

Лабораторная работа «Основы программирования в VBA» (VBA №0)

Цель работы: получить простейшие навыки в реализации программ в VBA.

Задания

Лабораторная работа состоит из 11 заданий (см. 6-8, приложение 1). Каждое задание - программный код, который нужно набрать в **VBA**, протестировать и получить результат, уметь ответить на вопрос, что реализовано той или иной строкой программного кода, что означает каждое «слово» программы. Для самотестирования рекомендуется выполнить устно задание «Контрольное задание» на стр. 9-10 приложения 1.

Для написания программного кода используется редактор **Visual Basic Editor (VBE)**.

Технология работы

- Открыть *Microsoft Excel*.
- Создать в своей папке файл *Excel с поддержкой макросов*.

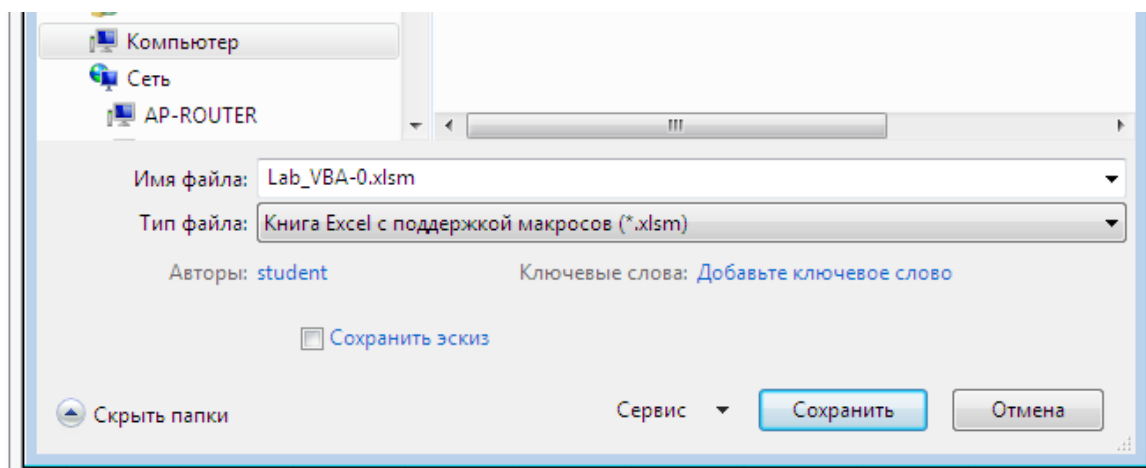
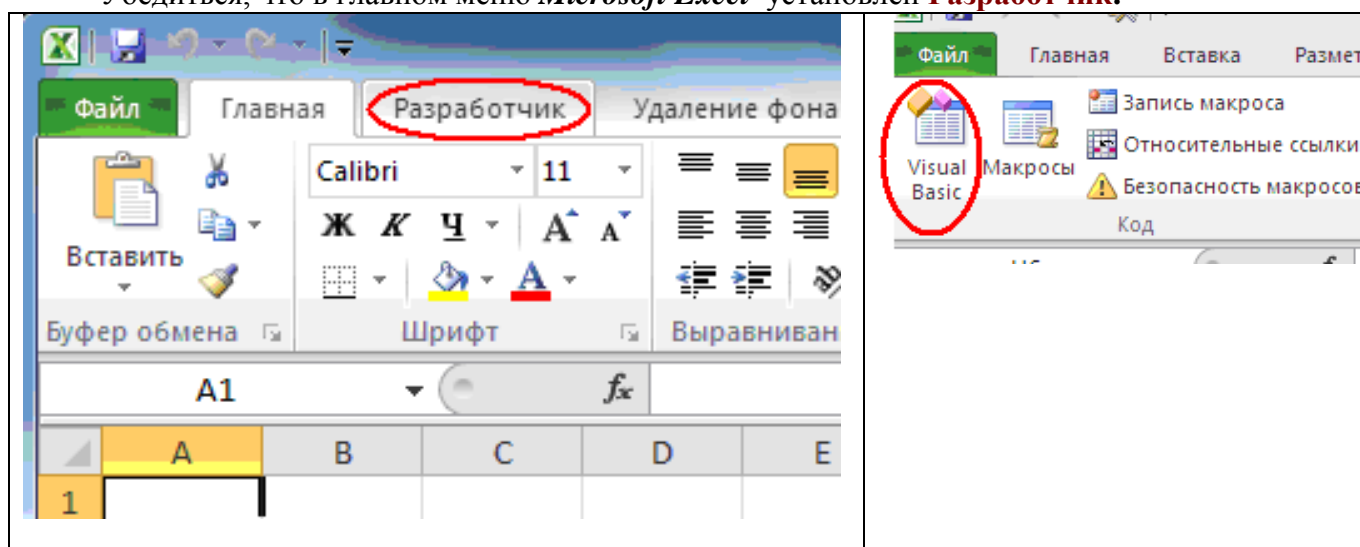


Рис. 1

Убедиться, что в главном меню *Microsoft Excel* установлен **Разработчик**.



сРис. 2

Если в главном меню *Microsoft Excel* **Разработчик** не вынесен, то в приложении 2 (см. стр. 11–15) представлена технология установки **Разработчика**.

- В главном меню *Microsoft Excel* «кликнуть» **Разработчик**.
- В появившемся меню выбрать **Visual Basic**. Появится окно

