Лабораторный практикум по курсу  
«Информационные системы и технологии»  
Часть 2. MS Excel 2007

**Лабораторная работа №3**

**«Макросы и элементы управления»**

**Подготовила А.А. Коробецкая**

Лабораторная работа содержит три задания, демонстрирующие возможности использования макросов и элементов управления в MS Excel.

Первое задание выполняется на основе варианта таблицы, полученной в лабораторной работе №1. Другие два задания выполняются без вариантов.

Каждое задание оформляется на отдельном листе. В шапке листа указывается номер и название задания, ФИО и группа выполнившего, номер варианта.

Отчет по работе выполнять не требуется, к сдаче предоставляется только файл Excel.

Содержание

[Макросы и элементы управления 2](#_Toc370993349)

[Задание 1. Запись макросов 4](#_Toc370993350)

[Задание 2. Использование элементов управления 10](#_Toc370993351)

[Задание 3. Редактирование макросов через VBA 15](#_Toc370993352)

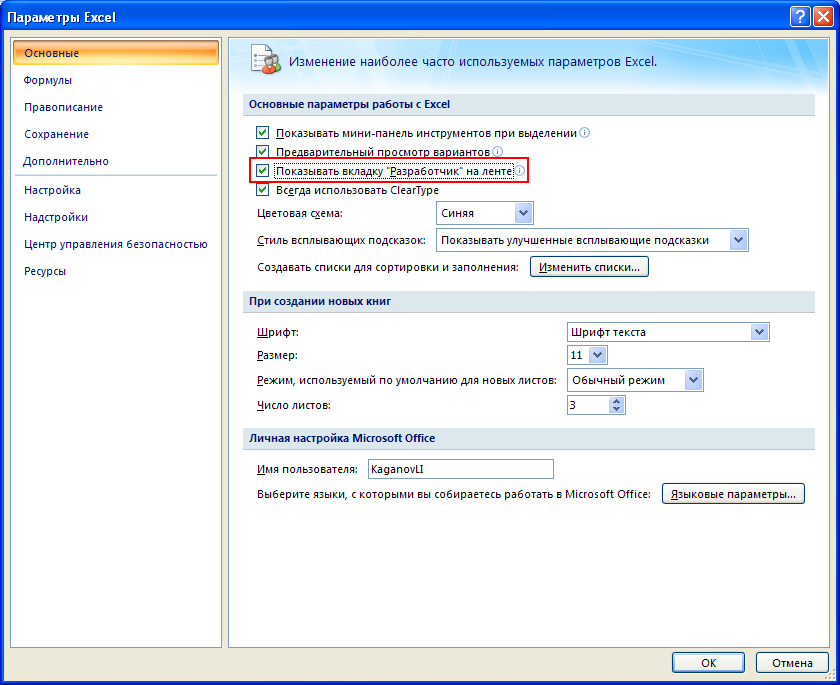
Макросы и элементы управления

**Макросы** – это подпрограммы на языке Visual Basic for Applications (VBA), но простые макросы можно создавать и не зная языков программирования. Они позволяют многократно повторять однотипные действия: удаление, копирование, вставку строк, заполнение типовыми формулами, генерацию случайных чисел и др.

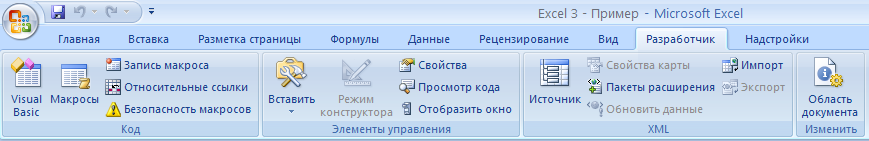
Простейший способ создания макросов – это *запись* действий, выполняемых пользователем. В дальнейшем эти действия можно будет повторить сколько угодно раз, просто нажав на кнопку.

Элементы управления – это кнопки, выпадающие списки, переключатели, флажки и т.д. Они позволяют изменять значения ячеек и запускать макросы.

Чтобы включить инструменты для работы с макросами, откройте «Параметры Excel» (кнопка в меню Office). На вкладке «Основные» установите флажок «Показывать вкладку "Разработчик" на ленте».



