

Тематика практических занятий.

VII семестр

Пр. 1. Разработка макросов и модулей VBA в MS Excel, в MS Word. [Л. 1.1: с. 231 – 274; 276 – 295, Л. 2.1: с. 18 - 33]

Задание.

Создать новую книгу **Excel**, используя метод **Add** .

Решение. Следующая процедура создает новую книгу. По умолчанию Excel автоматически присваивает ей имя – Book*N*, где *N* является следующим доступным номером. Новая книга становится активной.

```
Sub AddOne ()
    Workbooks.Add
End Sub
```

Более совершенным подходом является использование ссылки на создаваемую книгу с помощью объектной переменной. Использование объектной переменной делает более доступным управление созданным объектом (новой книгой). В следующем фрагменте объект **Workbook**, возвращаемый методом **Add**, связывается с объектной переменной *NewBook*. Далее выполняется установка значений некоторых свойств объекта *NewBook*. Можно легко управлять вновь созданным объектом *NewBook*, используя объектную переменную.

```
Sub AddNew ()
    Set NewBook = Workbooks.Add
    With NewBook
        .Title = "All Sales"
        .Subject = "Sales"
        .SaveAs Filename:="Allsales.xls"
    End With
End Sub
```

-
-
-

Задание.

Создать новый документ **Word**

Решение. Семейство **Documents** содержит все открытые документы. Для создания нового документа используется метод **Add** , добавляющий объект **Document** в семейство **Documents**. Следующая инструкция создает новый документ.

```
Documents.Add
```

Усовершенствованный прием состоит в назначении величины, возвращаемой методом **Add**, некоторой объектной переменной. В следующем фрагменте **Document** – объект, возвращаемый методом **Add**, назначается объектной переменной docNew. Благодаря этому, вновь созданный объект становится более доступным в управлении. Сразу же можно установить свойства и методы для нового объекта **Document**.

```
Sub NewSampleDoc ()
    Dim docNew As Document
    Set docNew = Documents.Add
    With docNew
        .Content.Font.Name = "Tahoma"
        .SaveAs FileName:="Sample.doc"
    End With
End Sub
```

•
•
•

Пр. 2. Разработка макросов и модулей VBA в MS Access. [Л. 1.1: с. 315 – 338, Л 1.2: с. 192 – 300, Л. 2.1: с. 33 – 52, Л. 4.2]

Задание.

Создать новый экземпляр класса приложения **Access** из некоторого приложения, поддерживающего или нет ключевое слово **New**.

Решение. Если приложение поддерживает ключевое слово **New**, то достаточно одной инструкции, декларирующей объектную переменную:

```
Dim appAccess As New Access.Application
```

Из приложения, не поддерживающего **New**, новый экземпляр класса **Application** можно создать с помощью метода **CreateObject** :

```
Dim appAccess As Object

Set appAccess = CreateObject("Access.Application")
```

Задание.

Создать новую базу данных Microsoft Access из другого приложения, являющегося COM – компонентой. В новой базе данных создать таблицу "Contacts", содержащую поле "CompanyName".

Решение. Переменную – указатель на объект **Application** описываем как переменную уровня модуля, помещая ее в раздел описаний модуля.

```
Dim appAccess As Access.Application

Sub NewAccessDatabase ()
    Dim dbs As Object, tdf As Object, fld As Variant
    Dim strDB As String
```

```

Const DB_Text As Long = 10
Const FldLen As Integer = 40

' Указываем путь к создаваемой базе данных.
strDB = "C:\My Documents\Newdb.mdb"
' Создаём новый экземпляр класса приложения Microsoft Access.
Set appAccess = _
    CreateObject("Access.Application.9")
' Создаем и открываем базу данных в окне Microsoft Access.
appAccess.NewCurrentDatabase strDB
' Устанавливаем ссылку на вновь созданную базу данных.
Set dbs = appAccess.CurrentDb
' Создаем новую таблицу и устанавливаем ссылку на нее.
Set tdf = dbs.CreateTableDef("Contacts")
' Создаем поле в новой таблице и устанавливаем ссылку на него.
Set fld = tdf._
    CreateField("CompanyName", DB_Text, FldLen)
' Добавляем созданные объекты - поле и таблицу в соответствующие
семейства.
tdf.Fields.Append fld
dbs.TableDefs.Append tdf
Set appAccess = Nothing
End Sub

```

•

•

•

VIII семестр

Пр.1. Программирование с использованием объектов DAO в Microsoft Access. [Л. 1.1: с. 340 – 360, 528 -538 Л. 1.2: с. 70 - 77]

Задание.