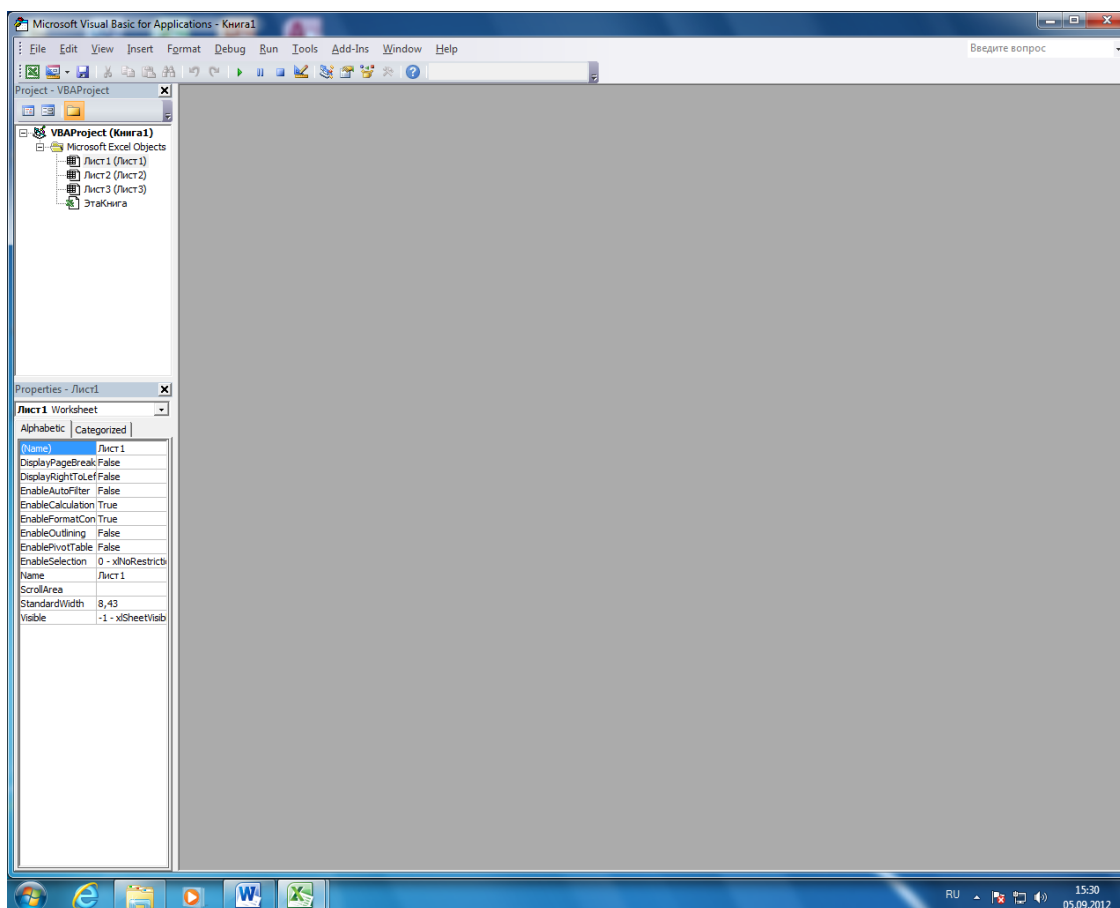


Рис.3

Создание программного модуля в составе рабочей книги: *Insert / Module* (рис. 4).

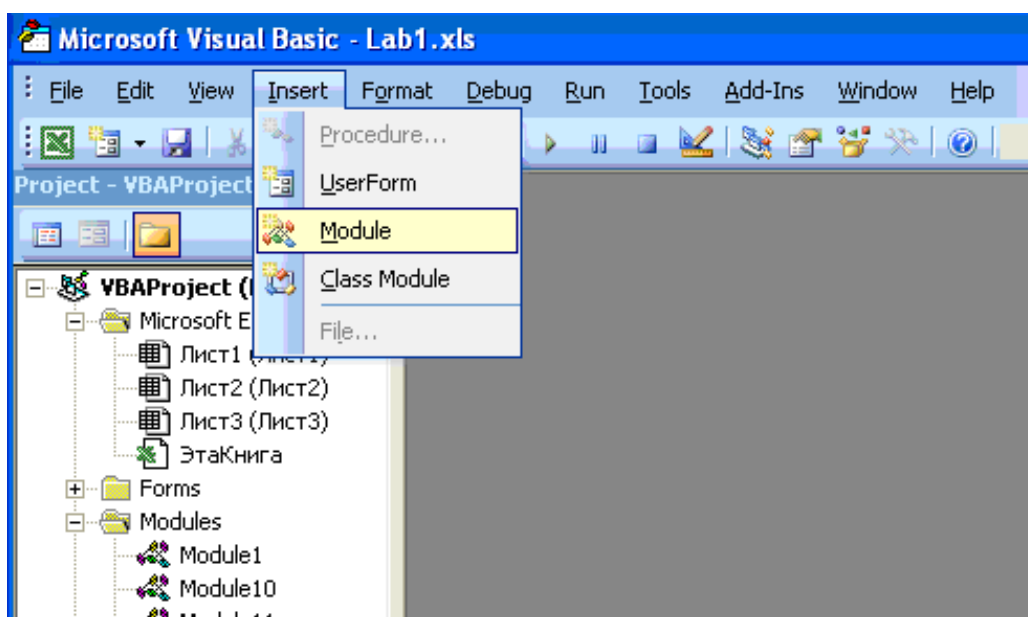


Рис. 4. Создание программного модуля в среде *Microsoft Excel*

При этом создается новый рабочий лист со стандартным именем *Module* и порядковым номером программного модуля в рабочей книге.