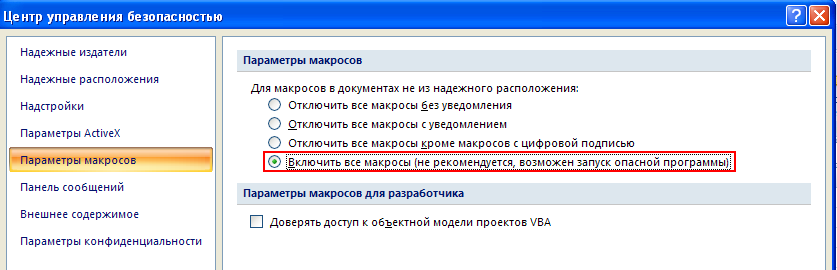
После нажатия ОК на ленте появится еще одна вкладка «Разработчик»:



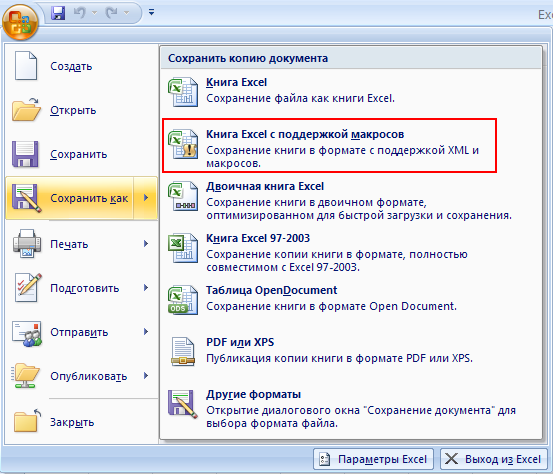
На ней находятся и инструменты для работы с макросами, и элементы управления.

Кроме того, кнопки для записи макросов можно найти на вкладке «Вид».

Из соображений безопасности макросы по умолчанию блокируются. На вкладке «Разработчик» нажмите кнопку «Безопасность макросов». Выберите «Включить все макросы». Учтите, что при этом запускаться будут не только ваши, но и все остальные макросы, в которые может быть встроен вредоносный код. После завершения лабораторного практикума рекомендуется вернуть настройки безопасности.



Кроме того, чтобы макросы сохранились, файл необходимо сохранять как «Книга Excel с поддержкой макросов».



# Запись макросов

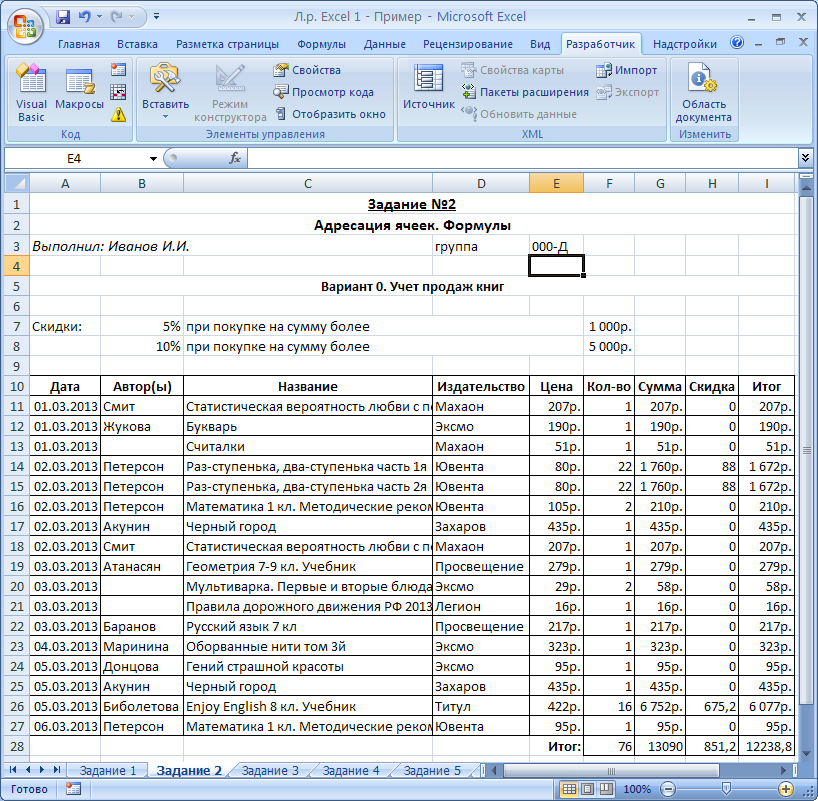
## Задание

Автоматизировать работу с электронной таблицей, созданной в лабораторной работе №1. Создать макросы для:

* добавления пустой строки в начало таблицы;
* вставки копии выделенной строки;
* удаления выделенной строка;
* очистки всей таблицы;
* сортировки по трем разным признакам.

