

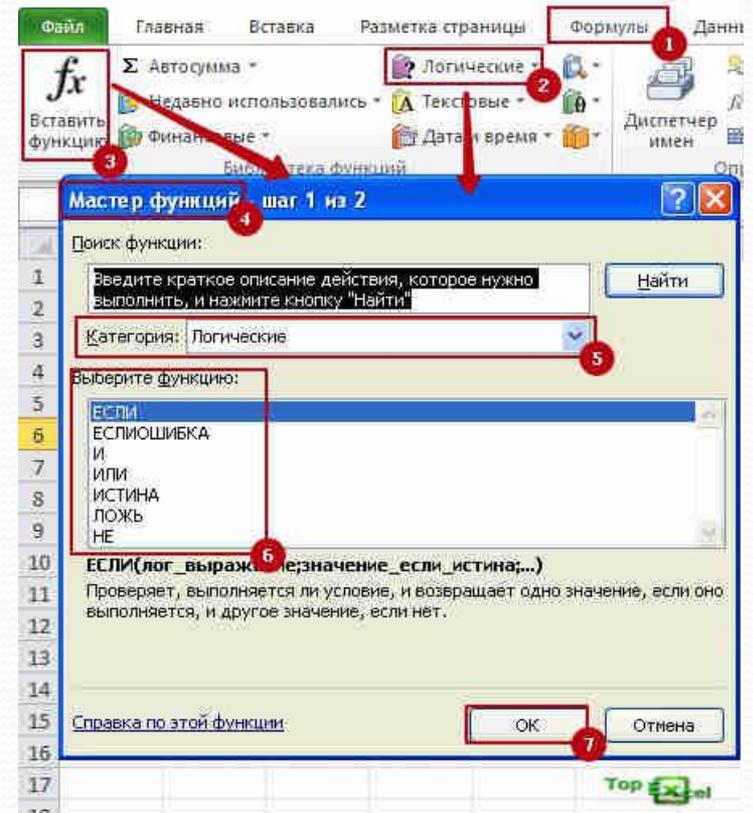
Логика в простых алгоритмах

Excel и VBA

Логические функции в Excel

Логические функции в Excel проверяют данные и возвращают результат «ИСТИНА», если условие выполняется, и «ЛОЖЬ», если нет.

- =И(выражение1; выражение2)
- =ИЛИ(выражение1; выражение2)
- =НЕ (выражение)
- =ЕСЛИ (условие; действие в случае «истина»; действие в случае «ложь»)



Функция ЕСЛИ

=ЕСЛИ(*лог_выражение*, *значение_если_истина*, *значение_если_ложь*)

- **Логическое выражение** – указывается практически любое выражение или значение, которое в процессе вычисления получает значение «ИСТИНА» или «ЛОЖЬ»;
- **Значение если истина** – это значение, которое будет возвращено, если аргумент «*Логическое выражение*» получит результат «ИСТИНА». В случае, когда этот аргумент не указан в формуле, возвращается результат «ИСТИНА»;
- **Значение если ложь** — это значение, которое будет возвращено, если аргумент «*Логическое выражение*» получит результат «ЛОЖЬ». В случае, когда этот аргумент опущен в формуле, возвращается результат «ЛОЖЬ».

Пример 1: простое условие

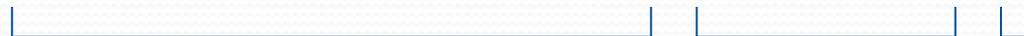
Проанализировать стоимость товарных остатков после уценки. Если цена продукта после переоценки ниже средних значений, то списать со склада этот продукт.

	А	В	С
	Название товара	Стоимость на момент поступления на склад	Срок хранения, мес.
1			
2	Товар 1	200р.	6
3	Товар 2	500р.	12
4	Товар 3	700р.	5
5	Товар 4	400р.	7
6	Товар 5	600р.	10
7	Товар 6	300р.	3

=ЕСЛИ(D2<СРЗНАЧ(D2:D7);"списан";"")

С	D	E
Срок хранения, мес.	Стоимость после переоценки	
6	133р.	списан
12	250р.	списан
5	467р.	
7	267р.	списан
10	300р.	
3	300р.	

=ЕСЛИ(D2<СРЗНАЧ(D2:D7);"списан";"")



Пример 2: вложенное условие

Расчет доплаты к ставке заработной платы персоналу, в зависимости от стажа работы. Всего существует три вида доплат:

- стаж работы 3 года – 10%,
- стаж работы 10 лет – 20%
- стаж работы 20 лет – 30%.

Для получения результата и автоматического расчёта доплаты в зависимости от стажа нам нужна единая формула.