Интерфейс редактора VBE

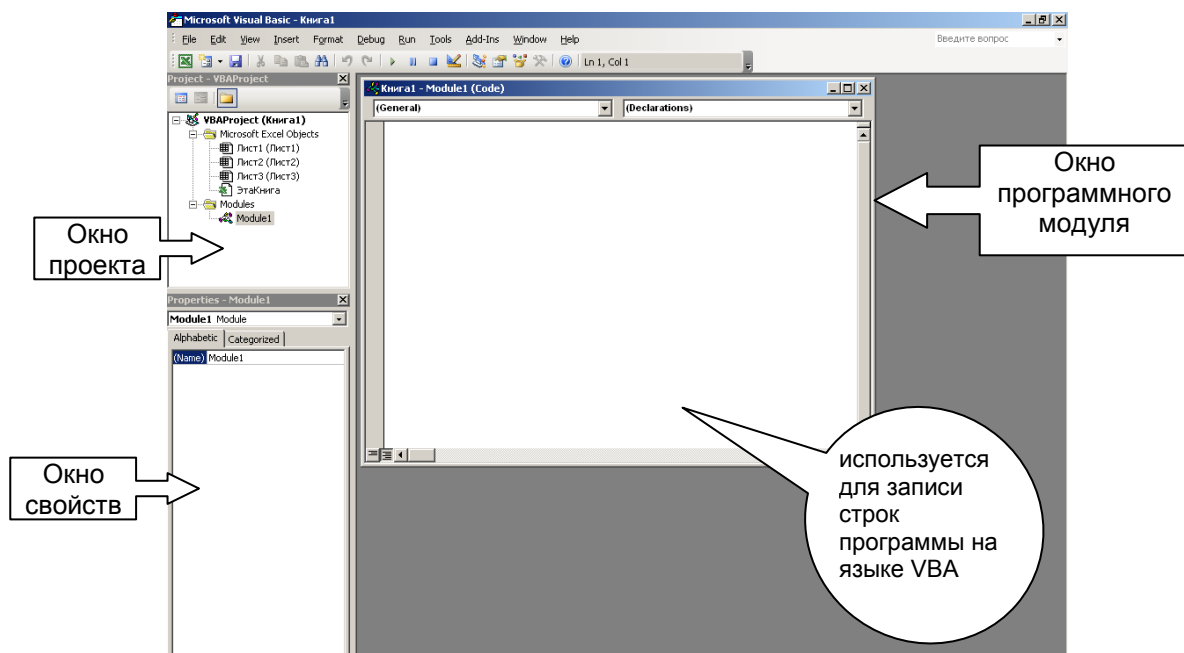


Рис. 5

Список классов объектов можно просмотреть, вызвав окно для просмотра объектов командой **View / Object Browser**.

В окне программного модуля записывается программный код.

Пример:

```
Sub example1()  
Union(Range(Cells(1, 1), Cells(2, 2)), Range(Cells(5, 5), Cells(7, 7))).Select  
End Sub
```

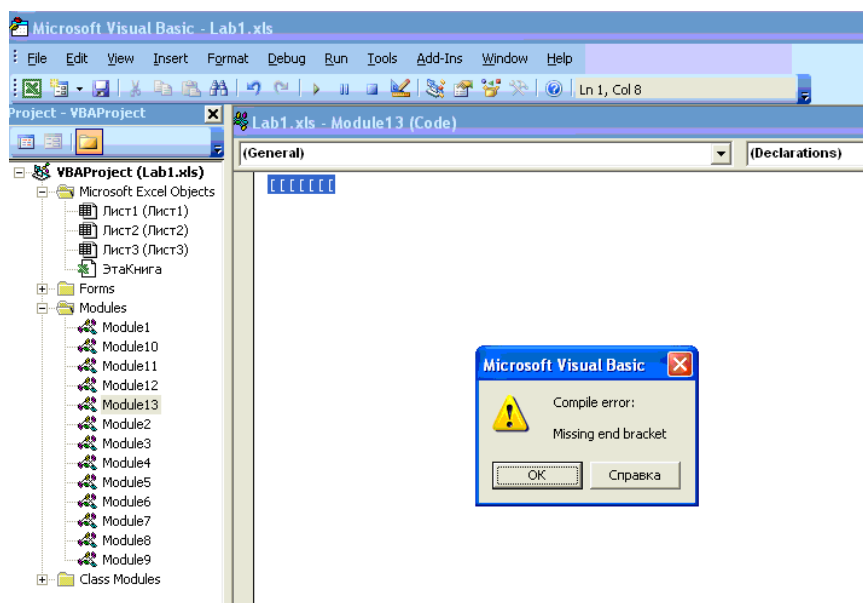
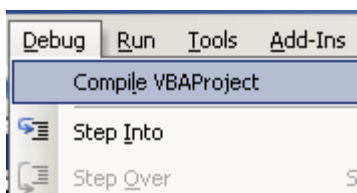


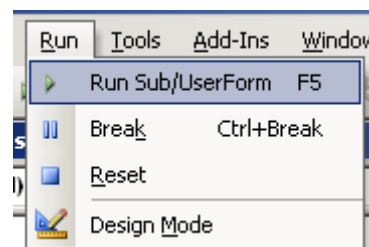
Рис. 6. Интерфейс редактора VBE в среде *Microsoft Excel*, иллюстрирующий наличие ошибки

Если в программе обнаружены ошибки, их необходимо исправить, написать правильно код программы.

Используя главное меню, выполняем последовательность действий: **Debug/ Compile VBAProject**. Чтобы отправить на счет, выполняем последовательность действий: **Run / Run Sub/UserForm**



а



б

Рис. 7.