Перечислить семейства Properties и Workspaces объекта **DBEngine**.

Решение. В процессе перечисления предусмотрим вывод на печать свойств, значения которых ошибочны.

```

Sub DBEngineX()

    Dim wrkLoop As Workspace
    Dim prpLoop As Property

    With DBEngine
        Debug.Print "DBEngine Properties"

        ' Перечисление Property-объектов семейства Properties объекта DBEngine,
        ' с указанием тех свойств, значения которых не являются допустимыми.
        For Each prpLoop In .Properties
            On Error Resume Next
            Debug.Print " " & prpLoop.Name & " = " _
                & prpLoop
            On Error GoTo 0
        Next prpLoop
    End With

```

```

Debug.Print "Workspaces collection of DBEngine"

' Перечисление Workspace-объектов семейства Workspaces объекта DBEngine.
For Each wrkLoop In .Workspaces
    Debug.Print " " & wrkLoop.Name

    ' Перечисление свойств соответствующего объекта Workspace,
    ' с указанием тех свойств, значения которых не являются допустимыми.
    For Each prpLoop In wrkLoop.Properties
        On Error Resume Next
        Debug.Print " " & prpLoop.Name & _
            " = " & prpLoop
        On Error GoTo 0
    Next prpLoop

Next wrkLoop

End With

End Sub

```

•
•
•

Пр.2. Динамический обмен данными – DDE: работа Приложений в режиме "Клиент – Сервер". [Л. 1.1: с. 483 – 510, Л. 2.2, Л. 4.2]

Задание.

Открыть сеанс связи из VB-приложения с документом Formletr.doc в MS Word. Распечатать документ в Word'e, послав команду FilePrint в WordBasic.

Решение.

```

channelNumber = Application.DDEInitiate( _
    app:"WinWord", _
    topic:"C:\WINWORD\FORMLETR.DOC")
Application.DDEExecute channelNumber, "[FILEPRINT]"
Application.DDETerminate channelNumber

```

Задание.

Создать процедуру–макрос в приложении-клиенте MS Word, позволяющую использовать MS Access в роли сервера DDE.

Решение. Будем считать, что на момент выполнения макроса, MS Access запущен.

```

Sub AccessDDE ()
    Dim intChan1 As Integer, intChan2 As Integer
    Dim strQueryData As String

    ' Используем документ System для открытия базы данных Northwind,
    ' в качестве примера.
    ' База данных должна быть открыта перед использованием других
    ' DDE-документов.

```

```

intChan1 = DDEInitiate("MSAccess", "System")
' Может понадобиться изменить путь к базе данных, для того чтобы
  он был действительным.
DDEExecute intChan1, "[OpenDatabase C:\Access\Samples\Northwind.mdb]"

' Получаем все данные по запросу о десяти наиболее дорогих породах.
intChan2 = DDEInitiate("MSAccess", "Northwind.mdb;" _
    & "QUERY Ten Most Expensive Products")
strQueryData = DDERequest(intChan2, "All")
DDETerminate intChan2

' Закрываем базу данных.
DDEExecute intChan1, "[CloseDatabase]"
DDETerminate intChan1

' Печатаем полученные данные для просмотра.
Debug.Print strQueryData
End Sub

```

•
•
•

Литература

- 1.1. Пол Санна и др. Visual Basic для приложений в подлиннике. Пер. с англ. - СПб.: ВHV - Санкт - Петербург, 1997.
- 1.2. Нортон П., Андерсен В. Разработка приложений в Access 97 в подлиннике. Пер. с англ. - СПб.: ВHV - Санкт-Петербург, 1998.
- 2.1. Коновалов В. М. Методические указания к выполнению лабораторных работ по дисциплине "Системное и прикладное программное обеспечение" - вып.2, М.: МГТУ ГА, 1998.
- 2.2. Коновалов В. М. Пособие к выполнению лабораторных работ по дисциплине "Прикладное программное обеспечение" - вып.3, М.: МГТУ ГА, 2002.
- 4.2. Андерсен В. Базы данных Access: Проблемы и решения. Практическое пособие. Издательство: ЭКОМ, 2001.