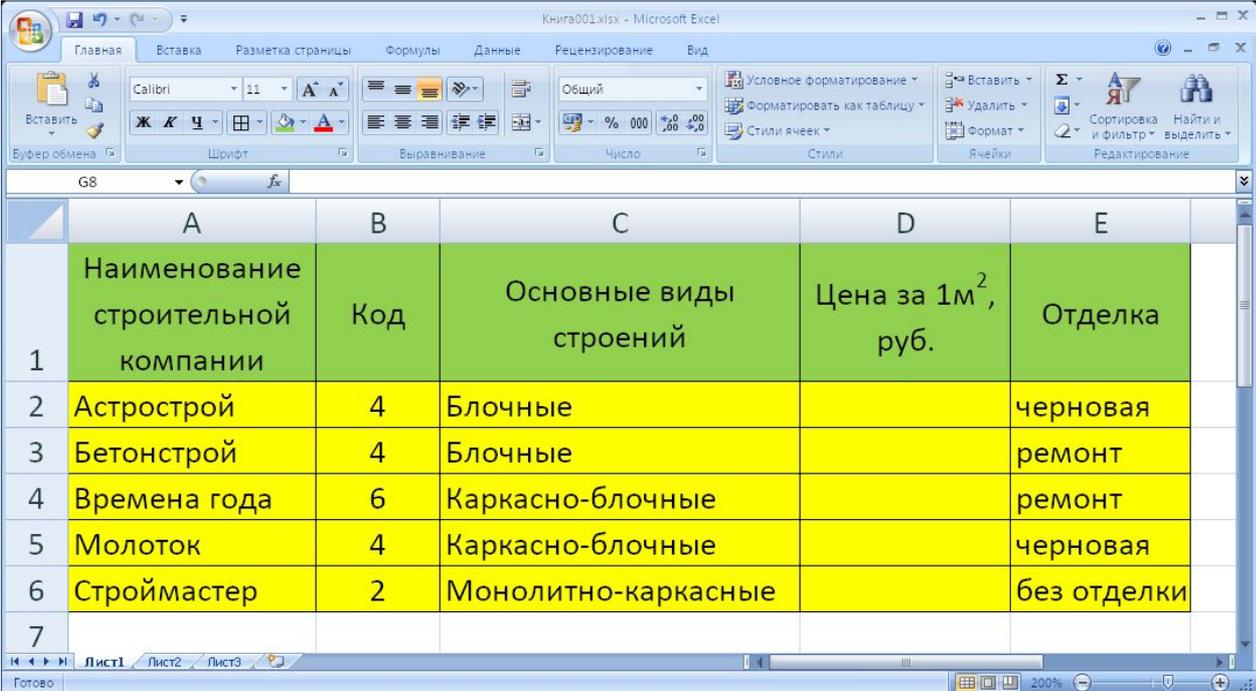
E2		fx =ЕСЛИ(C2>=20;"30%";ЕСЛИ(C2>=10;"20%";ЕСЛИ(C2>=3;"10%";"0%")))						
	A	B	C	D	E	F	G	H
1	№ п/п	Сотрудник	Стаж	Ставка	% за выслугу	Доплата за выслугу	К выдаче	
2	1	Нагаев Артём	12	30 000	20%	6 000	36 000	
3	2	Иванов Сергей	3	30 000	10%	3 000	33 000	
4	3	Петров Иван	21	30 000	30%	9 000	39 000	
5	4	Сидоров Пётр	18	30 000	20%	6 000	36 000	
6	5	Абазов Василий	6	30 000	10%	3 000	33 000	
7	6	Козуб Марина	1	30 000	0%	0	30 000	
8								

=ЕСЛИ(C2>=20; «30%»;ЕСЛИ(C2>=10; «20%»;ЕСЛИ(C2>=3; «10%»;»0%«))).



Пример 3: сложное логическое выражение

В представленной электронной таблице столбцу «Цена за 1м², руб» должны соответствовать следующие значения цены: Астрострой и Молоток – 47000; Бетонстрой и Времена года – 60000; Строймастер – 40000. Формула вводится в ячейку D2, а в остальные ячейки копируется методом автозаполнения.



	A	B	C	D	E
	Наименование строительной компании	Код	Основные виды строений	Цена за 1м ² , руб.	Отделка
1					
2	Астрострой	4	Блочные		черновая
3	Бетонстрой	4	Блочные		ремонт
4	Времена года	6	Каркасно-блочные		ремонт
5	Молоток	4	Каркасно-блочные		черновая
6	Строймастер	2	Монолитно-каркасные		без отделки
7					

- =ЕСЛИ(ИЛИ(А2="Астрострой";А2="Молоток");47000; ЕСЛИ(И(А2="Бетонстрой";А2="Времена года");60000;40000))
- =ЕСЛИ(И(А2="Астрострой";А2="Молоток");47000; ЕСЛИ(И(А2="Бетонстрой";А2="Времена года");60000;40000))
- =ЕСЛИ(ИЛИ(А2=Астрострой;А2=Молоток);47000; ЕСЛИ(ИЛИ(А2=Бетонстрой;А2=Времена года);60000;40000))
- =ЕСЛИ(ИЛИ(А2="Астрострой";А5="Молоток");47000; ЕСЛИ(ИЛИ(А3="Бетонстрой";А4="Времена года");60000;40000))
- =ЕСЛИ(ИЛИ(А2="Бетонстрой";А2="Времена года");60000; ЕСЛИ(ИЛИ(А2="Астрострой";А2="Молоток");47000; 40000))

Visual Basic for Application (VBA)

VBA

- Visual Basic для приложений - это инструмент разработки приложений
- VBA –приложение, которое встраивается в другое приложение и расширяет его функциональные возможности

Пример: можно добавить собственное меню или функцию к встроенным средствам Excel

VBA не является языком объектно-ориентированного программирования !!!!