Добавить элементы управления для запуска макросов.

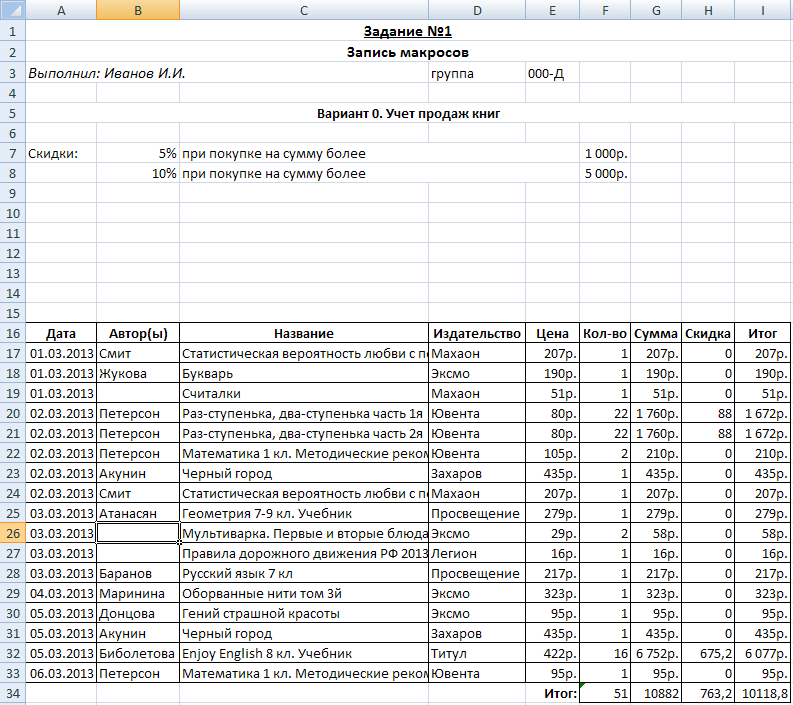
## Пример варианта задания



## Указания к выполнению

1. Скопируйте лист «Задание 2» из л.р.1 в новую книгу. Сохраните ее как книгу Excel с поддержкой макросов (см. выше). Переименуйте лист в «Задание 1».

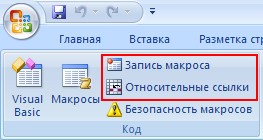
В ходе выполнения работы нам потребуется некоторое пространство для размещения кнопок. Добавьте 5-6 пустых строк перед шапкой таблицы.



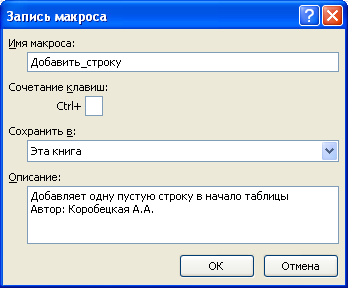
1. Первым запишем макрос для добавления строки в начало таблицы.

На вкладке разработчик проверьте, чтобы кнопка «Относительные ссылки» **не** была нажата.

Нажмите кнопку «**Запись макроса**».



В появившемся окне необходимо указать свойства макроса: название, сочетание клавиш, описание и в какую книгу его необходимо сохранить. Укажите понятное название и описание макроса, горячую клавишу назначать не нужно. Имя макроса может содержать цифры и буквы, можно русские, знак подчеркивания. Пробелов и знаков препинания быть не должно.

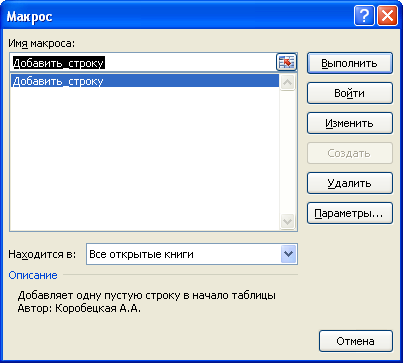


После нажатия кнопки ОК начнется запись макроса, и все выполненные на листах действия будут повторяться при его запуске.

Выделите первую строку таблицы (в примере строка 17), и вставьте новую строку («Главная» – «Вставка» – «Строки» или через правый клик по номеру строки).

На этом первый макрос закончен. Нажмите «**Остановить запись**».

Просмотреть и запустить получившийся макрос можно через окно макросов (кнопка «**Макросы**» на вкладке «Разработчик» или Alt+F8):



При нажатии кнопки «**Выполнить**» в таблицу должна добавиться еще одна строка.