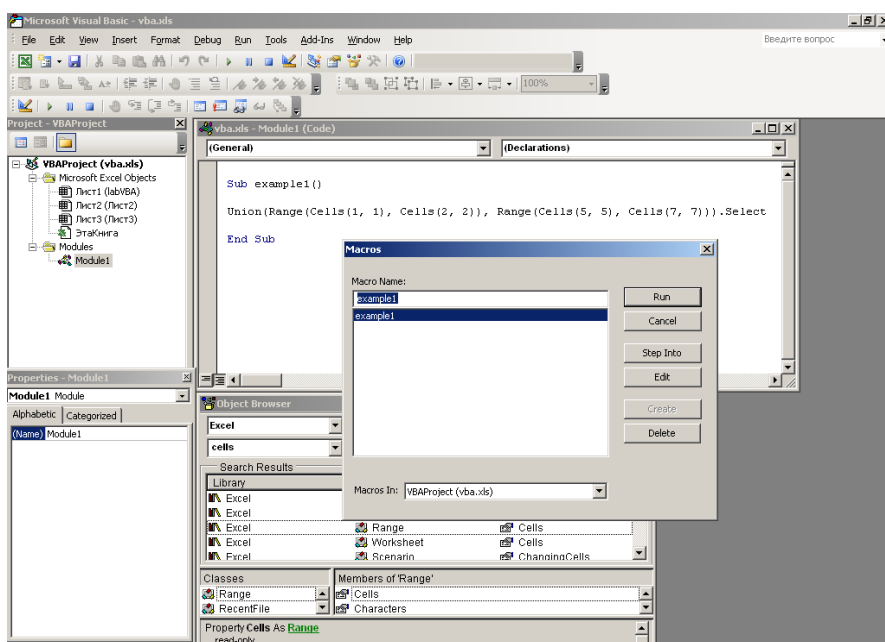


Рис. 8

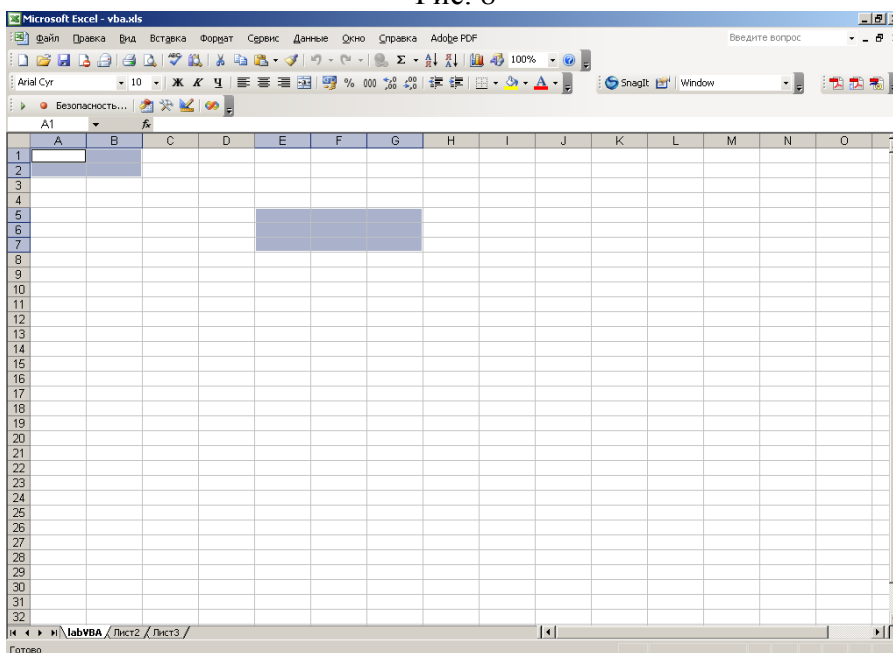


Рис.9. Результат работы программы

Среда разработки программ

Основными пунктами меню, необходимыми для запуска и отладки программы являются:

- Текст (код) программы набирают в окне редактора.
- *Run/Run Sub* – запуск программы;
- *Run/Break* – приостановка выполнения программы;
- *Run/Reset* – прерывание выполнения программы;
- *Debug/Step Into* – выполнить очередной оператор программы;
- *Debug/Step Over* – выполнить оператор программы без входа в процедуры;
- *Debug/Toggle BreakPoint* – установить точку прерывания;
- *Debug/Clear All BreakPoints* – очистить все точки прерывания.

Задания лаб. раб. №0

1.

```
Sub UsingVariable()  
    Dim Num1 As Integer  
    Num1 = Worksheets(1).Range("A1").Value  
    MsgBox Num1  
End Sub
```

2.

```
Public Num1 As Integer  
'  
'*****  
Sub UsingVariable()  
    Num1 = Worksheets(1).Range("A1").Value  
    MsgBox Num1  
End Sub  
'*****
```

3.

```
'*****  
Sub UsingConst()  
    Const Num1 As String = "Tomsk Polytechnic Univ."  
    MsgBox Num1  
End Sub  
'*****
```

4.

```
Const Num1 As String = "Tomsk Polytechnic Univ."  
'*****  
Sub UsingConst()  
    MsgBox Num1  
End Sub  
'*****
```

5.

```
Public Const Num1 As String = "Tomsk Polytechnic Univ."  
'*****  
Sub UsingConst()  
    MsgBox Num1  
End Sub  
'*****
```

6.

Sub Call_Function()

Dim var1 **As Integer**

Dim var2 **As Integer**

Dim var3 **As Integer**

 var1 = 5

 var2 = 10

 var3 = Multiply(var1, var2)

 MsgBox (var3)

End Sub

Function Multiply(**ByVal** var1 **As Integer**, **ByVal** var2 **As Integer**)

 Multiply = var1 * var2

End Function

7.

Sub Proc()

 var1 = GetRand

 ModuleLevelProcedure var1

End Sub

Private Sub ModuleLevelProcedure(**ByVal** var1)

MsgBox var1 * GetRand

End Sub

Private Function GetRand()

 GetRand = **Int**(100 * **Rnd**())

End Function

8.

Sub PassArgumentByReference()

Dim Username **As String**

 Username = "Mike"

 ChangeName Username

MsgBox Username

End Sub

Sub ChangeName(**ByRef** Username)

 Username = "Kate"

End Sub

9.

Sub PassArgumentbyValue()

Dim Username **As String**

 Username = "Mike"

 ChangeName1 Username

MsgBox "Yet remains as " & Username & " in the original proc."

End Sub

Sub ChangeName1(**ByVal** Username)

MsgBox "The original name is " & Username & "."

 Username = "Kate"

MsgBox "The name is changed to " & Username & " in the called proc."

End Sub

10.

Option Base 1

Sub UsingArray()

Dim Vals(3) **As Integer**

 Vals(1) = **Int**(100 * **Rnd**())

 Vals(2) = **Int**(100 * **Rnd**())

 Vals(3) = **Int**(100 * **Rnd**())

MsgBox "Lottery numbers: " & Vals(1) & ", " & Vals(2) & ", " & Vals(3)

End Sub

11.

Option Base 1

Sub Using2DVariantArray()

Dim Data2D(10, 3) **As Variant**

 Data2D(1, 1) = "Kate"

 Data2D(1, 2) = 43

 Data2D(1, 3) = #4/15/1962#

MsgBox Data2D(1, 1) & ", age " & Data2D(1, 2) & ", born " & Data2D(1, 3)

End Sub

Контрольное задание
(Использование основных структур при написании программ на VBA.)

Примеры контрольных заданий

Что реализовано каждой строкой и каждым словом, предложенного программного кода?
Записать арифметическое выражение с использованием стандартных функций VBA.

1).

- **Dim Val As String**
- **Num1 = Worksheets(1).Range("A1").Value**
- **Dim Data2D(10, 3) As Variant**
- **MsgBox Num1**
- **Option Base 1**
- **Sub UsingArray()**
- **Dim Vals(3) As Integer**

2).