Но в нем широко используются элементы объектно-ориентированного подхода и связанные с ним понятия. Назначением VBA является совместная работа с прикладными программами, а с помощью Visual Basic можно разрабатывать самостоятельное приложение.

Основное различие – в Visual Basic можно создавать полностью самостоятельное приложение, а проекты VBA выполняются только с помощью приложения, которое поддерживает VBA.

Возможности VBA

Язык VBA- это подмножество VB, которое включает почти все его средства создания приложений, структуры данных и управляющие структуры, возможность создания пользовательских типов данных. Это позволяет эффективно его использовать для автоматизации деятельности, связанной с обработкой различных типов документов, например:

- Создать собственное диалоговое окно и придать ему требуемый внешний вид
- Создать макросы, расширяющие функциональные возможности приложения, в которое встроен VBA
- Изменить меню приложения Office
- Управлять другими приложениями Office или принадлежащими ему данными
- Объединить данные из нескольких приложений Office в одном документе Office
- Автоматически создавать или изменять страницы Web, совместно используя приложения Office и VBA

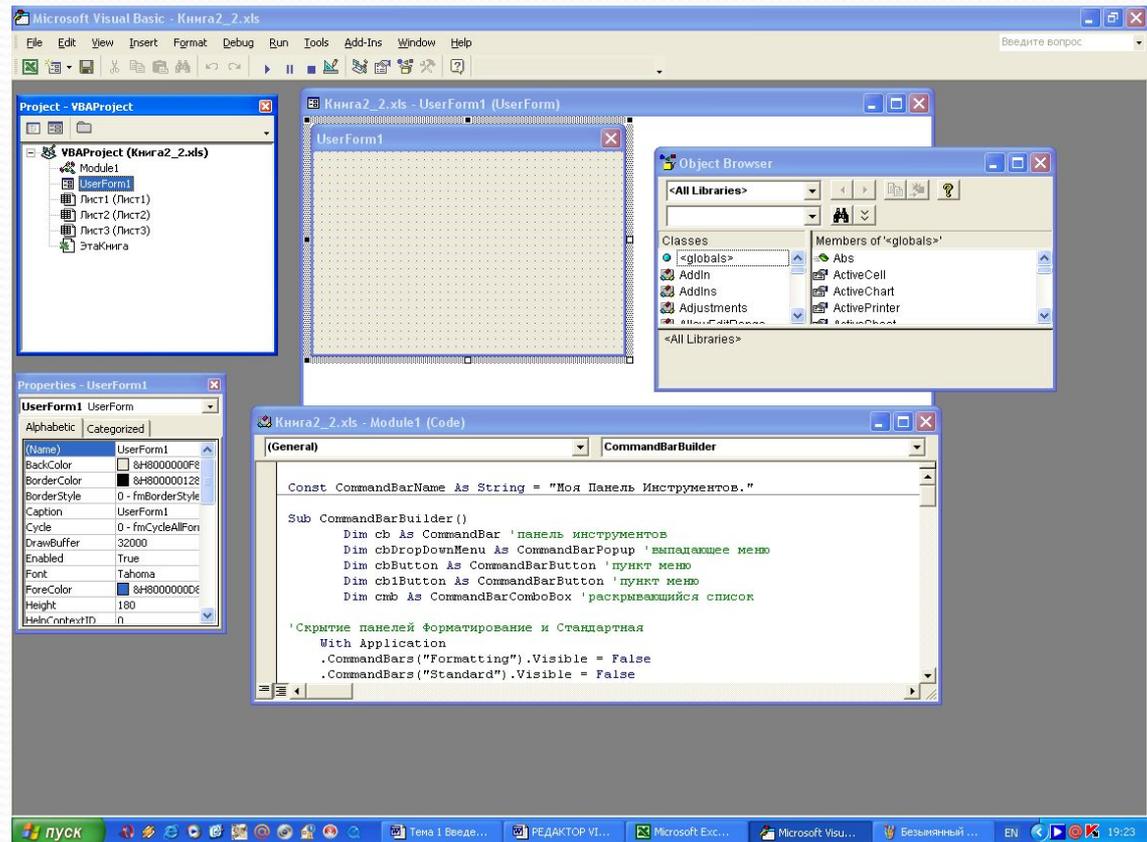
Элементы проекта VBA

При решении задач с помощью VBA требуется создать проект. Проект включает несколько элементов:

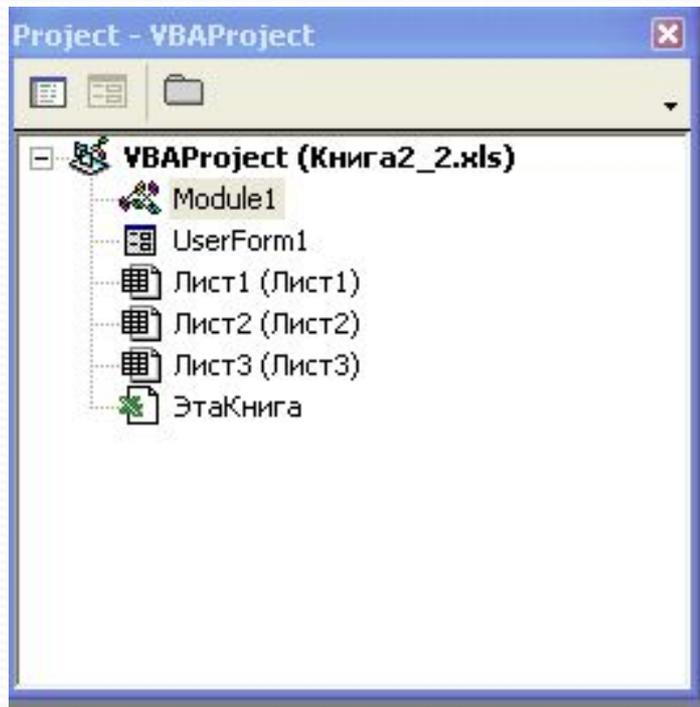
- *Основное приложение* - приложение, в котором разрабатывается и выполняется проект VBA (WORD, EXCEL, ACCESS, POWER POINT)
- *Среда разработки VBA* - интегрированная среда разработки или IDE
- *Формы*, которые используются для ввода или вывода данных, а так же процедура обработки событий, возникающих в этих формах
- *Код (программа)*
- *Модули кода (подпрограммы)*
- *Модули класса* - содержат полную информацию о каждом типе объекта

РЕДАКТОР VISUAL BASIC APPLICATION

- окно проекта (Project)
- окно свойств (Properties Window)
- окно формы (UserForm),
- окно программного кода (Code),
- окно просмотра (Object Browser)



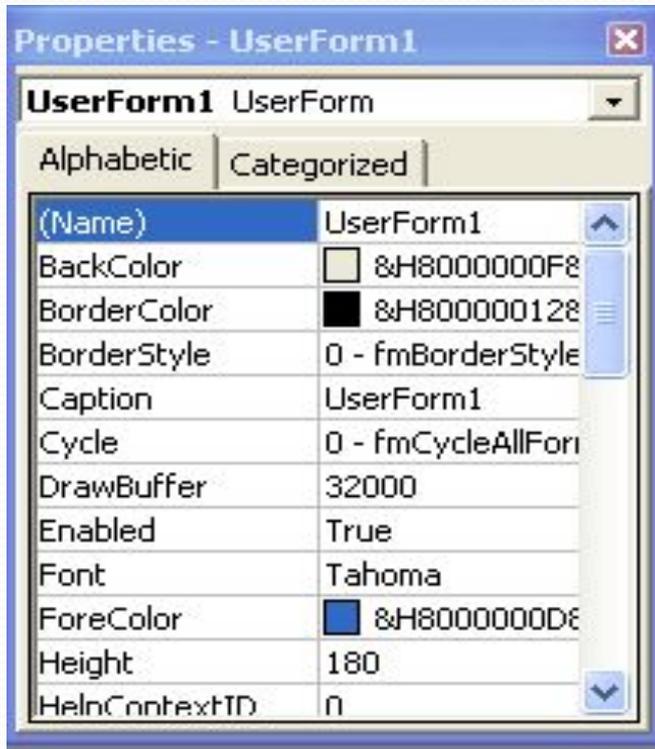
Окно проекта (Project)



содержит древо-диаграмму открытых в данный момент файлов (документы, шаблоны или рабочие книги) и объектов, содержащихся в этих файлах (модули, ссылки, формы).

<Ctrl>+<R>

Окно СВОЙСТВ (Properties Window)

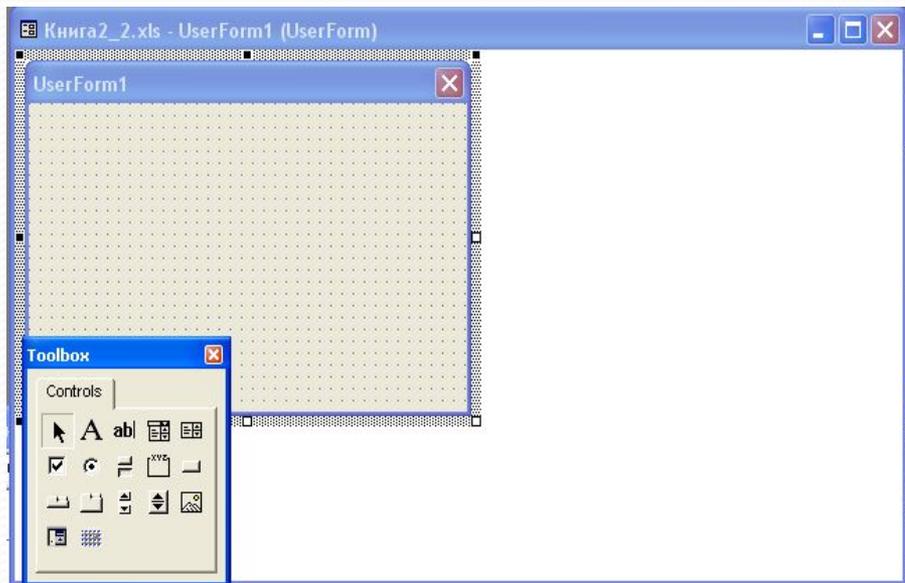


Содержит все свойства объекта текущего выбора.