Здесь же можно удалить неправильно записанный макрос.

1. Теперь создадим макросы для удаления и копирования строки.

Их отличие состоит в том, что удалять или копировать нужно выделенную в данный момент строку, т.е. заранее неизвестно, какую именно.

Для этого существует кнопка «**Относительные ссылки**». Если она нажата, то все действия будут записываться относительно той ячейки (или диапазона), которая была выделена в момент начала записи.

Поэтому сначала выделите последнюю строку таблицы (можно любую). Включите кнопку «Относительные ссылки» и начните запись макроса. Самостоятельно задайте его имя и описание.

Не изменяя выделения, на вкладке «Главная» выберите «Удаление» – «Удалить строки с листа».

Остановите запись макроса.

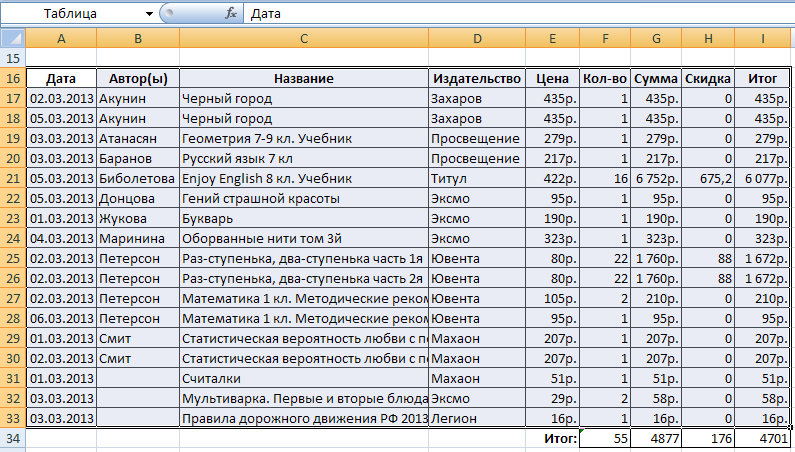
Проверьте его работу – макрос будет удалять все строки, которые в данный момент выделены.

Самостоятельно запишите макрос для копирования выделенной строки. Скопированную строку нужно вставить в таблицу над выделенной.

1. Добавим макросы для сортировки таблицы. Как выполнять сортировку, вам уже известно (л.р.1 задание 3).

Однако в данном случае нам не подходят ни абсолютные, ни относительные ссылки. При использовании абсолютных ссылок не будет учитываться, что строки в таблице могут добавляться или удаляться. При использовании относительных ссылок от пользователя требуется сначала выделить всю таблицу, а он может это сделать неправильно.

Лучше использовать не адреса, а *имена* ячеек. При добавлении или удалении ячеек внутри именованной области она автоматически растягивается или сжимается. Задайте имя области таблицы вместе с заголовками столбцов.

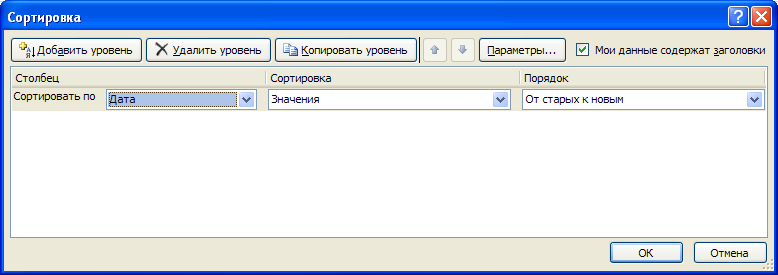


Замечание. Если в предыдущем пункте вы вставляли скопированную строку ниже исходной, то при копировании последней строки, она не добавится к области таблицы.

Теперь выполним запись макроса для сортировки по дате. После начала записи макроса выделите созданную область таблицы.



После этого выполним сортировку:



Остановим запись макроса. Проверьте – теперь даже после добавления и удаления строк, сортироваться будет вся таблица.

Самостоятельно добавьте еще два макроса для сортировки по одному или нескольким признакам.

1. Последним запишем макрос, очищающий всю таблицу.

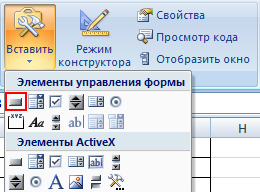
Последовательность действий при записи макроса:

1. Скопировать заголовки столбцов таблицы в свободные ячейки.
2. Выделить таблицу по имени области.
3. Очистить таблицу.
4. Вырезать и вставить заголовки столбцов обратно.