Sub Proc()

var1 = GetRand

ModuleLevelProcedure var1

End Sub

Private Sub ModuleLevelProcedure(ByVal var1)

MsgBox var1 * GetRand

End Sub

Private Function GetRand()

GetRand = Int(100 * Rnd())

End Function

3).
$$y_i = x_i^2 \cos^2 e^{\sqrt{x_i}} \frac{|bx_i| - \sqrt{acd}}{\lg|acd| - tg^2 x_i}$$

Рядом с «**MsgBox (var1)**», «**MsgBox (var2)**», «**MsgBox (var3)**» записать результат вывода (число)

Sub Call_Function()

Dim var1 As Integer

Dim var2 As Integer

Dim var3 As Integer

var1 = 5

var2 = 10

MsgBox (var1)

MsgBox (var2)

var3 = Multiply(var1, var2)

MsgBox (var1)

MsgBox (var2)

MsgBox (var3)

End Sub

Function Multiply(ByRef var1 As Integer, ByRef var2 As Integer)

MsgBox (var1)

MsgBox (var2)

Multiply = var1 * var2

var1 = 55

var2 = 77

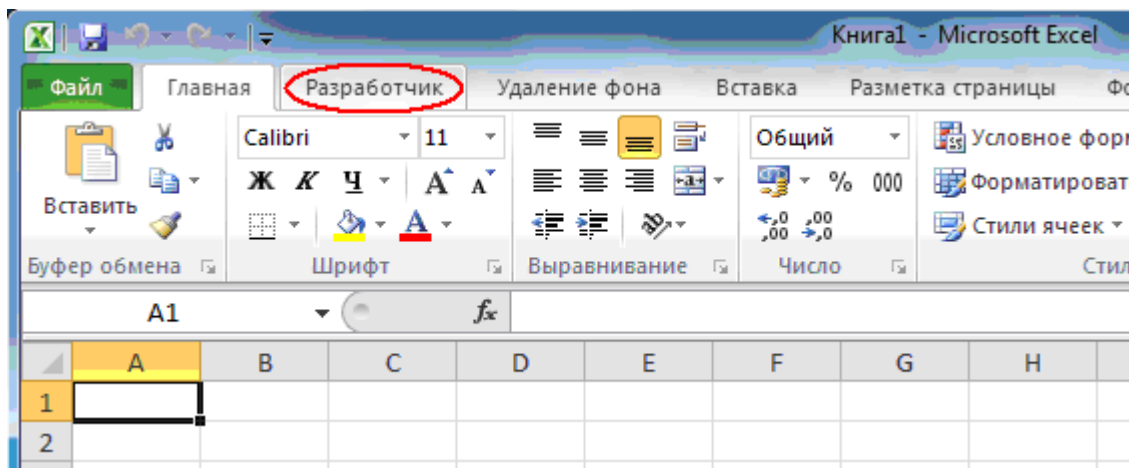
MsgBox (var1)

MsgBox (var2)

End Function

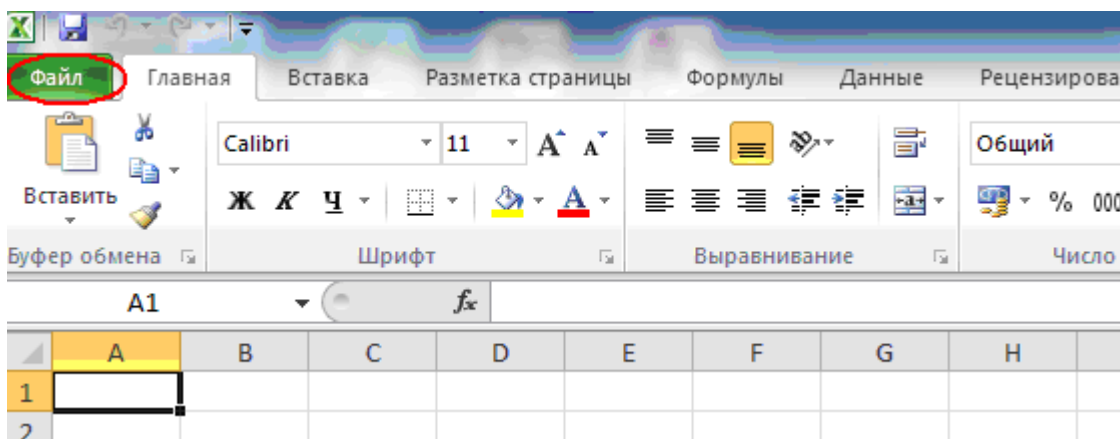
Путь к обращению «**Редактор Visual Basic. Excel 2010**»

Если в главном меню *Excel* отсутствует **Разработчик** (необходимый для работы с объектами *VBA*), то его надо установить.



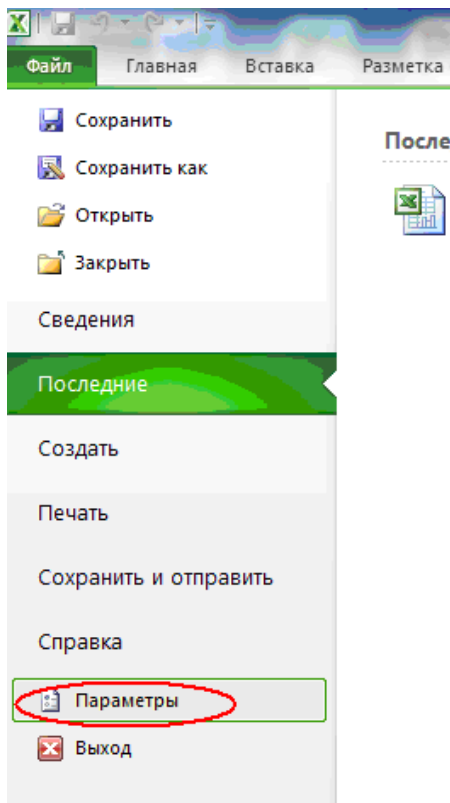
а

Для установки **Разработчика** раскройте вкладку **Файл** главного меню.