Вкладка ***Alphabetic*** (по алфавиту) этого окна предоставляет список свойств выделенного объекта по алфавиту.

Вкладка ***Categorized*** (по категориям) – список свойств по категориям.

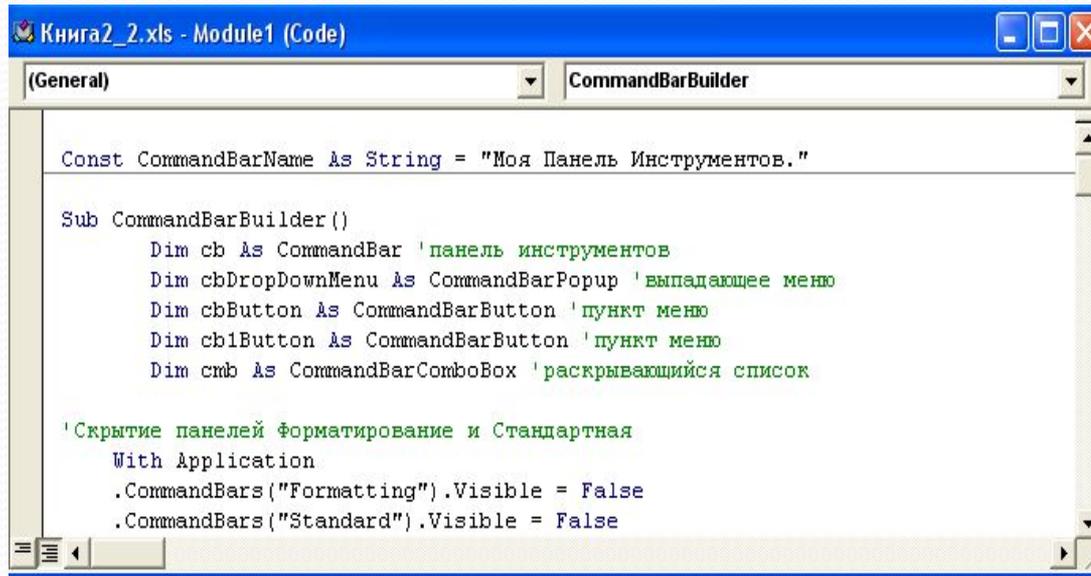
Окно формы (UserForm)



Содержит элементы управления, при помощи которых разработчик может создавать диалоговые окна и формы произвольной конфигурации, максимально приспособленные для решения конкретных задач приложения.

Элементы управления группируются в панели инструментов **Toolbox**

Окно программного кода (Code)



```
Книга2_2.xls - Module1 (Code)
(General) CommandBarBuilder

Const CommandBarName As String = "Моя Панель Инструментов."

Sub CommandBarBuilder()
    Dim cb As CommandBar 'панель инструментов
    Dim cbDropDownMenu As CommandBarPopup 'выпадающее меню
    Dim cbButton As CommandBarButton 'пункт меню
    Dim cb1Button As CommandBarButton 'пункт меню
    Dim cmb As CommandBarComboBox 'раскрывающийся список

    'Скрытие панелей Форматирование и Стандартная
    With Application
        .CommandBars("Formatting").Visible = False
        .CommandBars("Standard").Visible = False
    End With
End Sub
```

окно предназначенное для просмотра, редактирования или создания программного кода VBA.

Типы данных

- **Byte** – (1 байт) для хранения положительных чисел от 0 до 255
- **Boolean** – (2 байта) для хранения логических значений. Может содержать значения False, True
- **Currency** – (8 байт) от -922337203685477.5808 до 922337203685477.5807
- **Date** – (8 байт) для хранения информации о дате и времени
- **Decimal** – (12 байт) переменные сохраняются как 96-битные знаковые целые
- **Double** – (8 байт) отрицательные и положительные числа
- **Integer** – (2 байта) все целые числа от -32768 до 32767
- **Long** – (4 байта) все целые числа от -2417483648 до 2417483647
- **Object** – (4 байта) используется для доступа к любому объекту, распознаваемому VBA. Сохраняет адрес объекта в памяти
- **Single** – (4 байта) используется для хранения текста. Может содержать от 0 до 2 миллиардов символов. Может содержать от одного до 6540000
- **Variant** – (16 байт) может хранить любой другой тип данных. Диапазон зависит от фактически сохраняемых данных

Переменные

Переменная – имя, которое программист дает области компьютерной памяти, используемой для хранения данных какого-либо типа.