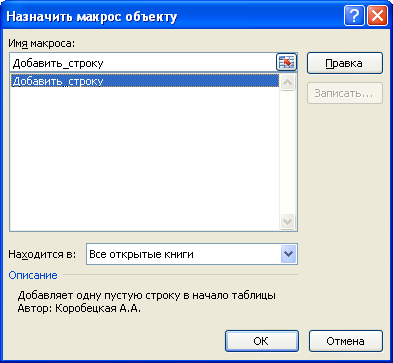
Проверьте – таблица должна очищаться полностью даже после добавления и удаления строк.

1. Каждый раз запускать макросы через вкладку «Разработчик» неудобно. Добавим прямо на лист элементы управления для их запуска.

Макросы для редактирования таблицы будем запускать через кнопки. На вкладке «Разработчик» выберите «Вставить» – «Кнопка» (из группы «Элементы управления формы»).



Разместите кнопку на листе. Сразу же после этого появится окно для выбора связанного макроса. Укажите макрос на добавление строки и нажмите ОК. Измените надпись на кнопке на «Добавить».



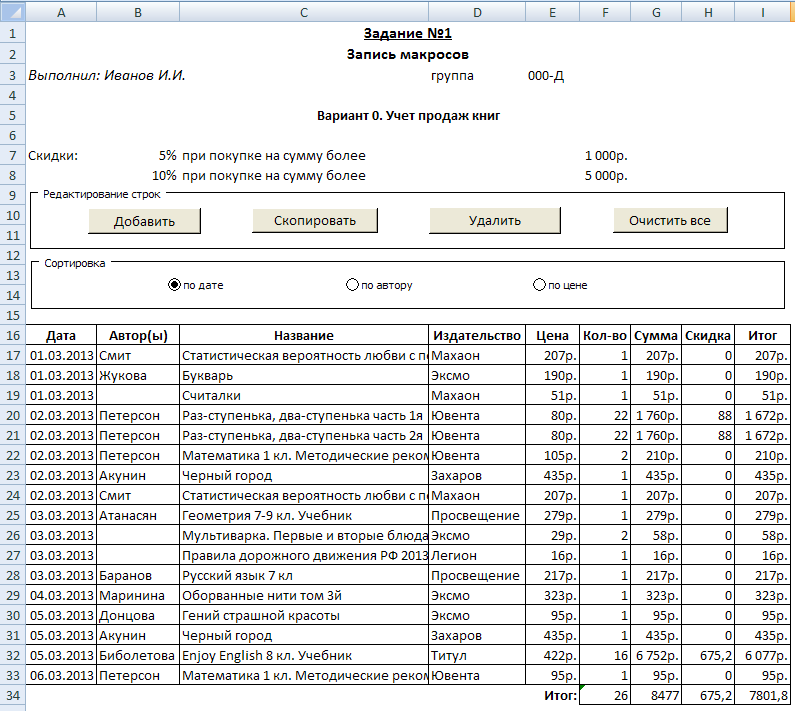
Теперь при левом клике по кнопке будет запускаться макрос. Чтобы выделить кнопку для редактирования используйте правую кнопку мыши или на ленте вкладка «Разметка страницы» – «Область выделения». Отобразится список всех элементов управления на листе.

По аналогии добавьте кнопки для остальных макросов. Разместите их над таблицей и оформите рамочкой («Вставить» – «Группа»).

Для выбора варианта сортировки используйте переключатель .

Чтобы привязать макрос к переключателю, щелкните по нему правой кнопкой и выберите «Назначить макрос…».

Итоговый вид таблицы:



Задание для продвинутых

После выполнения третьего задания, отредактируйте макросы через VBA: макросы на копирование и удаление должны действовать только внутри таблицы. Для этого необходимо проверить номер выделенной строки относительно границ области «Таблица».

# Использование элементов управления

## Задание

Создать графики с возможностью выбора рядов для отображения. На одном графике отображается только отмеченные «галочками» ряды, на другом – только один ряд (выбор через выпадающий список). Использовать макросы не нужно.