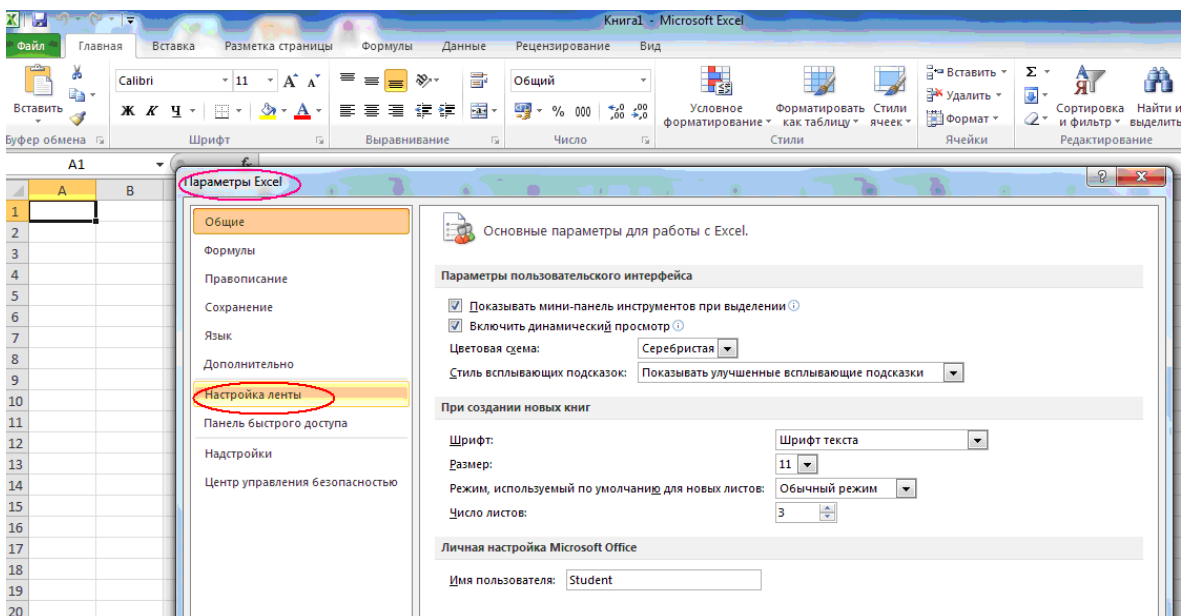
б

Выберите пункт **Параметры** (см. рис. ниже).



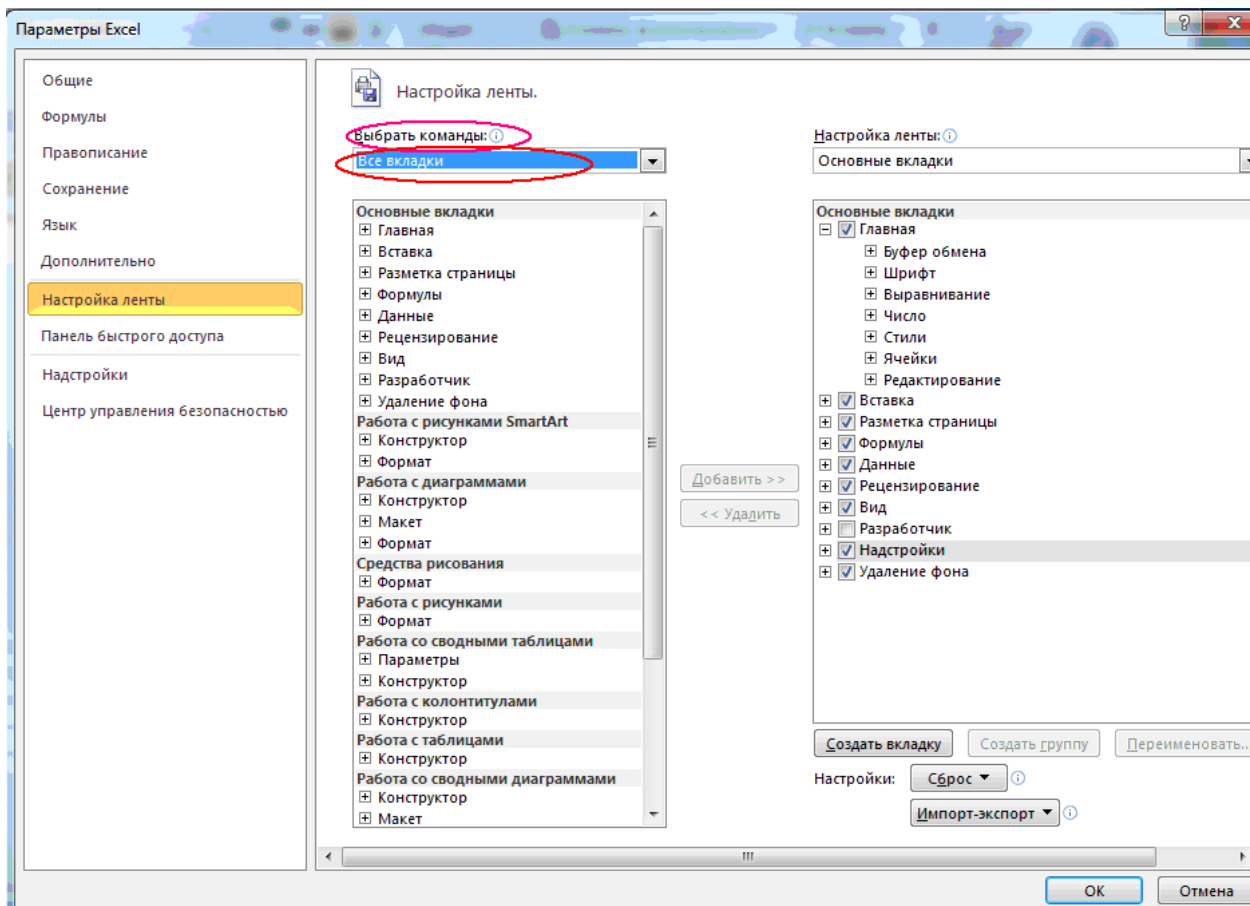
6

В появившемся окне **Параметры Excel** выберите пункт **Настройка ленты** (см. рис. ниже).



