

Практикум по дисциплине
Технологии и методы программирования
(технологии VBA)

Общие сведения

Основные понятия языка программирования VBA.

Объект - это объединение данных с кодом, предназначенным для их обработки.

Объект представляет собой именованный элемент, имеющий:

- **свойства**, то есть характеристики, которые можно проверить или изменить;
- **методы**, то есть действия, которые можно выполнить над объектом;
- **события**, то есть возможные для объекта ситуации, на которые он может ответить заранее предопределенными действиями.

Оператор (инструкция) - это выполняемая единица VBA-кода. Оператор может объявлять или определять переменную или выполнять какое-либо действие в программе. Операторы в общем случае состоят из ключевых слов, выражений и операндов.

Ключевые (зарезервированные) слова - это имена команд, операторов, процедур, функций.

Операнды - это данные, используемые в программе. К ним относятся константы, переменные и значения функций.

Выражения - действия над данными, делятся на арифметические, логические, условные и строковые.

Процедура - это наименьшая единица программного кода, на которую можно ссылаться по имени и которая может выполняться независимо. В VBA основные типы процедур - это **Sub** и **Function**. Любая процедура содержит один или более операторов, помещенных между двумя специальными

операторами: объявлением процедуры в начале и оператором завершения процедуры в конце.

Модуль - это именованная единица, состоящая из одной или нескольких процедур, а также общих объявлений, относящихся ко всем процедурам в модуле. В VBA также присутствуют стандартные модули и модули класса, где пользователь может определять свои классы

Проект - эта часть программы, которая видна на экране при ее создании. Проект VBA имеет иерархическую структуру и включает: объекты, формы, стандартные модули, модули классов и модули форм. Например, объектами Excel, входящими в проект, являются рабочие книги (WorkBooks), рабочие листы (Worksheets), диаграммы (Charts). С каждым из этих объектов связан специальный модуль, в который может быть помещен программный код, выполняющий определенные действия.

В модулях форм записываются коды процедур обработки событий формы и элементов управления (объектов), размещенных на ней.

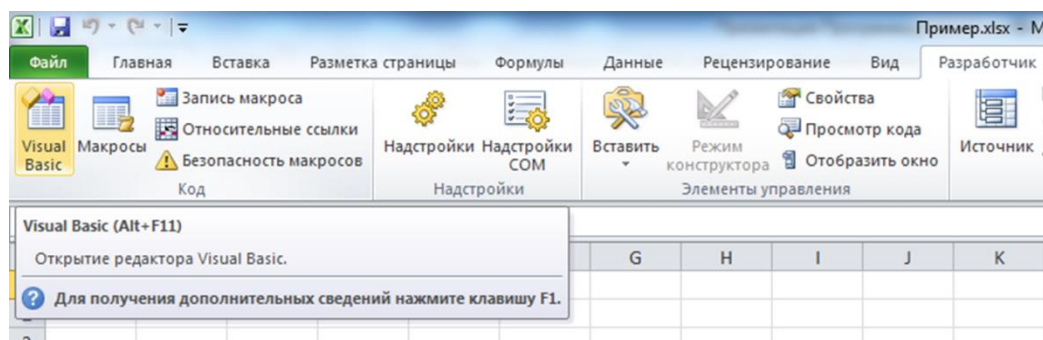
В модулях рабочих листов помещаются процедуры обработки событий рабочих листов и элементов управления, размещаемых на рабочих листах.

Проект включает две части: интерфейсную и программную, которая находится в различных модулях и реализует выполнение заданных разработчиком действий.

Весь проект, включая классы модулей, модули и пользовательские формы, сохраняется вместе с рабочей книгой в особом формате, т.е. типе файла MS Excel *.xlsm с поддержкой макросов. Модули можно сохранить посредством процедуры экспорта с целью их использования в других приложениях.

Практическое задание 1. Знакомство со средой.

1. Запустить MS Excel, в новой рабочей книге запустить VBA. (запустить VBA можно, нажав соответствующую кнопку на закладке ленты Разработчик или использовать комбинацию клавиш Alt+F11).



2. Вставить в проект новую форму – Insert → UserForm.

3. Изучить окно среды VBA, найти все области интерфейса среды VBA, представленные на рис.1.

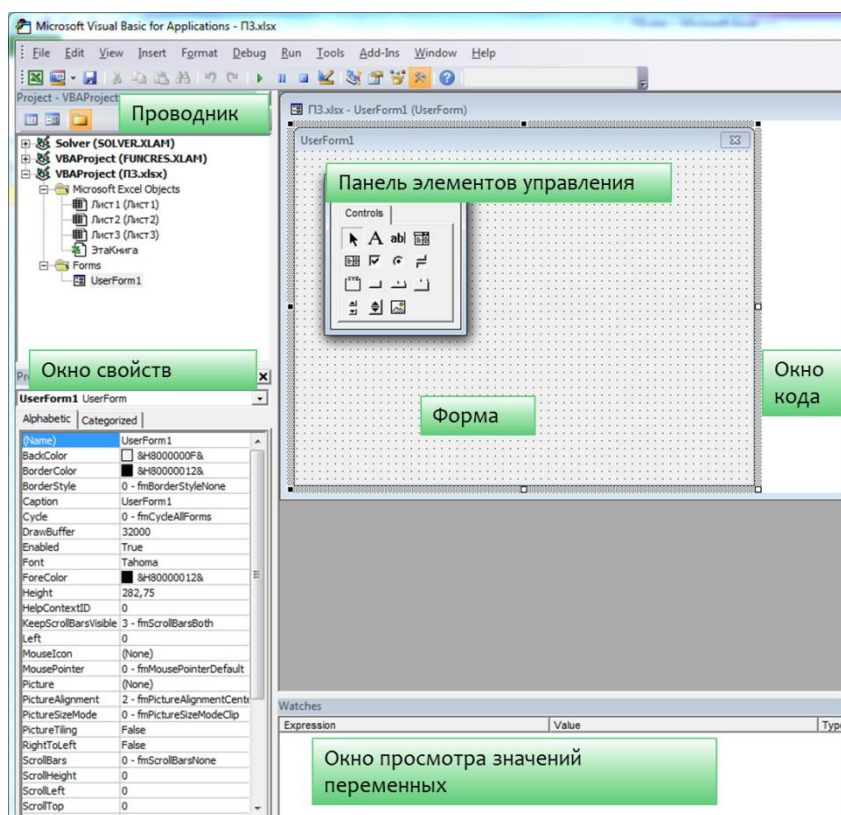


Рис. 1. Общий вид редактора VBE

Дключить отображение любых областей окна VBA можно в меню View (например Project Explorer - проводник проекта, Properties Window - окно свойств, ToolBox - окно панели инструментов, рис. 2).

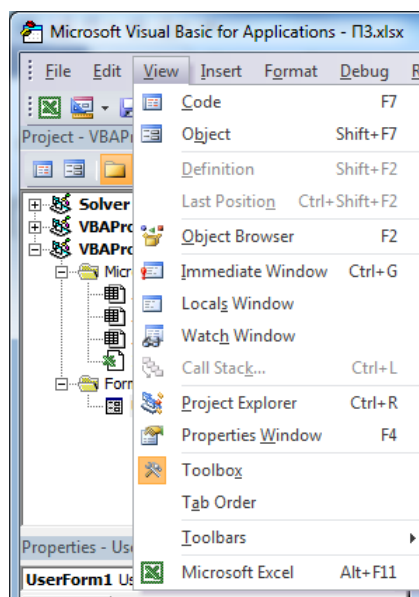


Рис. 2. Меню View

4. Найти в интерфейсе среды VBA:

4.1) панель инструментов ToolBox и разобрать способ работы с ней;

4.2) виды элементов управления (объектов) панели инструментов ToolBox, размещаемых на форме - CommandButton, Label, Textbox, Image;

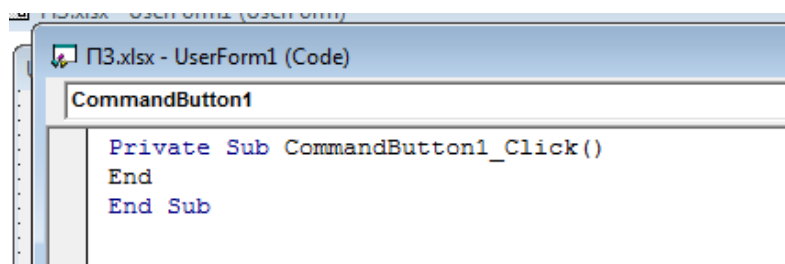
4.3) для указанных элементов управления (объектов) определить параметры указанных свойств:

Объект	Свойство
Form (Форма)	<i>Name</i> – имя объекта; <i>Caption</i> – текст, выводимый в строке заголовка; <i>Picture</i> – фоновый рисунок на форме, путь к файлу рисунка.
TextBox (Тестовое поле)	<i>Name</i> – имя объекта; <i>Text</i> – текст, выводимый в поле на экране; <i>Font</i> – настройка шрифта; <i>ForeColor</i> – цвет шрифта; <i>BackColor</i> – цвет фона в поле; <i>TextAlign</i> – выравнивание текста в поле.
Label (Метка)	<i>Name</i> – имя объекта; <i>Caption</i> – текст, выводимый в поле на экране; <i>BackColor</i> – цвет фона в поле; <i>BorderStyle</i> – Fixed Single (углубление текста в поле); <i>Font</i> – настройка шрифта; <i>ForeColor</i> – цвет шрифта; <i>TextAlign</i> – выравнивание текста в поле.
Image (Изображение)	<i>Name</i> – имя объекта; <i>Picture</i> – путь к файлу рисунка; <i>PictureSizeMode</i> – <i>ImageSizeModeStretch</i> – (заполнение отведенного окна).
CommandButton (Командная кнопка)	<i>Name</i> – имя объекта; <i>Caption</i> – текст, выводимый в поле на экране; <i>Font</i> – настройка шрифта; <i>ForeColor</i> – цвет шрифта; <i>AutoSize</i> – автоматический подбор размера объекта под текст; Для создания программного кода в виде подпрограммы вида: <i>Private Sub Command_Click() ...</i> <i>End Sub</i> , можно выполнить двойной щелчок мышью по объекту.