## Исходные данные

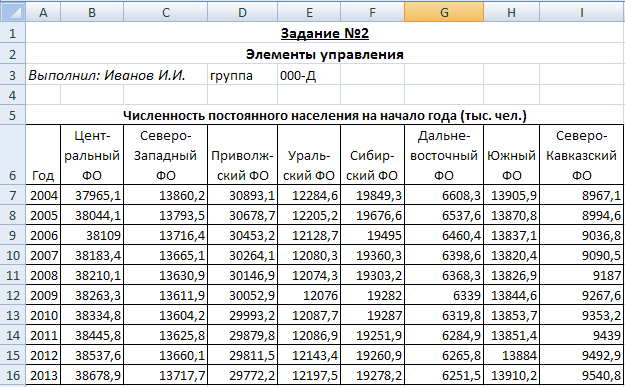
Дана таблица с численностью постоянного населения по федеральным округам (ФО) на начало года (тыс. чел.) для федеральных округов:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Год | Цент­ральный ФО | Северо-Западный ФО | Приволж­ский ФО | Ураль­ский ФО | Сибир­ский ФО | Дальне­восточный ФО | Южный ФО | Северо-Кавказский ФО |
| 2004 | 37965,1 | 13860,2 | 30893,1 | 12284,6 | 19849,3 | 6608,3 | 13905,9 | 8967,1 |
| 2005 | 38044,1 | 13793,5 | 30678,7 | 12205,2 | 19676,6 | 6537,6 | 13870,8 | 8994,6 |
| 2006 | 38109,0 | 13716,4 | 30453,2 | 12128,7 | 19495,0 | 6460,4 | 13837,1 | 9036,8 |
| 2007 | 38183,4 | 13665,1 | 30264,1 | 12080,3 | 19360,3 | 6398,6 | 13820,4 | 9090,5 |
| 2008 | 38210,1 | 13630,9 | 30146,9 | 12074,3 | 19303,2 | 6368,3 | 13826,9 | 9187,0 |
| 2009 | 38263,3 | 13611,9 | 30052,9 | 12076,0 | 19282,0 | 6339,0 | 13844,6 | 9267,6 |
| 2010 | 38334,8 | 13604,2 | 29993,2 | 12087,7 | 19287,0 | 6319,8 | 13853,7 | 9353,2 |
| 2011 | 38445,8 | 13625,8 | 29879,8 | 12086,9 | 19251,9 | 6284,9 | 13851,4 | 9439,0 |
| 2012 | 38537,6 | 13660,1 | 29811,5 | 12143,4 | 19260,9 | 6265,8 | 13884,0 | 9492,9 |
| 2013 | 38678,9 | 13717,7 | 29772,2 | 12197,5 | 19278,2 | 6251,5 | 13910,2 | 9540,8 |

## Указания к выполнению

Элементы управления можно использовать и без макросов, а связывать их непосредственно с ячейками листа.

1. Создайте новый лист, назовите его «Задание 2». Скопируйте на лист исходные данные, оформите шапку.

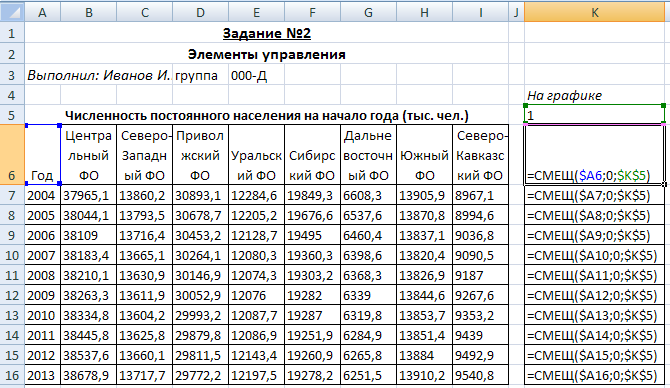


1. Создадим график с выводом одного ряда (столбца из таблицы) с выбором через выпадающий список. Результат должен выглядеть приблизительно так:

Чтобы реализовать такой график без макросов, потребуется отдельный столбец, в который будут копироваться данные для выбранного ФО из исходной таблицы. На графике будет отображаться именно этот столбец.

Разместите столбец справа от исходной таблицы (столбец K). В ячейку над столбцом (K5) введите 1 – это номер выбранного столбца (отсчет идет с 1).