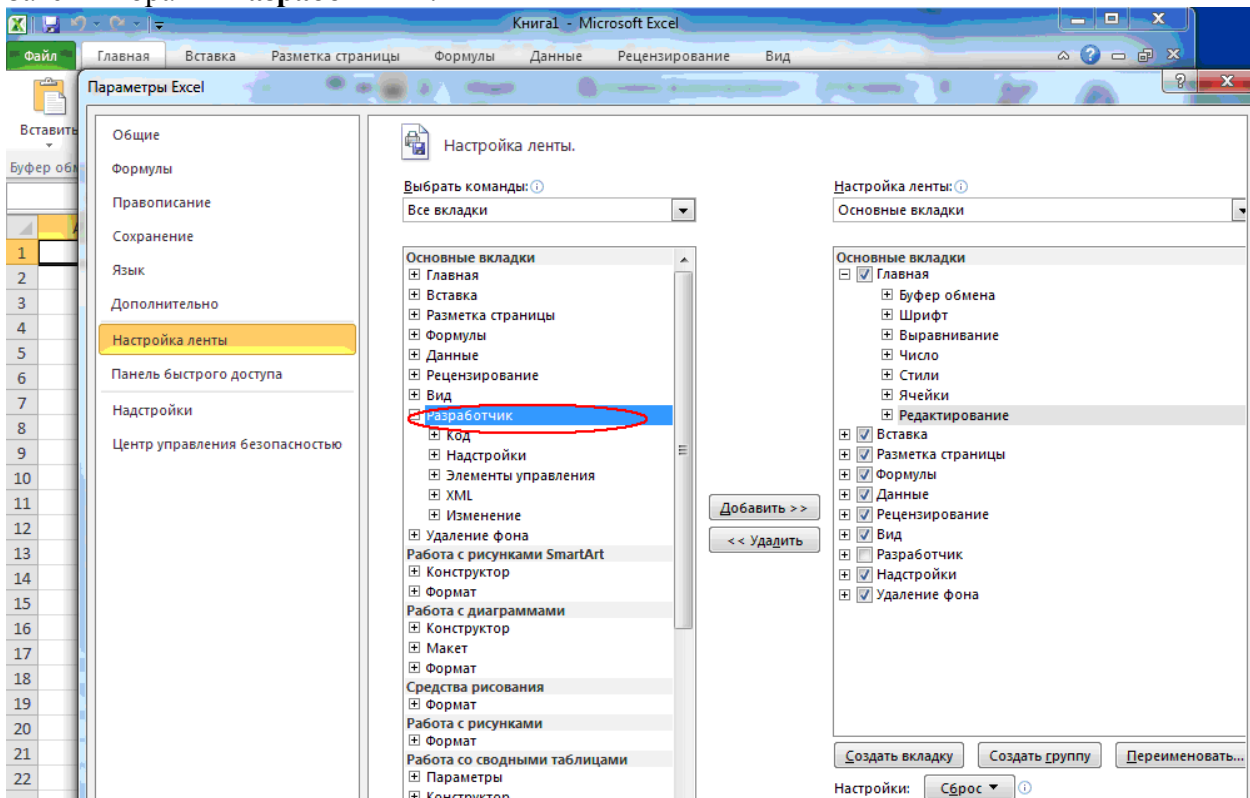
2

В настройке ленты в меню «**Выбрать команды**» выберите пункт «**Все вкладки**» (см. рис. ниже).



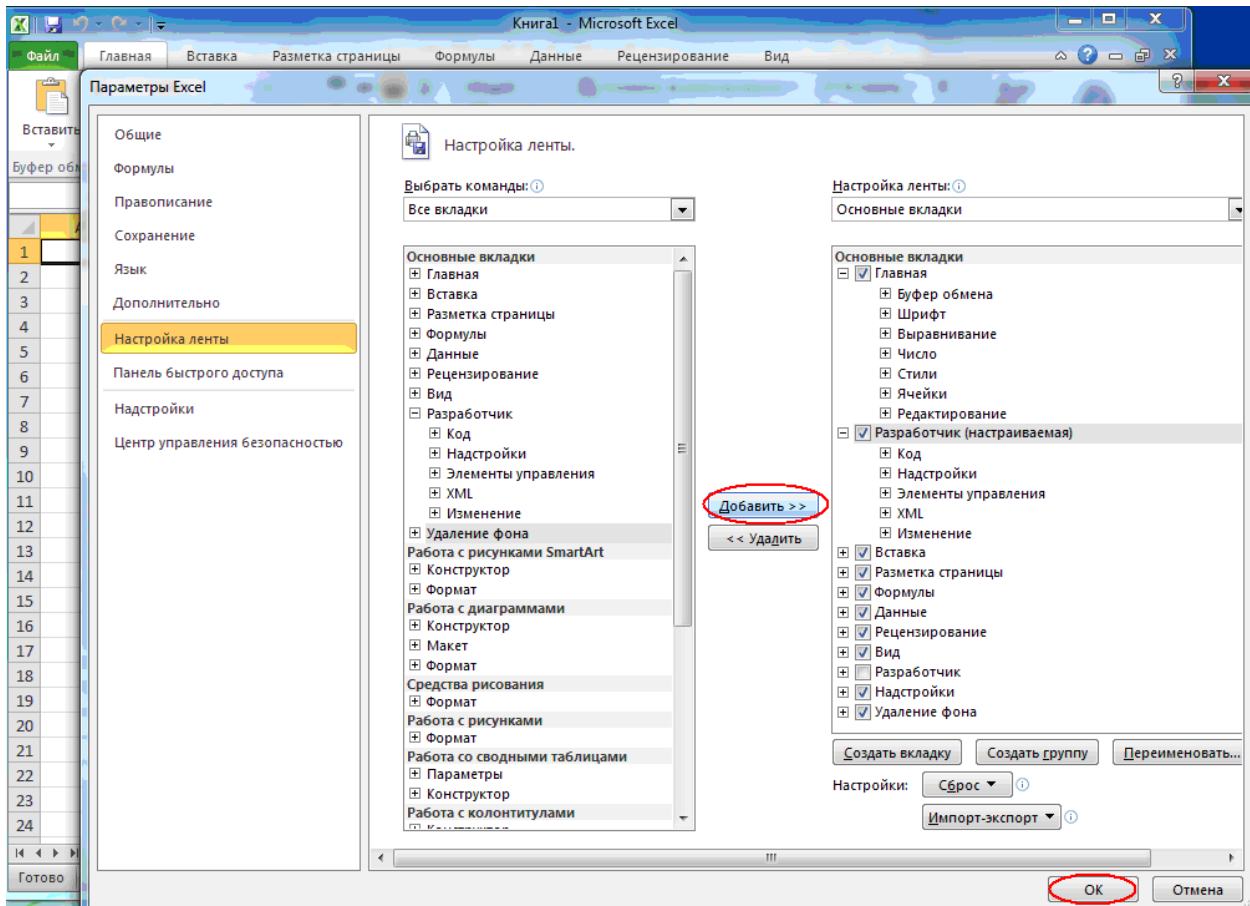
d

Затем выбрать «Разработчик».