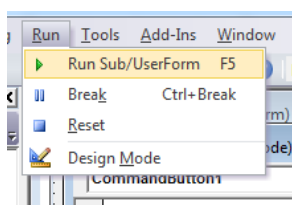
5. Изменить у формы UserForm1 свойство Caption (Имя) на Финуниверситет и разместить на ней элемент управления CommandButton1 и настроить параметры его свойств:

Свойство	Параметр
Caption	Закреть
AutoSize	True
BackColor	Желтый цвет из палитры

6. Создать программный код, связанный с элементом управления CommandButton1, выполнив двойной щелчок мышью по объекту. Ввести программный код для завершения работы с формой:

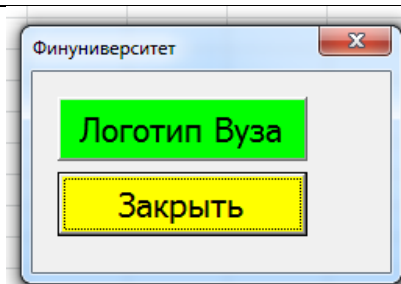


7. Запустить проект UserForm1 в среде VBA (в меню Run run или F5) и проверить работу командной кнопки «Заккрыть», размещенной на этой форме.



8. Добавить выше командной кнопки «Заккрыть» новый элемент управления CommandButton2 и настроить его свойства:

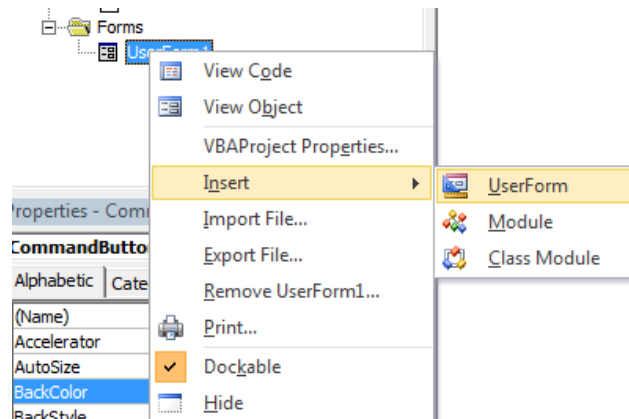
Свойство	Параметр
Caption	Логотип Вуза
AutoSize	True
BackColor	Зеленый цвет из палитры



9. Создать программный код, связанный с элементом управления CommandButton2, для перехода к новой форме2:

```
Private Sub CommandButton2_Click()
    UserForm2.Show
End Sub
```

10. Добавить в проект новую форму2,



перейти к ней, настроить ее свойства:

<i>Свойство</i>	<i>Параметр</i>
Caption	Логотип Вуза
BackColor	Голубой цвет из палитры

11. Разместить на форме2 элемент управления CommandButton1и настроить параметры его свойств:

<i>Свойство</i>	<i>Параметр</i>
Caption	Вернуться к форме Вуза
AutoSize	True
BackColor	Зеленый цвет из палитры

12. Создать программный код для элемента управления CommandButton1 для закрытия текущей формы и возврата к форме1:

```
CommandButton1
Private Sub CommandButton1_Click()
    Unload UserForm2
End Sub
```

13. На форму2 добавить элемент управления CommandButton2 для завершения работы проекта и написать для него программный код.

14. Перейти к форме1 (Финуниверситет) и запустить проект на выполнение (F5), проверить работу кнопки вызова формы2 проекта «Логотип Вуза», а на форме2 проверить работу кнопки «Вернуться к форме Вуза».

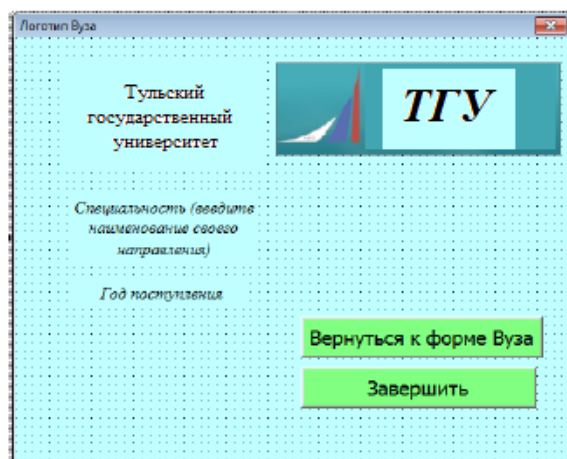
15. Завершить работу проекта.

16. Сохранить проект в своей папке в специальном формате (типе) рабочей книги MS Excel.xlsm (с поддержкой макросов) с именем «ФИО_Проект1».

17. Заккрыть VBA и MS Excel. Найти свой файл и открыть его, запустить VBA и продемонстрировать работу проекта преподавателю.

18. Перейти к форме2 вашего проекта и добавить на нее новые элементы управления, придерживаясь оформления, представленного на рисунке, и настроить для элементов управления следующие параметры свойств:

<i>Объект</i>	<i>Свойство</i>
Label1	<i>Caption</i> = Тульский государственный университет
Label2	<i>Caption</i> = Специальность (введите наименование своего направления)
Label3	<i>Caption</i> = Год поступления (введите текущий год)
Image1	<i>Picture</i> = путь\имя объекта



19. Настроить невидимость элементов управления Label (название вуза, название специальности/направления) и Image1, изменив их свойство Visible на False.

20. Добавить на форму2, выше командной кнопки «Вернуться к форме Вуза», новый элемент управления CommandButton3 «Показать скрытые поля», предназначенный для визуализации (отображения) невидимых объектов.

21. Для нового объекта CommandButton3 «Показать скрытые поля» создать процедуру обработки:

```
Private Sub CommandButton3_Click()  
    Label1.Visible = True  
    Label2.Visible = True  
    Label3.Visible = True  
End Sub
```

22. Запустить проект при активной форме2, отобразить невидимые на ней поля, проверив работу кнопки «Показать скрытые поля». Закрывать форму.

23. В среде VBA сохранить проект в свой папке с именем «ФИО_Проект1», не забывайте указать тип файла с поддержкой макросов.

24. Показать свою работу преподавателю, запустив проект на выполнение и демонстрируя работу командных кнопок, размещенных на форме.

25. Завершить работу с VBA и MS Excel.

Практическое задание 2. Элементы управления формы.

1. Открыть файл «ФИО_Проект1», сохраненный в практическом задании 1.

2. Запустить VBA и перейти к форме2.

3. Настроить на форме2 фоновый рисунок, настроив свойство формы:

<i>Объект</i>	<i>Свойство</i>
Form2	<i>Picture</i> = путь\имя объекта

4. Удалить с формы2 элемент управления Label с годом поступления.

5. Добавить на это место элемент управления TextBox1 для ввода года поступления, очистив, в случае необходимости его свойство Text.

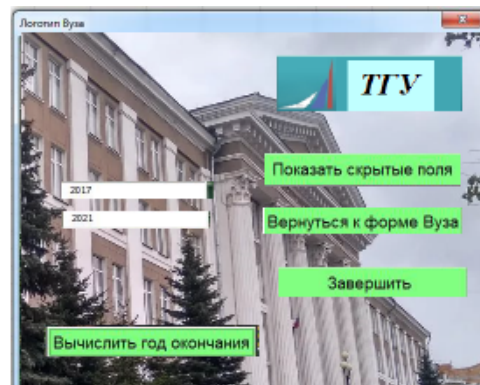
6. Добавить рядом с TextBox1 элемент управления TextBox2 для вывода рассчитанного года окончания, в случае необходимости очистить свойство Text.

7. Добавить выше элемента управления TextBox2 новый элемент управления CommandButton4 «Вычислить год окончания» и ввести программный код:

```
Private Sub CommandButton4_Click()  
    TextBox2.text=Val(TextBox1.text)+4  
End Sub
```

(Val – функция преобразования текста в числовой формат).

8. Запустить проект, отобразить невидимые поля, ввести дату в TextBox1 и выполнить расчет года окончания вуза. Заккрыть проект.



9. Добавить над элементом управления TextBox1 новый элемент управления CommandButton5 «Ввести год поступления» для ввода года поступления в специальном окне ввода.

```
Private Sub CommandButton5_Click()  
    Dim data As Single  
    data = Val(InputBox("Введите год поступления в вуз", _  
        "Год поступления"))  
    TextBox1.Text = data  
End Sub  
|
```

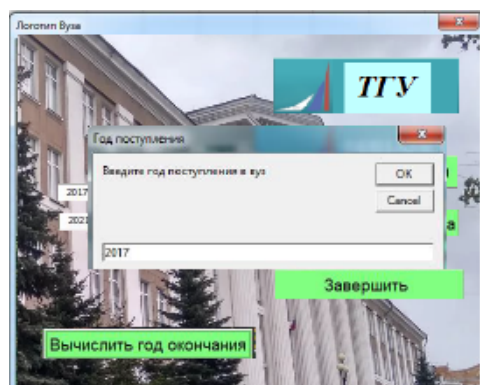
10. Добавить новый элемент CommandButton6 «Очистить даты» для очистки объектов TextBox с датами и для него ввести программный код:

```
Private Sub CommandButton6_Click()  
    TextBox1.Text = " "  
    TextBox2.Text = " "  
End Sub
```

11. Сохранить проект в свой папке с новым именем «ФИО_Проект №1» тип файла с поддержкой макросов.

12. Сделать активной форму1 и запустить проект. Кнопкой «Логотип вуза» перейти к форме2.