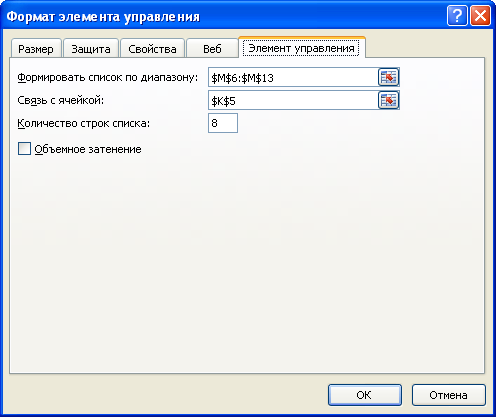
Сам столбец заполните с помощью формулы СМЕЩ. Эта формула копирует данные из ячейки, смещенной на указанное число строк и столбцов. Нам необходимо сместить данные на K5 столбцов от A5. По строкам смещения нет:



Добавьте на лист элемент управления «Поле со списком». Ему необходимо указать ячейки со списком значений (диапазон для формирования списка) и ячейку, в которую будут выводиться номер выбранного пункта (связь с ячейкой).

К сожалению, список можно сформировать только из значений, записанных в столбик, поэтому шапка таблицы не подойдет. Скопируйте ее и вставьте отдельно с транспонированием, например в диапазон M6:M13.

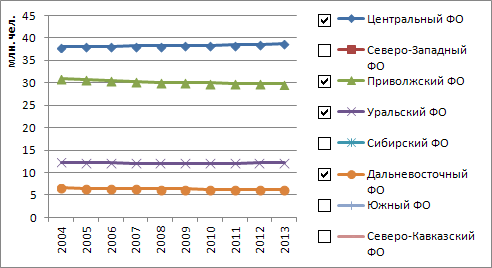
Щелкните полю со списком правой кнопкой и выберите «Формат объекта». Укажите созданный диапазон для формирования списка и назначьте связь с ячейкой K5.



Теперь можно добавить сам график. Выделите созданный дополнительный столбец и добавьте на лист «График с маркерами».

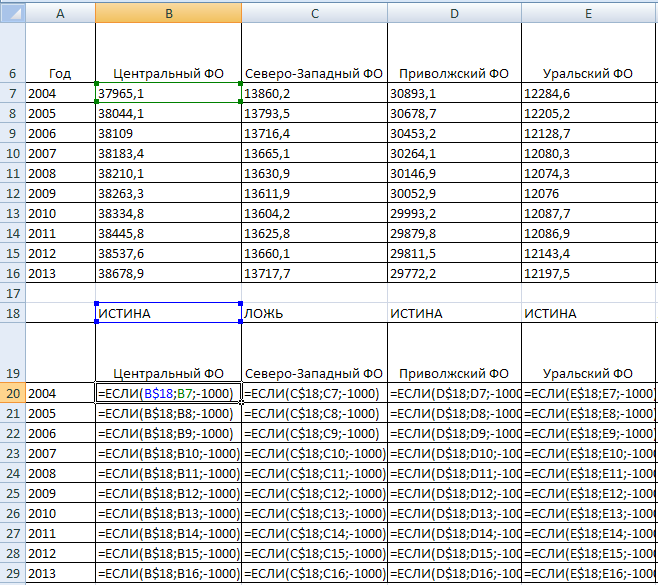
Настройте оформление графика. Чтобы вспомогательных ячеек не было видно, их можно скрыть или выбрать цвет шрифта – белый.

1. Другой график с возможностью выбора рядов для отображения должен выглядеть приблизительно так:



Формировать его будем тоже с помощью вспомогательных ячеек, но теперь потребуется копия всех столбцов таблицы.

Заполнять ее будем с помощью функции ЕСЛИ: если в связанной ячейке стоит ИСТИНА (1), то взять значение из исходной таблицы; иначе (ЛОЖЬ = 0) – отрицательное значение, например -1000.



Добавьте график и флажки. Флажки свяжите с ячейками от B18 до I18. Чтобы ряды с отрицательными числами не были видны на графике, установите минимальное значение по вертикальной оси равным 0.

Настройте оформления графика, скройте вспомогательные ячейки.

# Редактирование макросов через VBA

## Задание

Разработать макрос для табулирования функции:

.

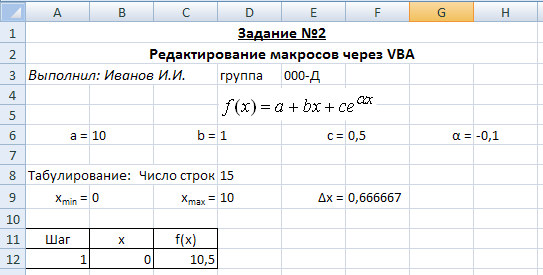
Значения параметров, диапазон и шаг табулирования задаются пользователем.