Правила:

- Имена переменных не могут содержать пробелы, точку или любой другой символ, который VBA использует для обозначения математических операций и операций сравнения
- Имя должно начинаться с буквы алфавита. После первой буквы имя переменной может состоять из любых комбинаций, цифр, букв, символов
- Имя переменной не может дублировать определенные ключевые слова VBA. Если вы выберете имена переменных
- Имя переменной должно быть уникальным в рамках его области действия, т.е. имя переменной должно быть уникальным в пределах процедуры или модуля, в котором вы объявляете эту переменную

Пример правильных имен переменных:

- *Var1*
- *Pay_date*
- *Add_Item*
- *Part*
- *Line12*

Пример неправильных имен переменных:

- *New Item* – содержит пробел
- *5thDimension* – начинается с цифры
- *Dim* – дублирует ключевое слово
- *Week\Day* – содержит неверный символ \
- *_Pay* – содержит недопустимый символ вначале

Объявление переменных

- **Неявное**

Использование в операторе VBA. Все переменные, которые были объявлены неявно, имеют тип VARIANT

- **Явное**

ускоряет выполнение кода, уменьшает количество ошибок в результате неправильного написания имени, код становится более читабельным и ПОНЯТНЫМ

Dim ИмяПеременной1 [**As Тип**], ИмяПеременной2 [**As Тип**]

Примеры

- *DIM Товар As Single*
- *DIM NetValue As Integer*
- *Dim NetValue As String*30* – строковая переменная фиксированной длины (30 символов)

ориентированное

программирование

Класс - это шаблон для создания объектов. Класс определяет свойства и методы объекта.

Объект - это нечто целостное, неделимое, содержащее в себе данные и код. В объекте может храниться информация и программные инструкции для обработки этой информации.

У объектов есть свойства и методы. Объекты могут откликаться на события.

Объект.Метод или Объект.Метод (Параметры)

Объект.Свойство=Значение

Переменная=Объект.Свойство

Пример: `ActiveSheet . Name = "Январь"` - изменить имя текущего листа

`Worksheets("Обзор") . Delete` – удалить лист

Структура программного кода

[Option Explicit]

**[Private | Public] [Static] Sub <Имя> ([<Список
аргументов>])**

[Dim <Имя> [As <Тип>]]

[Const <Имя> [As <Тип>] = <Выражение>]

[Инструкции]

[Exit Sub]

[Инструкции]

End Sub

- Option Explicit** – инструкция, предписывающая явное описание всех переменных, встречающихся в программе;
- Public** – ключевое слово, которое указывает, что процедура доступна для всех других процедур во всех модулях;
- Private** – ключевое слово, указывающее, что процедура доступна для других процедур только того модуля, в котором она описана;
- Static** – ключевое слово, указывающее, что локальные переменные процедуры сохраняются в промежутках времени между вызовами этой процедуры;

<Имя> – имя процедуры, удовлетворяющее стандартным правилам именованя;

<Список аргументов> – список переменных, представляющий аргументы, которые передаются в процедуру при ее вызове. Имена переменных разделяются запятой.

Dim <Имя> [As <Тип>] – блок описания переменных;

Const <Имя> [As <Тип>] = <Выражение> – блок описания констант;

Инструкции – любой набор любых команд VBA;

Exit Sub – инструкция, выполнение которой приводит к немедленному выходу из процедуры;

Sub, End Sub – служебные слова VBA.

Окно ввода (функция InputBox)

Синтаксис:

InputBox (prompt, [, title] [, default] [, xpos] [, ypos] [, helpfile, context])

Аргументы:

prompt – строковое выражение, отображаемое как сообщение в диалоговом окне.

title – строковое выражение, отображаемое в строке заголовка диалогового окна. Если этот аргумент опущен, в строку помещается имя приложения.

default – строковое выражение, отображаемое в поле ввода как используемое по умолчанию, если пользователь не введет другую строку. Если этот аргумент опущен, поле ввода изображается пустым.

xpos – числовое выражение, задающее расстояние по горизонтали между левой границей диалогового окна и левым краем экрана. Если этот аргумент опущен, диалоговое окно выравнивается по центру экрана по горизонтали.

ypos – числовое выражение, задающее расстояние по вертикали между верхней границей диалогового окна и верхним краем экрана. Если этот аргумент опущен, диалоговое окно помещается по вертикали примерно на одну треть высоты экрана.

helpfile – строковое выражение, определяющее имя файла справки, содержащего справочные сведения о данном диалоговом окне. Если этот аргумент указан, необходимо также наличие аргумента *context*.

context – числовое выражение, определяющее номер соответствующего раздела справочной системы. Если этот аргумент указан, необходимо также наличие аргумента *helpfile*.

Окно сообщений (процедура MsgBox)

Процедура **MsgBox** выводит на экран диалоговое окно, содержащее сообщение, устанавливает режим ожидания нажатия кнопки пользователем. Затем возвращает значение типа **Integer**, указывающее, какая кнопка была нажата.

Значение	Нажатая кнопка
1	ОК
2	Отмена
3	Стоп
4	Повтор
5	Пропустить
6	Да
7	Нет

Процедура MsgBox

Синтаксис:

MsgBox (prompt, [, button][, title] [, helpfile, context])

Аргументы:

prompt – строковое выражение, отображаемое как сообщение в диалоговом окне.

button – числовое выражение, представляющее сумму значений, которые указывают число и тип отображаемых кнопок, тип используемого значка, основную кнопку. Значение этого аргумента по умолчанию равняется 0.