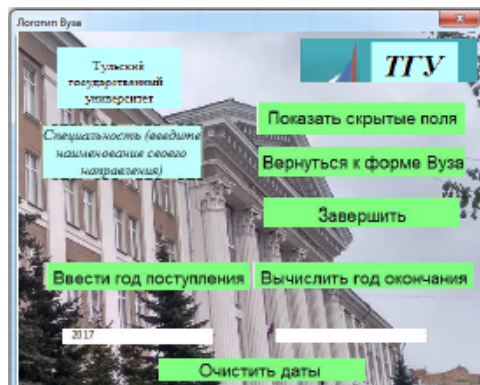
13. Отобразить невидимые поля на форме2. Использовать командную кнопку и ввести год вашего поступления в вуз.



Рассчитать год окончания вуза.

14. Очистить даты и ввести другой год поступления. Рассчитать год окончания вуза. Заккрыть проект.

15. Окончательный вид формы2 в проекте представлен на рисунке



Практическое задание 3. Ввод, вывод, вычисления в VBA.

1. Запустить MS Excel и среду VBA. Вставить в проект форму1.
2. Разместить на форме1 с помощью панели элементов ToolBox элементы управления, представленные на рисунке:
 - a) метки Label1, Label2, Label3;
 - b) текстовые поля TextBox1, TextBox2, TextBox3;
 - c) командные кнопки CommandButton1, CommandButton2 и CommandButton3, CommandButton4.
3. Установить указанные свойства для объектов:

Объект	Свойство
Form1	<i>Caption</i> =Вычисление числа ПИ с разной точностью
Label1	<i>Caption</i> = Число ПИ с одинарной точностью
Label2	<i>Caption</i> = Число ПИ с двойной точностью

Label3	<i>Caption</i> = Число ПИ, округленное до 4-х знаков после запятой
TextBox1	<i>Text</i> = (пустая строка)
TextBox2	<i>Text</i> = (пустая строка)
TextBox3	<i>Text</i> = (пустая строка)
CommandButton1	<i>Caption</i> = Вычисление с обычной точностью
CommandButton2	<i>Caption</i> = Вычисление с двойной точностью
CommandButton3	<i>Caption</i> = Вычисление с округлением
CommandButton4	<i>Caption</i> =Закреть

4. Выполнить разное цветовое оформление объектов по примеру рис.4:

5. Ввести программный код для командной кнопки «Вычисление с обычной точностью» (CommandButton1):

```

Private Sub CommandButton1_Click()
    Dim pi As Single
    pi = 4 * Atn(1)
    TextBox1.Text = pi
End Sub

```

6. Ввести программный код для командной кнопки «Вычисление с двойной точностью» (CommandButton2):

```

End Sub

Private Sub CommandButton2_Click()
    Dim pi As Double
    pi = 4 * Atn(1)
    TextBox2.Text = pi
End Sub

```

7. Ввести программный код для командной кнопки «Вычисление с округлением» (CommandButton3):

```

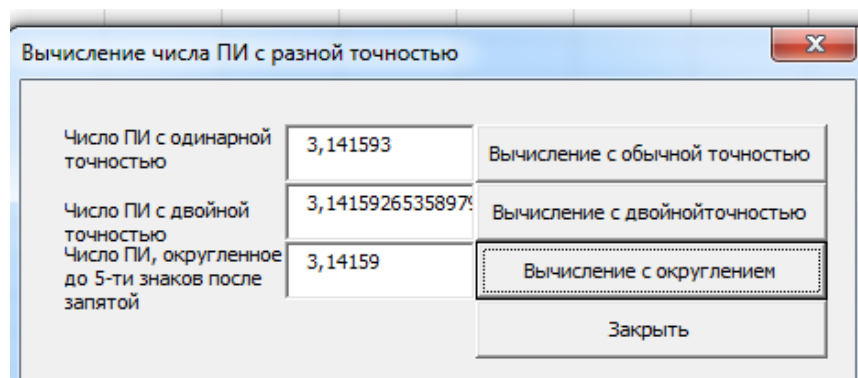
End Sub

Private Sub CommandButton3_Click()
    Dim pi As Single
    pi = Round(4 * Atn(1), 5)
    TextBox3.Text = pi
End Sub

```

8. Написать программный код для командной кнопки «Заккрыть».

9. Запустить проект на выполнение в среде VBA. Вычислить все значения числа π . Заккрыть проект.

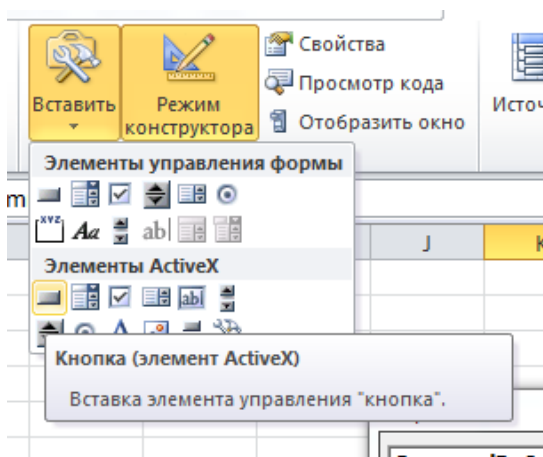


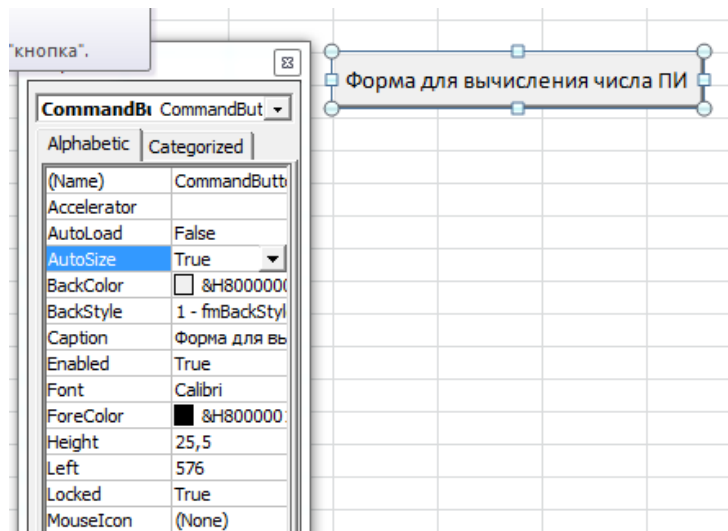
10. Свернуть окно VBA и перейти на лист1 рабочей книги MS Excel.

11. Перейти на закладку ленты Разработчик (если её нет, то в параметрах MS Excel включить её отображение).

12. На ленте включить режим Конструктора и с помощью кнопки Вставить разместить на листе1 элемент ActiveX – кнопку (CommandButton1) и настроить ее свойства:

Caption	Форма для вычисления числа ПИ
AutoSize	True (т.е. автоматический подбор размера кнопки)





14. Связать с кнопкой «Форма для вычисления числа ПИ» (CommandButton1) Процедуру:

CommandButton1

```
Private Sub CommandButton1_Click()  
    UserForm1.Show  
End Sub
```

15. Выключить режим конструктора и проверить на листе1 работу новой

кнопки, выполнить вычисления и закрыть проект (т.е. форму1).

16. Сохранить проект в рабочей книге с именем «Вычисление пи», тип файла

с поддержкой макросов в своей папке.

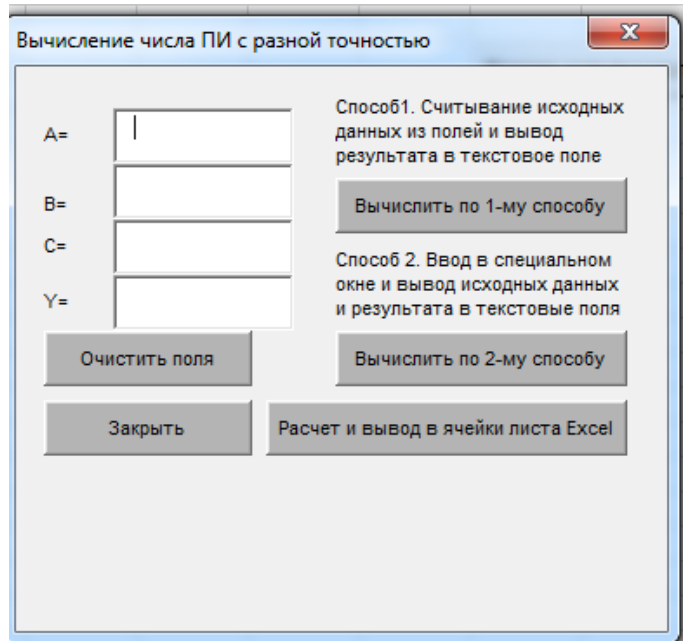
17. Завершить работу с VBA и закрыть рабочую книгу.

Практическое задание 4. Арифметические операции в VBA

Создать проект с пользовательской формой, в котором вычисляется значение арифметического выражения.

$$y = a + \sin^2(b) * 3^3 + \arctg(c/3)$$

1. Создайте новый проект
2. Разместите на форме1 элементы управления - метки, тестовые поля и командные кнопки и установите их свойства как на рисунке.



