева С. (2006). Използване на Visual Basic for Application в обучението по информационни технологии в средното училище, *Математика и информатика*, 2, 18–28.
- Рахнев, А., Гъров К. & Анева С. (2012). Примерна система от задачи при изучаване на Visual Basic for Application в средното училище, *Образование и технологии*, 3, 58–59,
- Роман С. (2004). *Да напишем макроси в Excel*, София, ЗеСТ Прес.
- Роман С. (1999). *Да напишем макроси в Word*, София, ЗеСТ Прес.

✉ **Асен Рахнев**

Професор, доктор
Факултет по математика и информатика
ПУ „Паисий Хилендарски“,
Бул. България № 236, 4003 Пловдив, България
E-mail: assen@uni-plovdiv.bg
dean-fmi@uni-plovdiv.bg

✉ **Коста Гъров**

Доцент, доктор
Факултет по математика и информатика
ПУ „Паисий Хилендарски“,
Бул. България № 236, 4003 Пловдив, България
E-mail: kosgar@uni-plovdiv.bg

✉ **Стефка Анева**

Главен асистент
Факултет по математика и информатика
ПУ „Паисий Хилендарски“,
Бул. България № 236, 4003 Пловдив, България
E-mail: stfaneva@uni-plovdiv.bg

USING VISUAL BASIC FOR APPLICATIONS IN SOLVING PRACTICAL PROBLEMS IN INFORMATION TECHNOLOGY TRAINING

Abstract. This paper is dedicated to the use of Visual Basic for Applications for teaching the module „Integration of activities and documents“ in specialized Information Technology training in upper classes of secondary school. We consider a practical problem where the main stages of the implementation of Visual Basic for Applications are made in MS Excel. The paper is intended for IT teachers.

✉ **Asen Rahnev**

Professor, PhD

Faculty of Mathematics and Informatics
Plovdiv University „Paisii Hilendarski“,
236 Bulgaria Blvd., 4003 Plovdiv, Bulgaria

E-mail: assen@uni-plovdiv.bg
dean-fmi@uni-plovdiv.bg

✉ **Kosta Garov**

Associated professor, PhD

Faculty of Mathematics and Informatics
Plovdiv University „Paisii Hilendarski“,
236 Bulgaria Blvd., 4003 Plovdiv, Bulgaria

E-mail: kosgar@uni-plovdiv.bg

✉ **Stefka Aneva**

Assistant professor

Faculty of Mathematics and Informatics
Plovdiv University „Paisii Hilendarski“,
236 Bulgaria Blvd., 4003 Plovdiv, Bulgaria

E-mail: stfaneva@uni-plovdiv.bg