Процедура MsgBox

title – строковое выражение, отображаемое в строке заголовка диалогового окна. Если этот аргумент опущен, в строку помещается имя приложения.

helpfile – строковое выражение, определяющее имя файла справки, содержащего справочные сведения о данном диалоговом окне. Если этот аргумент указан, необходимо также наличие аргумента *context*.

context – числовое выражение, определяющее номер соответствующего раздела справочной системы. Если этот аргумент указан, необходимо также наличие аргумента *helpfile*.

Процедура MsgBox

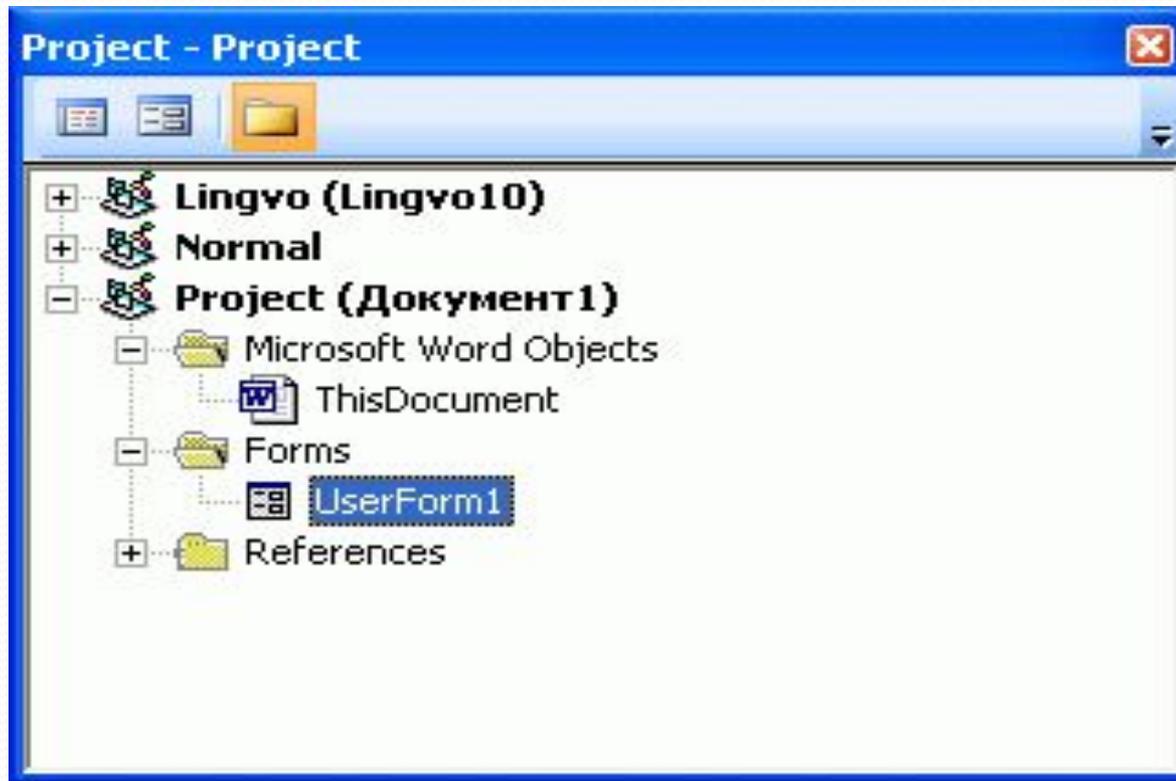
Значения аргумента,
определяющие отображаемые

Константа	Значение	Отображаемые кнопки
VbOKOnly	0	ОК
VbOKCancel	1	ОК, Отмена
VbAbortRetryIgnore	2	Стоп, Повтор, Пропустить
VbYesNoCancel	4	Да, Нет, Отмена
VbYesNo	5	Да, Нет
VbRetryCancel		Повтор, Отмена

Значения аргумента,
определяющие отображаемые

Константа	Значение	Значок сообщения
VbCritical	16	
VbQuestion	32	
VbExclamation	48	
VbInformation	64	