3. Сохраните проект в своей папке с новым именем «ФИО_Проект 4» с поддержкой макросов.
4. Введите программный код для командной кнопки «Вычислить по 1-му способу»:

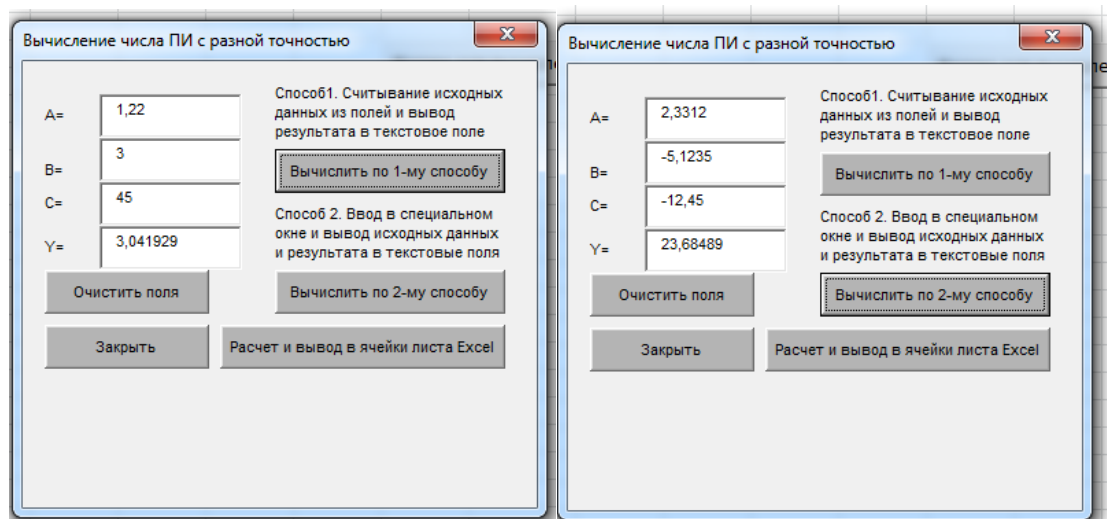
```
Private Sub CommandButton1_Click()
    Dim a As Single, b As Single, c As Single, y As Single
    a = Val(TextBox1.Text)
    b = Val(TextBox2.Text)
    c = Val(TextBox3.Text)
    y = a + (Sin(b)) ^ 2 * 3 ^ 3 + Atn(c / 3)
    TextBox4.Text = y
End Sub
```

5. Введите программный код для командной кнопки «Вычислить по 2-му способу»:

End Sub

```
Private Sub CommandButton3_Click()  
    Dim a As Single, b As Single, c As Single, y As Single  
    a = Val(InputBox("Введите a", "Ввод", "2.3312"))  
    b = Val(InputBox("Введите b", "Ввод ", "-5.1235"))  
    c = Val(InputBox("Введите c", "Ввод ", "-12.45"))  
    y = a + (Sin(b)) ^ 2 * 3 ^ 3 + Atn(c / 3)  
    TextBox1.Text = a  
    TextBox2.Text = b  
    TextBox3.Text = c  
    TextBox4.Text = y  
End Sub
```

6. Ввести программный код для командной кнопки «Очистить поля» (Как в предыдущем задании)
7. Написать программный код для командной кнопки «Закрыть».
8. Запустить проект с формой1 на выполнение и проверить работу двух способов, на контрольном примере, выполненном на рисунке. Для первого способа необходимо ввести в текстовые поля значения исходных данных. Закрыть форму.



9. Добавить для командной кнопки формы1 «Расчет и вывод в ячейки листа Excel» программный код:

```

End Sub

Private Sub CommandButton2_Click()
    a = Val(InputBox("Введите a", "Ввод", "2.3312"))
    b = Val(InputBox("Введите b", "Ввод ", "-5.1235"))
    c = Val(InputBox("Введите c", "Ввод ", "-12.45"))
    y = a + (Sin(b)) ^ 2 * 3 ^ 3 + Atn(c / 3)
    Cells(2, 1) = a 'вывод в первую ячейку второй строки листа
    Cells(2, 2) = b 'вывод во вторую ячейку второй строки листа
    Cells(2, 3) = c 'вывод в третью ячейку второй строки листа
    Cells(2, 4) = y 'вывод в четвертую ячейку второй строки листа
End Sub

```

10. Перейти в окне рабочей книги Excel на закладку ленты Разработчик.

11. На листе1 рабочей книги ввести данные в ячейки как на рисунке

	A	B	C	D	E	F	G	H
1	A	B	C	Y				
2								
3								
4								

Форма для вычисления выражения

12. Включить режим Конструктора и разместить на листе1 элемент ActiveX – кнопку1 (CommandButton1) и настроить её свойства:

Caption	Форма для вычисления выражения
AutoSize	True (т.е. автоматический подбор размера кнопки)

13. Связать с этой кнопкой программный код:

```

CommandButton1

Private Sub CommandButton1_Click()
    UserForm1.Show
End Sub

```

14. Выключить режим Конструктора, проверить работу кнопки2 «Форма для вычисления выражения», выполнить расчет и вывод в ячейки листа, используя соответствующую кнопку формы1. Закрыть форму1.

15. Просмотреть результат вывода на листе1 рабочей книги, он должен соответствовать рисунку ниже.

	A	B	C	D	E	F	G	H	I
A		B	C	D					
	2,3312	-5,1235	-12,45	23,68489	Форма для вычисления выражения				
					Расчет и вывод в ячейки листа Excel				

16. На закладке ленты Разработчик, включить режим Конструктора и разместить на листе1 еще один элемент ActiveX – кнопку2 (CommandButton2) и настроить её свойства:

Caption	Расчет и вывод в ячейки листа Excel
AutoSize	True (т.е. автоматический подбор размера кнопки)

17. Связать с этой кнопкой программный код, скопированный из программного кода кнопки «Расчет и вывод в ячейки листа Excel» формы1

18. Перейти из окна VBA на лист1 рабочей книги. Выключить режим конструктора.

19. Сохранить свой проект с текущим именем файла.

20. Используя кнопку листа1 «Форма для вычисления выражения» вызвать форму1, проверить работу ее командной кнопки «Расчет и вывод в ячейки листа Excel» используя, заданные по умолчанию, исходные данные для расчета.

21. Проверить, как изменился вид листа1 рабочей книги, он должен иметь вид, представленный на рисунке выше.

22. Удалите содержимое ячеек второй строки на листе1 рабочей книги Excel.

23. Проверьте работу кнопки листа1 «Вывести в ячейки рабочей книги» и сравните результат работы программного кода с результатом на рисунке.

24. Проанализируйте разницу в программных кодах всех способов ввода, вывода и разницу в результатах их выполнения.

25. Завершить работу с VBA и MS Excel.

Практическое задание 4. Работа со списками в VBA

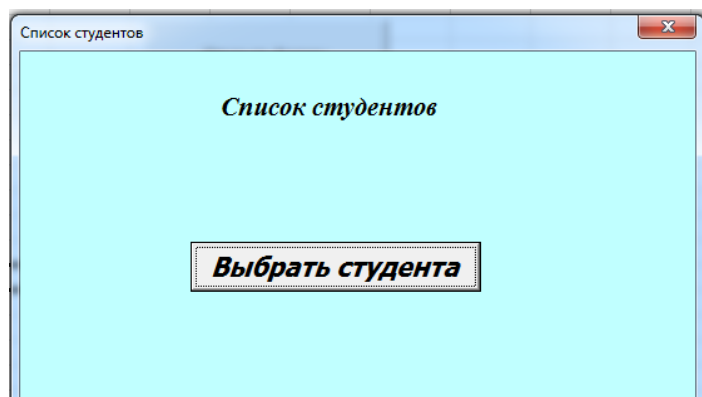
Список значений ListBox применяется для выбора значений переменной из списка.

Существует 3 вида списков:

- Простой список поддерживает выбор только одного элемента. Такой список напоминает группу переключателей, но позволяет более эффективно работать с большим числом элементов.
- Список связанного выбора позволяет выбрать один элемент, а также несколько расположенных рядом элементов.
- Список, разрешающий несвязный выбор нескольких строк, позволяет выбрать один элемент, расположенные рядом элементы, а также разрозненные элементы.

1. Запустить MS Excel и среду VBA. Вставить в проект форму1.

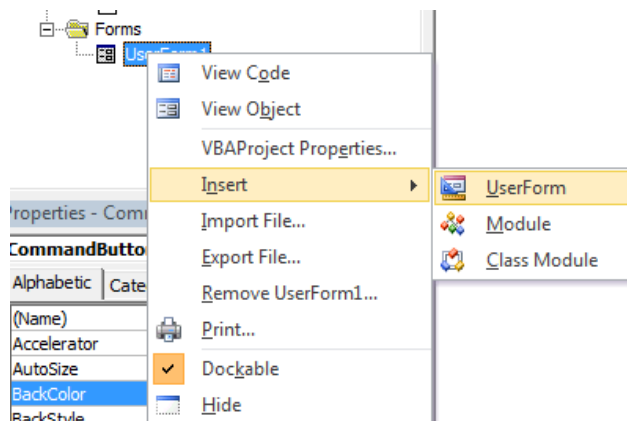
2. Разместить на форме1 с помощью панели элементов ToolBox элементы управления Label и CommandButton1, представленные на рисунке:



При нажатии на кнопку должен произойти переход на форму поиска студента по следующему коду:

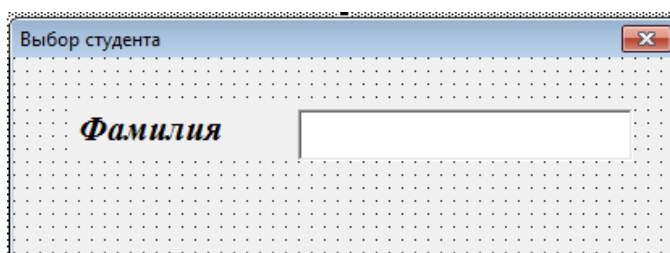
```
CommandButton1
Private Sub CommandButton1_Click()
    UserForm3.Show
End Sub
```

3. Добавить в проект новую форму3,



перейти к ней, настроить ее свойства:

4. С помощью панели инструментов разместить Label и ListBox



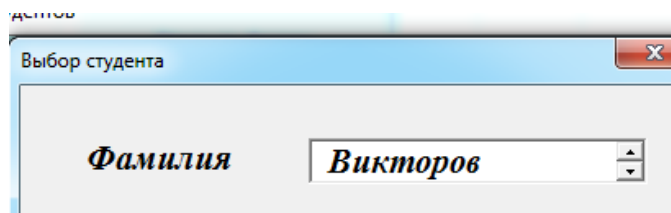
5. Заполните ListBox следующими значениями. Для этого создать новый модуль и записать в нем следующую процедуру.

```
General)
Public Sub sp()
    With UserForm3.ListBox1
        .List = Array("Абрамов", "Букин", "Викторов", "Настасьев", "Ромашкин", "Яблочкин")
        .ListIndex = 1
    End With
End Sub
```

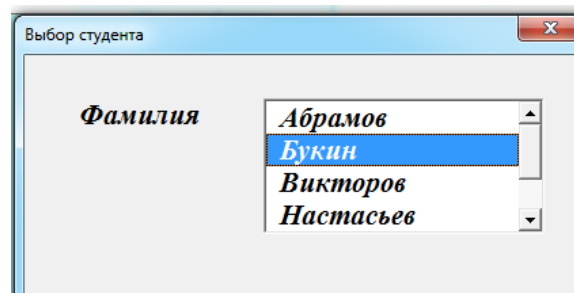
6. В форме1 для CommandButton1 дополнить строкой Module5.sp

```
CommandButton1
Private Sub CommandButton1_Click()
    sp
    UserForm3.Show
End Sub
```

7. Запустите Форму1, перейдите в форму 2 и оцените результат записи значений, используя скорллинг.