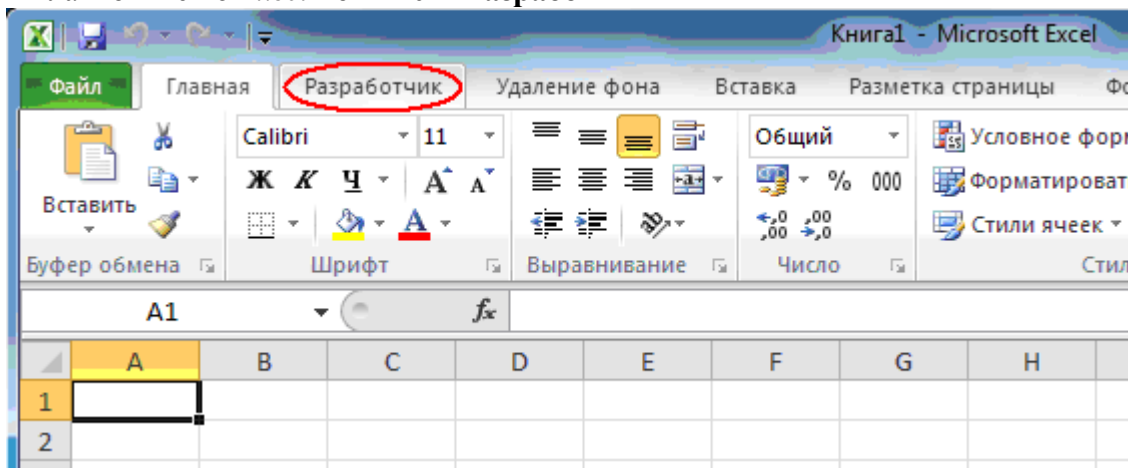
e

Нажать на кнопку «Добавить» и затем нажать на «ОК».



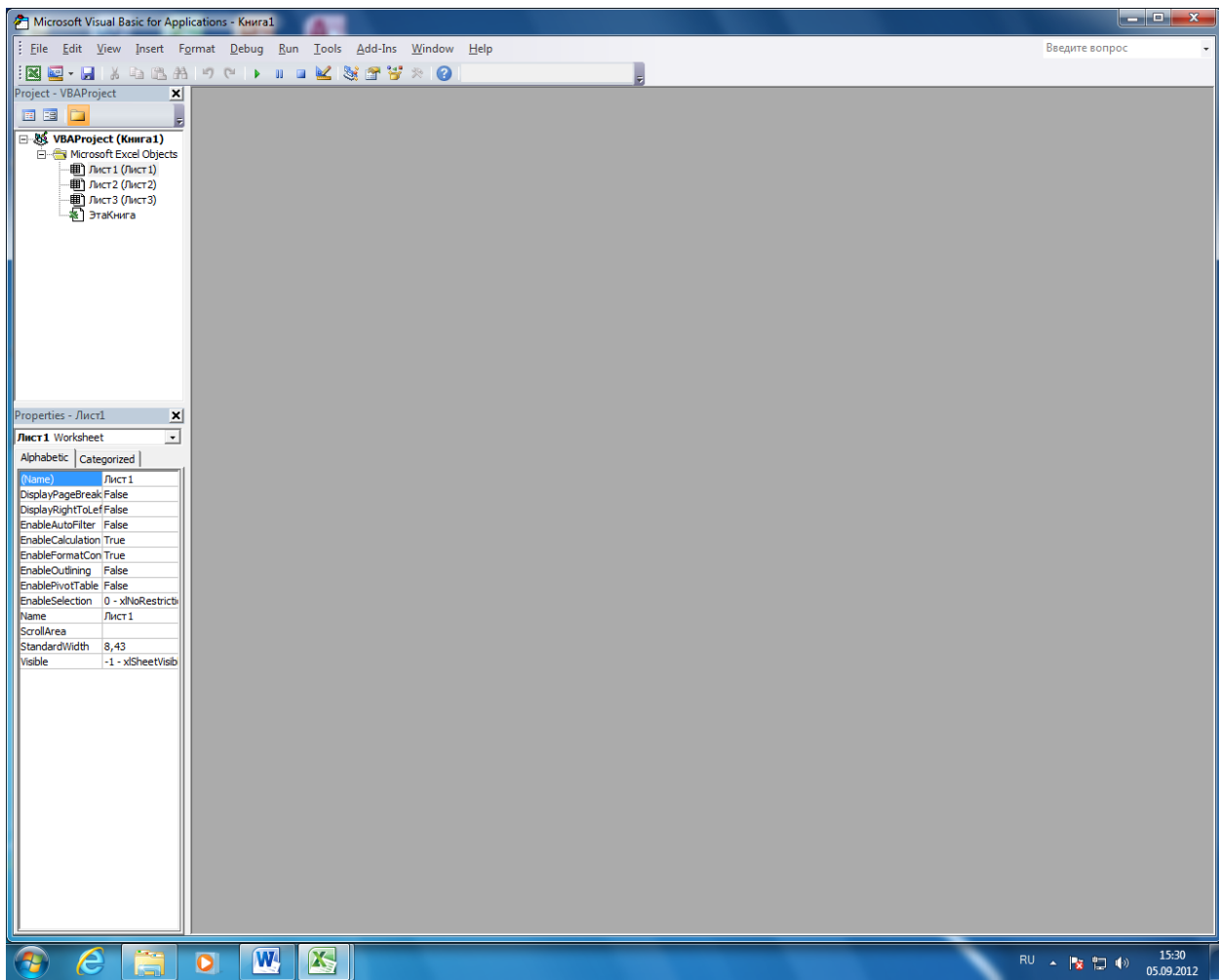
Ж

В главном меню *Excel* появится «Разработчик»



З

Раскрываем список **Разработчика**, вызываем *VBA*, далее работаем, как указано в методических материалах применительно к *Excel 2003*.



и

Рис. Пошаговая установка опции «Разработчик» в главном меню *Excel*