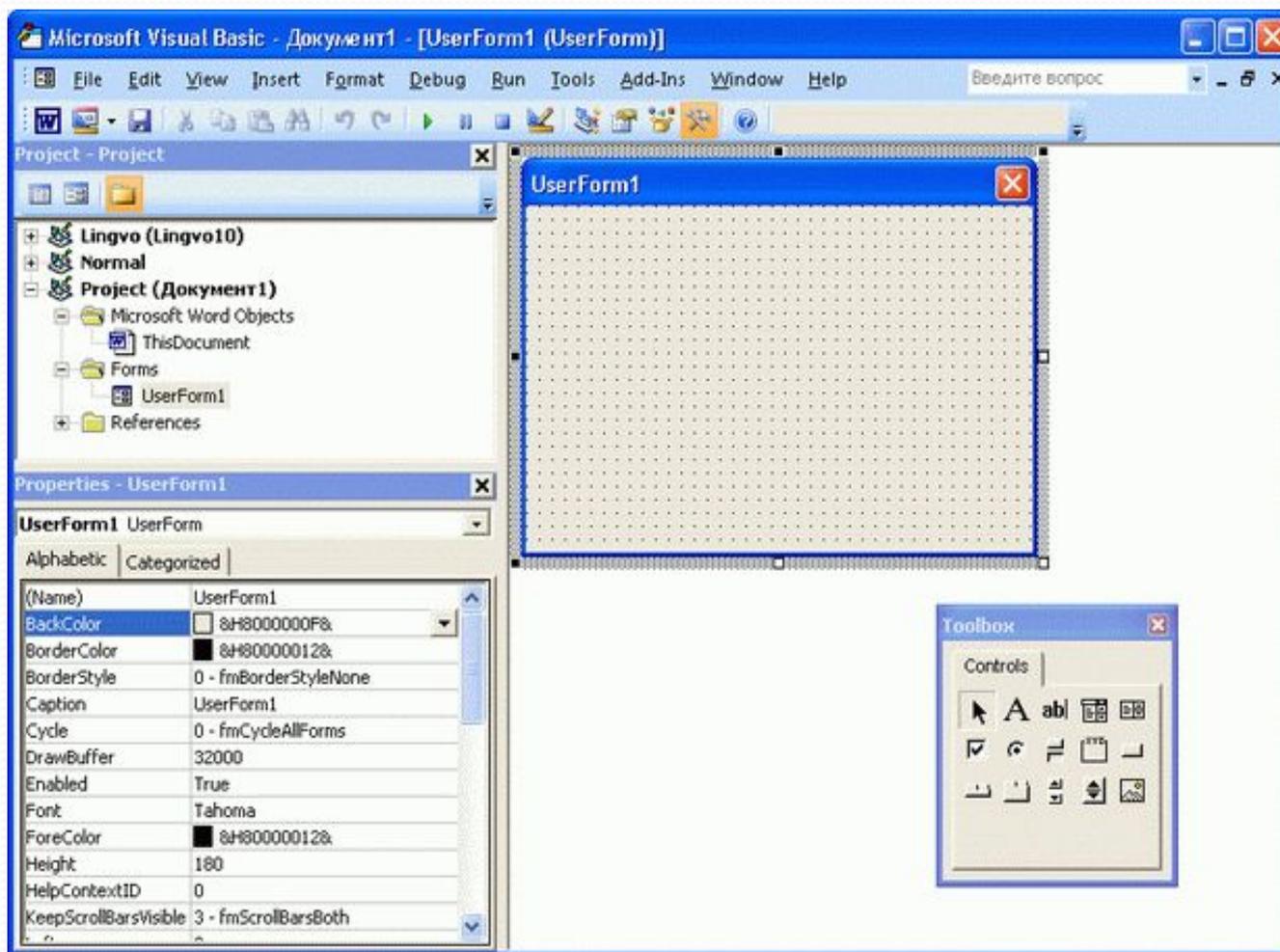
Окно Project



служит для навигации по проекту

Форма

Insert • User Form (Вставка • Форма)



Префиксы имен элементов

управления

Элемент управления	Префикс
UserForm (Форма)	<i>frm</i>
CommandButton (Кнопка)	<i>cmd</i>
TextBox (Текстовое поле)	<i>txt</i>
Label (Надпись)	<i>lbl</i>
ListBox (Список)	<i>lst</i>
ComboBox (Поле со списком)	<i>cbo</i>
ScrollBar (Полоса прокрутки)	<i>scr</i>
SpinButton (Счетчик)	<i>spn</i>
OptionButton (Радиокнопка)	<i>opt</i>
CheckBox (Флажок)	<i>chk</i>
ToggleButton (Выключатель)	<i>tgl</i>
Frame (Рамка)	<i>fra</i>
Image (Картинка)	<i>img</i>
MultiPage (Набор страниц)	<i>mlt</i>
TabStrip (Набор вкладок)	<i>tab</i>

Свойства элементов управления

Caption (Надпись) — определяет надпись, которая видна на элементе управления. Изменим свойство *Caption* для объекта `frm_Main` на "Моя первая программа", а для объекта `cmd_First` — "Нажми меня!".

Width (Ширина), Height (Высота) — определяют ширину и высоту объекта, которые измеряются в пикселях.

Font (Шрифт) — при попытке редактирования этого свойства открывается окно настройки параметров шрифта, в котором можно выбрать гарнитуру, размер, начертание шрифта.

ForeColor (Цвет шрифта) — задает цвет шрифта элемента управления.

Свойства элементов управления

BackColor (Цвет фона) — позволяет задать цвет фона элемента управления.

У кнопок и других элементов управления есть специфические свойства. В их число входят:

Top (Расстояние от верха формы), *Left* (Расстояние от левого края формы) — задают положение элемента на форме.

Visible (Видимость) — может принимать значения *True* (Истина) — то есть объект будет видимым, и *False* (Ложь) — объект будет невидимым.

Enable (Включено) — если это свойство установлено в *True* (Истина) — объектом можно пользоваться. Если в *False* (Ложь) — объект будет неактивен.