8. Расширьте ListBox1 и посмотрите результат выполнения.



9. Рядом установите кнопку ComboBox1.

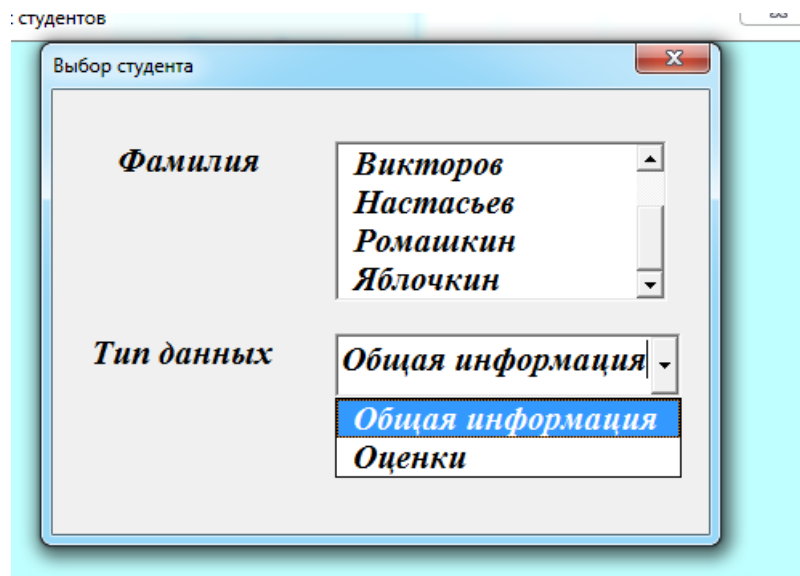
В модуле записать следующую процедуру:

```
Public Sub sp1()  
    With UserForm3.ComboBox1  
        .AddItem "Общая информация"  
        .AddItem "Оценки"  
        .ListIndex = 0  
    End With  
End Sub
```

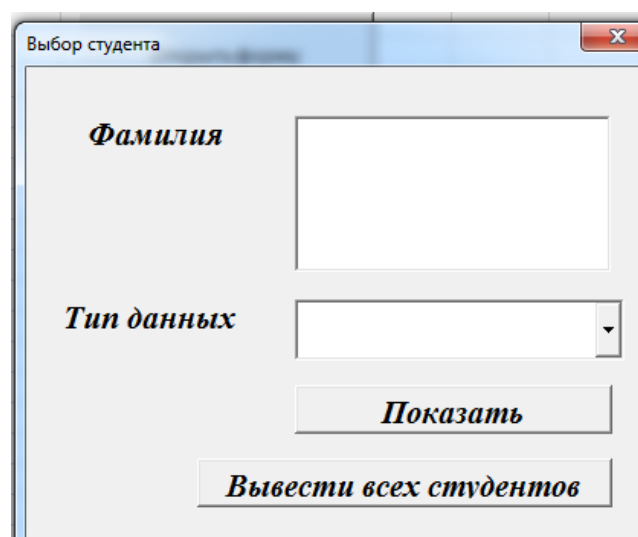
И добавить ее в процедуру CommandButton1_Click()

```
Private Sub CommandButton1_Click()  
    sp  
    sp1  
    UserForm3.Show  
End Sub
```

Запустите форму 1 и посмотрите результат



10. В форме1 создайте кнопку CommandButton1, которая будет выдавать результаты в зависимости от выбора пользователя и CommandButton2 для вывода общего списка.



11. Создайте новый лист с именем «Список студентов» с помощью процедуры:

```

End Sub

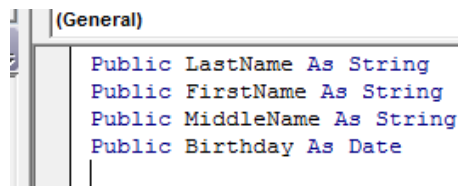
Public Sub AddWSH()
    Workbooks("Пример.xlsm").Activate
    ActiveWorkbook.Worksheets.Add.Name = "Список студентов"
End Sub

```

12. Заполните лист с помощью следующей процедуры:

```
Public Sub Stud()  
    ActiveWorkbook.Worksheets("Список студентов").Activate  
    ActiveSheet.Cells(1, 1).Value = "Ип/п"  
    ActiveSheet.Cells(1, 2).Value = "Фамилия"  
    ActiveSheet.Cells(1, 3).Value = "Имя"  
    ActiveSheet.Cells(1, 4).Value = "Отчество"  
    ActiveSheet.Cells(1, 5).Value = "Год рождения"  
    ActiveSheet.Cells(1, 6).Value = "Оц. Матем"  
    ActiveSheet.Cells(1, 7).Value = "Оц. Ин.яз"  
    ActiveSheet.Cells(1, 8).Value = "Оц. Прогр."  
    For i = 2 To 7  
        ActiveSheet.Cells(i, 1).Value = i - 1  
    Next  
End Sub
```

Создайте пользовательский тип Student. Для этого создайте модуль класса и напишите в нем следующий код программы:



```
(General)  
  
Public LastName As String  
Public FirstName As String  
Public MiddleName As String  
Public Birthday As Date
```

Для ввода записи первого студента напишите следующую процедуру

```
Public Sub Stud1()  
    Dim var1 As Students ' Объявляем переменную  
    Set var1 = New Students ' Присваиваем переменной новый объект  
    var1.LastName = "Абрамов" ' Далее задаем значения  
    var1.FirstName = "Иван"  
    var1.MiddleName = "Тристанович"  
    var1.Birthday = "26.07.98"  
  
    ActiveSheet.Cells(2, 2).Value = var1.LastName  
    ActiveSheet.Cells(2, 3).Value = var1.FirstName  
    ActiveSheet.Cells(2, 4).Value = var1.MiddleName  
    ActiveSheet.Cells(2, 5).Value = var1.Birthday  
  
End Sub
```

Посмотрите на результат.

Для ввода остальных студентов создадим следующую формулу:

Установим TextBox5 значение Текст = 2 и сделаем его невидимым Visible=False.

TabStop	True
Tag	
Text	3
TextAlign	1 - fmTextAlignLeft
Top	18
Value	3
Visible	False
Width	30
WordWrap	True

13. Повесим на CommandButton1 следующую процедуру.

```
Private Sub CommandButton1_Click()
    Dim var1 As Students ' Объявляем переменную
    Dim i As Integer
    Set var1 = New Students ' Присваиваем переменной новый объект
    var1.LastName = UserForm4.TextBox1 ' Далее задаем значения
    var1.FirstName = UserForm4.TextBox2
    var1.MiddleName = UserForm4.TextBox3
    var1.Birthday = CDate(UserForm4.TextBox4)
    i = CInt(UserForm4.TextBox5)
    ActiveSheet.Cells(i, 2).Value = var1.LastName
    ActiveSheet.Cells(i, 3).Value = var1.FirstName
    ActiveSheet.Cells(i, 4).Value = var1.MiddleName
    ActiveSheet.Cells(i, 5).Value = var1.Birthday
    UserForm4.TextBox5 = i + 1
End Sub

Private Sub CommandButton2_Click()
    End
End Sub
```

14. Запустим форму и введем соответствующие значения:

Посмотрим на результат.

	A	B	C	D	E	F
1	№п/п	Фамилия	Имя	Отчество	Год рождения	Оц. Матер.
2	1	Абрамов	Иван	Тристанов	26.07.1998	
3	2	Букин	Сергей	Иванович	01.05.1997	
4	3	Викторов	Арсен	Гурамович	05.06.1998	
5	4	Настасьев	Петр	Климович	15.07.1999	
6	5	Ромашкин	Илья	Петрович	25.03.1998	
7	6	Яблочкин	Семен	Фархатович	13.02.1996	
8						

15. Откроем форму 3 и кнопку «Вывести весь список» свяжем со следующей процедурой:

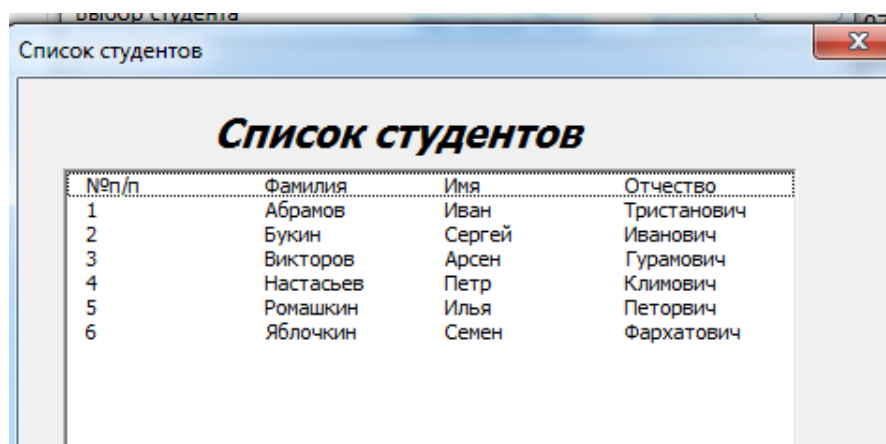
```
CommandButton2

Private Sub CommandButton2_Click()
    Module5.Spis1
    UserForm5.Show
End Sub
```

Где

```
Public Sub Spis1()
    With UserForm5.ListBox1
        .ColumnCount = 8
        .RowSource = "A1:H7"
        .ControlSource = "I1"
        .BoundColumn = 0
    End With
End Sub
```

В результате получим:



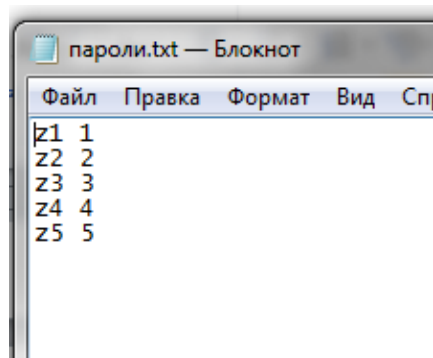
Практическое задание 5. Работа с файлами. Организация доступа к форме в VBA с использованием файла паролей

Справка:

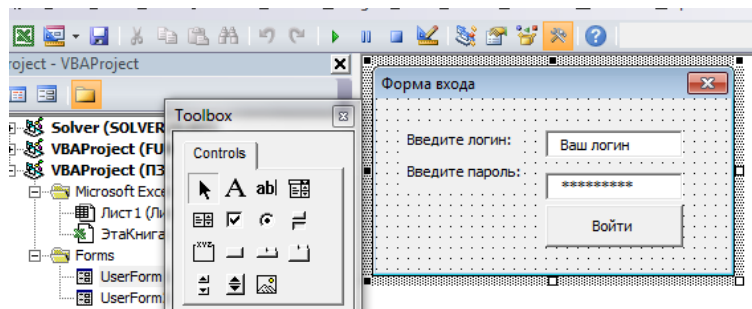
Input #	<p>Считывает данные из открытого файла последовательного доступа и присваивает их переменным. Данные, считываемые с помощью инструкции input, обычно записываются в файл с помощью инструкции Write #.</p> <p>Синтаксис: Input #НомерФайла, СписокПеременных номерФайла — номер файла СписокПеременных — список переменных, которым следует присвоить значения, считанные из файла. Переменные в списке разделяются запятыми</p>
Line Input #	<p>Считывает строку из открытого файла последовательного доступа и присваивает ее переменной типа string. Данные, считываемые с помощью инструкции Line input #, как правило, записываются в файл с помощью инструкции Print #.</p> <p>Синтаксис: Line Input #НомерФайла, ИмяПеременной Синтаксис инструкции Line Input# содержит следующие элементы: НомерФайла — номер файла ИмяПеременной — имя переменной типа Variant или String</p>
EOF	Функция возвращает значение True при

	<p>достижении конца файла. Синтаксис: EOF (НомерФайла} При последовательном считывании информации из файла часто используется следующий цикл: Do While Not EOF(1) Loop или, для тех пользователей, кто предпочитает инструкцию While – Wend инструкции Do While - Loop, следующий эквивалентный цикл: While Not EOF (1) Wend</p>
--	---

1. Создайте текстовый файл с логинами и паролями



2. Создайте форму входа



3. Для того чтобы ввод пароля был скрыт ***** добавьте в свойство PasswordChar TextBox *

MousePointer	0 - fmMousePointerDefault
MultiLine	False
PasswordChar	*
ScrollBars	0 - fmScrollBarsNone
SelectionMargin	True
SpecialEffect	2 - fmSpecialEffectSunken

4. Свяжите кнопку Войти со следующей процедурой:

```

CommandButton1
Private Sub CommandButton1_Click()
    Dim logpass As String
    Dim logpass_f As String
    Dim s As Boolean
    Dim i As Integer
    s = False
    logpass = TextBox1.Text & " " & TextBox2.Text
    Open "D:\Рома\пароли.txt" For Input As #1
    Do While Not EOF(1)
        Line Input #1, logpass_f
        If logpass = logpass_f Then
            MsgBox ("Вы вошли")
            Unload UserForm1
            UserForm2.Show
            s = True
        End If
    Loop
    If s = False Then
        MsgBox ("Неправильный логин или пароль, повторите попытку")
    End If
    Close #1
End Sub

```

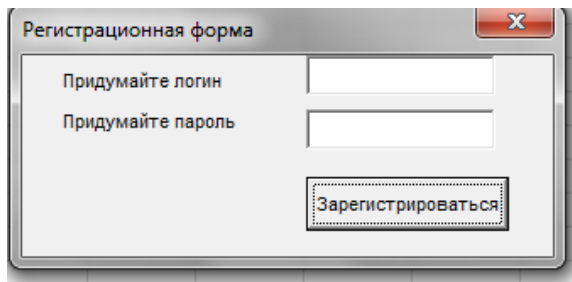
5. Создайте форму 2 произвольного вида

6. Проверьте работоспособность процедуры

Практическое задание 6. Работа с файлами. Форма регистрации

Создайте форму регистрации, которая бы записывала данные в файл в зашифрованном виде

1. Создайте форму следующего вида



2. Создайте пустой текстовый файл .txt

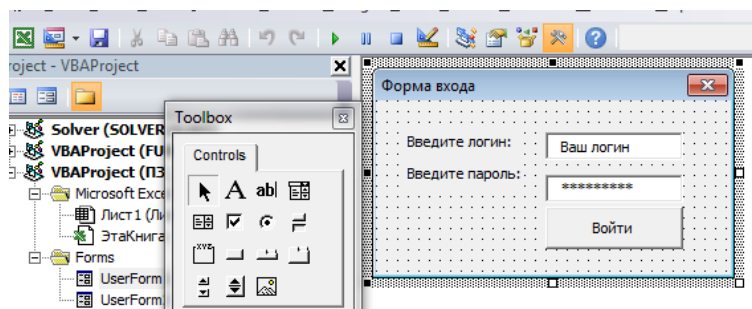
3. Зашифруйте вводимые данные.

Справка. Для того чтобы зашифровать данные, можно использовать один из алгоритмов шифрования, например, изменение кодировки . ASC()— эта функция позволяет вернуть числовой код для переданного символа. Например, ASC("D") вернет 68. Эту функцию удобно использовать для того, чтобы определить следующую или предыдущую букву. Обычно она используется вместе с функцией Chr(), которая производит обратную операцию — возвращает символ по переданному его числовому коду.

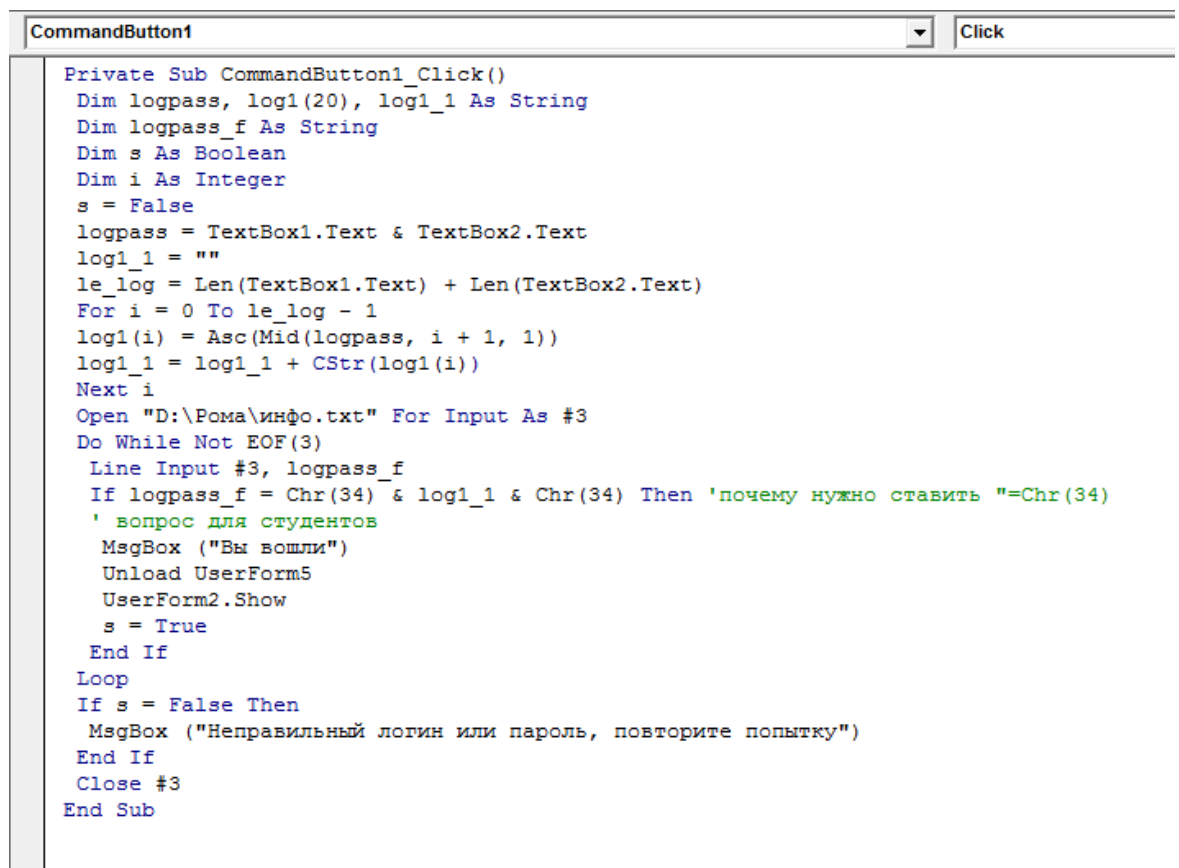
Введите следующий код программы.

```
Private Sub CommandButton1_Click()  
    Dim log, pass, log1(20), pass1(20), log1_1, pass1_1 As String  
    Dim i As Integer  
    log = TextBox1.Text  
    pass = TextBox2.Text  
    log1_1 = ""  
    pass1_1 = ""  
    le_log = Len(TextBox1.Text)  
    le_pass = Len(TextBox2.Text)  
    ' открываем файл для записи  
    Open "D:\Рома\инфо.txt" For Append As #2  
  
    For i = 0 To le_log - 1 ' шифрование логина  
        log1(i) = Asc(Mid(log, i + 1, 1))  
        log1_1 = log1_1 + CStr(log1(i))  
    Next i  
    For i = 0 To le_pass - 1 ' шифрование пароля  
        pass1(i) = Asc(Mid(pass, i + 1, 1))  
        pass1_1 = pass1_1 + CStr(pass1(i))  
    Next i  
    Write #2, log1_1 & pass1_1  
    Close #2 ' закрываем файл  
    MsgBox ("Вы зарегистрированы, зайдите под своей учетной записью")  
    Unload UserForm4 ' закрываем форму  
    UserForm5.Show ' открываем форму входа  
End Sub
```

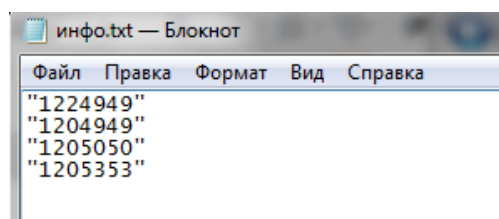

4. Создадим аналогичную форму входа, как в предыдущем задании.

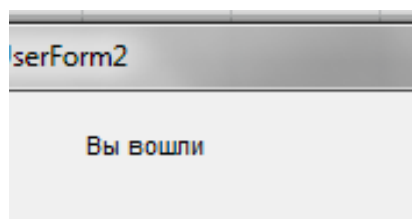
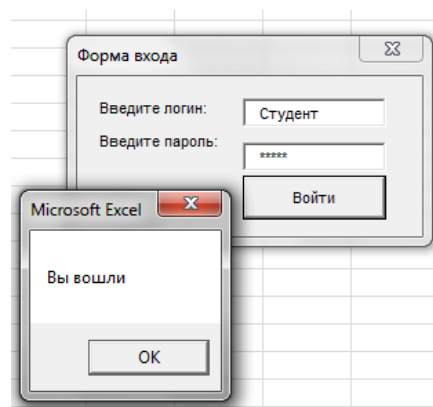
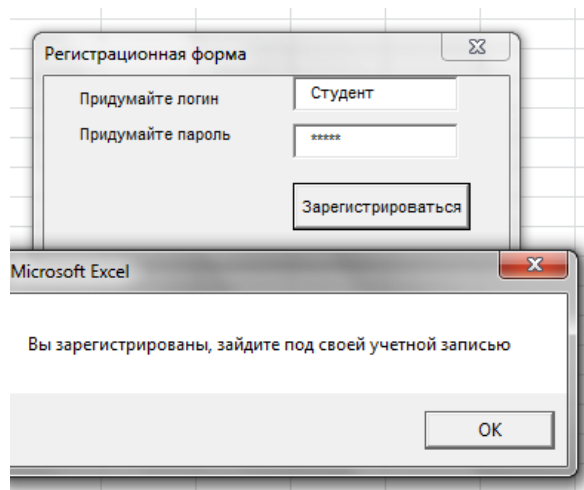


5. Видоизменим код обработчика кнопки войти:

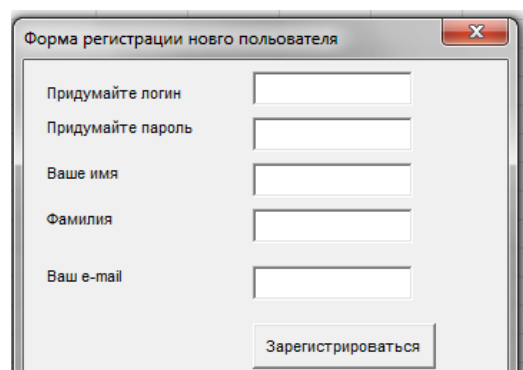


6. Протестируйте работу программы





7. Видоизмените форму регистрации как показано на рисунке и измените цветовое решение



8 (со звездочкой). Требуется чтобы после регистрации все данные были занесены в файл в зашифрованном виде, а при входе в форму 2 В ней отображалось: «Добро пожаловать Имя и Фамилия пользователя»

9 (со звездочкой). Измените алгоритм шифрования по своему усмотрению.

Практическое задание 7. Решение задач по пройденным темам

Создание функций в окне форм

1. Написать функцию Func_Z, которая принимает три целых числа в качестве параметров и возвращает целое число равное удвоенной сумме переданных параметров. Запустить функцию Func_Z из процедуры Proba_2. Слагаемые в процедуре должны вводиться при помощи окна InputBox. Результат должен быть выведен при помощи окна MsgBox

2. Написать функцию ZZZ, которая принимает четыре целых числа в качестве параметров и возвращает целое число равное удвоенной сумме первых двух чисел за вычетом утроенной суммы вторых двух чисел, которые были переданы в качестве параметров. Слагаемые должны вводиться при помощи окна InputBox. Результат должен быть выведен при помощи окна MsgBox

3. Написать функцию Func_33, которая принимает два числа с плавающей точкой в качестве параметров и возвращает число с плавающей точкой, равное удвоенной сумме переданных параметров, деленной на 33. Слагаемые должны вводиться при помощи окна InputBox. Результат должен быть выведен при помощи окна MsgBox

4. Написать функцию Func_Z1, которая принимает два целых числа и одно дробное в качестве параметров и возвращает число равное удвоенной сумме переданных параметров. Запустить функцию Func_Z1 из процедуры

Proba_24. Слагаемые в процедуре должны вводиться при помощи окна InputBox. Результат должен быть выведен при помощи окна MsgBox

5 (*). Написать функцию Func_X3, которая принимает в качестве параметров одну булеву переменную и три целых числа и возвращает целое число равное удвоенной сумме переданных целочисленных параметров, если первый пара-метр равен TRUE. Во всех других случаях функция должна возвращать -1. Функцию Func_X3 необходимо вызвать из процедуры Proba_2. Значения пара-метров должны вводиться в процедуре при помощи окна InputBox. Значение логической переменной должно быть введено в текстовом режиме словами «ИСТИНА», «ЛОЖЬ». При вводе должна осуществляться проверка адекватно-сти введенных данных требуемым условиям. Результат должен быть выведен при помощи окна MsgBox

6 (*). Написать функцию F_X1, которая принимает в качестве параметров две дробные переменные и одну булеву переменную и возвращает целое число равное удвоенной сумме, переданных дробных параметров поделенной на три, если логический параметр равен FALSE. Во всех других случаях функция должна возвращать -1111. Запустить функцию F_X1 из процедуры Proba_33. Значения параметров должны вводиться в процедуре при помощи окна InputBox. Значение логической переменной должно вводиться в текстовом режиме слова-ми «ИСТИНА», «ЛОЖЬ». При вводе должна осуществляться проверка адек-ватности введенных данных. Результат должен быть выведен при помощи окна MsgBox

Создание функций на рабочем листе

1. Написать функцию вычисляющую утроенную сумму двух целых чисел, которые передаются в функцию качестве параметров. Используя стандартный вызов функции с рабочего листа рабочей книги MS Excel, обеспечить вычис-ление утроенных сумм чисел, помещенных в ячейки

«A13» и «B17» рабочего листа «Лист1». Результат вычислений должен помещаться в ячейку «C22»

2. Написать функцию, вычисляющую утроенную сумму, деленную на два трех дробных чисел, которые передаются в функцию качестве параметров. Используя стандартный вызов функции с рабочего листа рабочей книги MS Excel, обеспечить вычисление искомым сумм чисел, помещенных в ячейки «A13» , «B17» и C19 рабочего листа «Лист2». Результат вычислений должен помещаться в ячейку «C17» рабочего листа «Лист1»

Работа с файлами

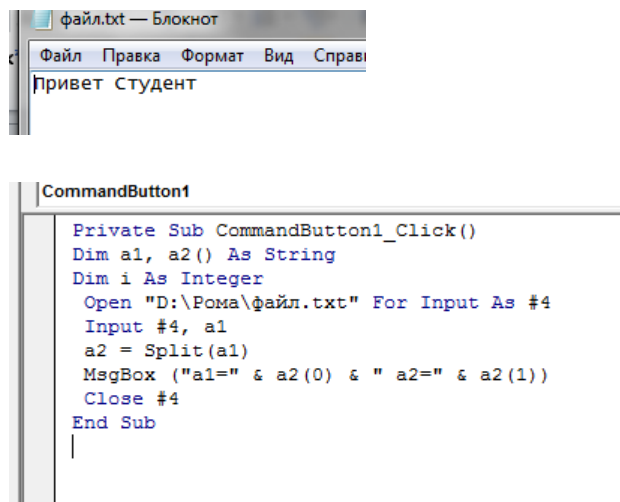
1. Вручную создать текстовый файл input.txt, в первую строку через пробел записать два числа. На языке VBA написать программу, считывающую первое число в переменную X, второе число – в переменную Y. Вычислить $Z=X^2+Y-14$. Результат записать в ячейку B12 листа Result того же файла, в котором на-писана программа

Справка. Для считывания данных построчно с разделителями используется команда split

Split (Expression,[Delimiter],[Limit],[Compare])

Параметр	Описание	Значение по умолчанию
Expression	Строка, содержащая подстроки и разделители	Нет
Delimiter	Разделитель, представляющий один или более символов	Пробел
Limit	Максимальное число подстрок, на которые должна быть разделена входная строка	-1
Compare*	Определяет, какое используется сравнение, двоичное - CompareMethod.Binary (0) или текстовое - CompareMethod.Text (1)	0

Пример



2. Вручную создать текстовый файл input1.txt, в первую строку через пробел записать три числа. На языке VBA написать программу, считывающую первое число в переменную X1, второе число – в переменную X2, третье число – в переменную X3 . Вычислить $Z=X1^2+X2-14*X3$. Результат записать в ячейку B1 листа Res того же файла, в котором написана программа

3(*). Вручную создать текстовый файл input.txt, в первую строку через пробел записать три числа, во вторую строку – через два пробела еще три числа. На языке VBA написать программу, считывающую числа из первой строки в массив X(3), числа из второй строки – в массив Y(3). Вычислить $Z=Y(2)*X(0)^2+Y(0)-14*X(2)$. Результат записать в файл, имя которого записано на в ячейке B12 листа Nastr того же файла, в котором написана программа.

4(*). Вручную создать текстовый файл inp.txt, в первую строку через три пробела записать четыре числа, во вторую строку – через два пробела еще четыре числа. На языке VBA написать программу, считывающую числа из первой строки в массив X(4), числа из второй строки – в массив Y(4). Вычислить $Z=Y(2)*X(0)^2+Y(0)-14*X(2)+2*X(3)+Y(1)$. Результат записать в

файл, имя которого записано на в ячейке A2 листа Лист1 того же файла, в котором написана программа.

5. На языке VBA написать программу, считывающую число из ячейки A1 листа Лист2 в переменную X, число из ячейки B3 того же листа – в переменную Y. Вычислить $Z=X^2+Y*43+15$. Результат записать в файл output.txt

6. На языке VBA написать программу, считывающую число из ячейки B3 листа Sheet2 в переменную X, число из ячейки B3 того же листа – в переменную Y. Вычислить $Z=X^2+Y*3+15$. Результат записать в файл output.txt

7. В текстовый файл MMZ.txt вручную через пробел занести три двузначных числа. Написать процедуру ZX(), которая открывает, текстовый файл с именем, получаемым из стандартного диалогового окна открытия файла (MMZ.txt), считывает искомые три числа, вычисляет их утроенную сумму и выводит результат в выходной файл с именем RESULT.txt и в окно сообщений MsgBox.

Работа со списками

1. В текстовый файл MMZ1.txt вручную через пробел занести четыре двузначных числа. Написать процедуру Z2X(), которая открывает, текстовый файл с именем, выбираемым из элемента управления ComboBox (в список имя файла MMZ1.txt должно быть занесено при инициализации формы). После открытия из файла должны быть считаны искомые четыре двузначных числа, вычислена их утроенная сумма, а результат выведен в выходной файл с именем RESULT1.txt и в окно сообщений MsgBox

2. В текстовый файл M.txt вручную через пробел занести три двузначных числа. Написать процедуру ZX(), которая открывает, текстовый

файл с именем, выбираемым из элемента управления ComboBox (в список имя файла M.txt должно быть занесено при инициализации формы). После открытия из файла должны быть считаны искомые три двузначных числа, вычислена разность первых двух чисел, умноженная на третье число, а результат выведен в выходной файл с именем RESULT13.txt и в окно сообщений MsgBox

3. Открыть текстовый файл, в котором записана одна строка, содержащая два числа и еще одна строка, содержащая одно число. Считать эти три числа, вычислить их утроенную сумму, вывести результат в выходной файл и в окно сообщений MsgBox. Имя выходного файла должно быть задано с использованием процедуры InputBox

4. Написать функцию Func_Z, которая принимает три числа с плавающей точкой в качестве параметров и возвращает целое число равное целой части удвоенной суммы переданных параметров. Запустить функцию Func_Z из процедуры Proba_2. Слагаемые должны вводиться при помощи окна InputBox. Результат должен быть выведен при помощи окна MsgBox и записан в ячейку A17 рабочего листа. Имя листа необходимо запросить с использованием элемента управления TextBox1 формы UF1

5. Написать функцию Function_Z1, которая принимает два числа с плавающей точкой в качестве параметров и возвращает целое число равное целой части утроенной суммы переданных параметров. Запустить функцию Function_Z1 из процедуры Proba_1. Слагаемые должны вводиться при помощи окна InputBox. Результат должен быть выведен при помощи окна MsgBox и записан в ячейку B21 рабочего листа. Имя листа необходимо запросить с использованием элемента управления TextBox1 формы UserF1

Работа с листом и формами, циклы

1. На заданном рабочем листе заданной рабочей книги создать таблицу заказов, содержащую информацию о пяти товарах: Название товара, количество, цена, единицы измерения, сумма. Написать макрос, который выдает отчет о сделанных покупках с использованием окна MsgBox. Отчет должен формироваться на основании информации, содержащейся в таблице заказов.

2. На заданном рабочем листе заданной рабочей книги создать таблицу заказов, содержащую информацию о пяти товарах. Написать макрос, который в виде периодически появляющихся сообщений выдает отчет о сделанных покупках. Отчет должен формироваться на основании информации, содержащейся в таблице. При формировании отчета необходимо использовать цикл FOR и окно стандартного вывода MsgBox.

3. На заданном рабочем листе заданной рабочей книги создать таблицу заказов, содержащую информацию о пяти товарах: Название товара, количество, цена, единицы измерения, сумма. Написать макрос, который выдает отчет о сделанных покупках с использованием элемента TextBox пользовательской формы. После вывода последней строки отчета форма должна закрыться. Отчет должен формироваться на основании информации, содержащейся в таблице заказов.

4. На заданном рабочем листе заданной рабочей книги создать таблицу заказов, содержащую информацию о пяти товарах: Название товара, количество, цена, единицы измерения, сумма. Написать макрос, который выдает отчет о сделанных покупках с использованием элемента ListBox1 пользовательской формы. Название листа, на который помещена форма, необходимо получить из элемента ComboBox1 той же формы. После вывода последней строки отчета форма должна закрыться. Отчет должен формироваться на основании информации, содержащейся в таблице заказов. Содержимое в ComboBox должно быть внесено при инициализации формы.

Работа с условиями

1. Создать процедуру Proba_X1, которая запускается при открытии рабочей книги и в окне InputBox запрашивает имя студента. Если имена: Василий, Елена, Михаил или Семен, то в окне сообщения должно появиться поздравление с началом зимней сессии. Если имена: Мария, Марина, Маргарита и Самуил, то в окне сообщения должно появиться поздравление с окончанием зимней сессии. В остальных случаях должно быть сообщение: «Добро пожаловать».

2. Создать процедуру Proba_X1, которая запускается при открытии рабочей книги и в окне InputBox запрашивает имя студента. Затем, если сегодня четное число, то в окне сообщения должно появиться персональное поздравление студента с окончанием летней сессии, если число нечетное, то поздравление должно быть с окончанием сессии.

3. Создать процедуру Proba_Z3, которая запускается при открытии рабочей книги и в окне InputBox сначала запрашивает имя студента, а затем его фамилию. В итоге программа должна в окне сообщения MsgBox по имени и фамилии поздравить студента с окончанием летней сессии, если его фамилия отличается от Иванов или Иванова. В противном случае необходимо просто поздороваться

4. Создать процедуру Proba_X1, которая запускается при открытии рабочей книги, просит ввести имя пользователя, сравнивает его с пятью именами, хранящимися в начальных строках столбца А Листа1. Если введенное имя совпадает, то в окне сообщения выводится текущая дата, время и поздравление с окончанием летней сессии, в противном случае выводится просто поздравление с началом сессии.

5. Создать процедуру PP_X2, которая запускается при открытии рабочей книги, просит ввести фамилию пользователя, сравнивает ее с пятью фамилиями, хранящимися в начальных строках столбца С Листа2. Если

введенная фамилия совпадает, то в окне сообщения выводится текущая дата, в противном случае выводится просто поздравление с удачным началом дня.

Работа с массивами

1. Написать функцию F_MAX, получающую в качестве параметра по ссылке массив целочисленных элементов и возвращающую значение максимального элемента из данного массива. Написать процедуру, в которой необходимо задать массив из десяти элементов: 3, 7, 21, 11, 4, 4, 67, 8, 91, 1 и вызвать функцию F_MAX для данного массива. Результат вывести в MsgBox

2. Написать функцию F_MIN, получающую в качестве параметра по ссылке массив целочисленных элементов и возвращающую значение минимального элемента из данного массива. Написать процедуру, в которой необходимо задать массив из десяти элементов: 33, 17, 121, 131, 4, 44, 67, 18, 91, 11 и вызвать функцию F_MIN для данного массива. Результат вывести в MsgBox

3. Написать функцию F_SORT, получающую в качестве параметра по ссылке массив целочисленных элементов и возвращающую значение максимального элемента из данного массива. Результат сортировки остается в том же массиве. Написать процедуру, в которой необходимо задать массив из десяти элементов: 3, 71, 1, 11, 54, 4, 67, 8, 91, 1 и вызвать функцию F_SORT для данного массива. Результат сортировки вывести в MsgBox

4. Написать функцию F_SORT1, получающую в качестве параметра по ссылке массив дробных элементов и возвращающую значение минимального элемента из данного массива. Результат сортировки остается в том же массиве. Написать процедуру, в которой необходимо задать массив из десяти элементов: 33.11, 1.134, 1, 0.1, 55, 4.7, 6.7, 8, 9.1, 1 и вызвать функцию F_SORT1 для данного массива. Результат сортировки вывести в MsgBox

5. Написать процедуру, которая получает в качестве первого параметра значение начального элемента массива `MMas`, в качестве второго параметра - количество элементов этого массива, с использованием элементов управления `TextBox`, `Label`, `Command Button`, размещенных на форме, запрашивает и вводит шаг изменения значений элементов массива, генерирует массив, вычисляет среднее арифметическое элементов массива и выводит в столбик элементы массива и среднее арифметическое с использованием `MsgBox`. Примечание: при написании процедуры необходимо сначала объявить массив, заполнить его значениями, затем выполнить необходимые вычисления

Написать процедуру, которая получает в качестве первого параметра количество элементов массива `Mas`, в качестве второго параметра – значение начального элемента массива, запрашивает с использованием `InputBox` шаг изменения значений элементов массива, вычисляет среднее арифметическое элементов массива, выводит в окно сообщения объекта `MsgBox` значения всех элементов массива, искомое среднее арифметическое и записывает его в ячейку `C1` рабочего листа «Лист1»