## Указания к выполнению

Макросы могут быть написаны на VBA «с нуля», но мы пойдем более простым путем – отредактируем записанный макрос, добавив нужные строки кода. В этом случае не потребуется хорошее знание языка Basic, достаточно основ синтаксиса и знания общих принципов программирования.

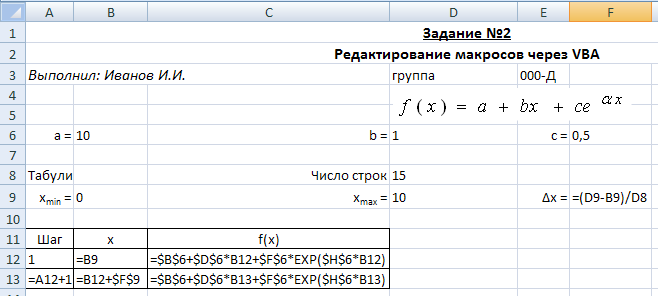
Мы запишем макрос для копирования первой строки таблицы, а затем дополним его циклом для заполнения указанного пользователем числа строк.

1. Создайте новый лист и назовите его «Задание 3». Разместите на листе «шапку», формулу функции и ячейки для ввода параметров. Заполните названия столбцов и первую строку итоговой таблицы. Оставьте место для кнопки.



1. Запишем макрос, который будет заполнять вторую строку таблицы. В дальнейшем нам потребуется лишь дописать цикл для повторения этих действий.

Начните запись макроса. Заполните ячейки A13:C13 формулами. Формулу для f(x) лучше ввести вручную или скопировать из ячейки C12, но не растягивать. Настройте оформление (границы ячейки). На этом запись макроса закончена.



1. Рассмотрим исходный код записанного макроса. Откройте окно «Макросы», выделите созданный макрос в списке и нажмите «Изменить».

Вот исходный код с подробными комментариями. Комментарии в VBA обозначаются знаком апострофа '. Точка с запятой после команд **не** ставится. Присваивание записывается через =, а не :=. Текстовые константы пишутся в двойных кавычках "".

Sub Табулирование() 'начало подпрограммы (макроса)

'

' Табулирование Макрос

' Автор: Коробецкая А.А.

'

'================= ВВОД ФОРМУЛ В ЯЧЕЙКИ ==================

Range("A13").Select 'выделение ячейки А13 (шаг)

ActiveCell.FormulaR1C1 = "=R[-1]C+1" 'ввод формулы в выделенную ячейку (А13)

Range("B13").Select 'выделение ячейки B13 (x)

ActiveCell.FormulaR1C1 = "=R[-1]C+R9C6" 'ввод формулы в ячейку B13

Range("C12").Select 'выделение ячейки C12 (f(x), первая строка)

Selection.Copy 'копирование выделения

Range("C13").Select 'выделение ячейки C13 (f(x), вторая строка)

ActiveSheet.Paste 'вставка скопированной формулы

'================= НАСТРОЙКА ФОРМАТА ЯЧЕЕК ==================

Range("A13:C13").Select 'выделение ячеек в строке 13

Application.CutCopyMode = False 'одновременно отменяется копирование

'отключение диагональных линий

Selection.Borders(xlDiagonalDown).LineStyle = xlNone

Selection.Borders(xlDiagonalUp).LineStyle = xlNone

'для выделенных ячеек, левая граница

With Selection.Borders(xlEdgeLeft)

'применить настройки

.LineStyle = xlContinuous 'сплошная линия

.ColorIndex = 0 'цвет - черный

.TintAndShade = 0 'без теней и объема

.Weight = xlThin 'толщина - тонкая

End With 'конец блока with

'аналогично для верхней границы

With Selection.Borders(xlEdgeTop)

.LineStyle = xlContinuous

.ColorIndex = 0

.TintAndShade = 0

.Weight = xlThin

End With

'аналогично для нижней границы

With Selection.Borders(xlEdgeBottom)

.LineStyle = xlContinuous

.ColorIndex = 0

.TintAndShade = 0

.Weight = xlThin

End With

'аналогично для правой границы

With Selection.Borders(xlEdgeRight)

.LineStyle = xlContinuous

.ColorIndex = 0

.TintAndShade = 0

.Weight = xlThin

End With

'аналогично для внутренних вертикальных границ

With Selection.Borders(xlInsideVertical)

.LineStyle = xlContinuous

.ColorIndex = 0

.TintAndShade = 0

.Weight = xlThin

End With

'аналогично для внутренних горизонтальных границ

With Selection.Borders(xlInsideHorizontal)

.LineStyle = xlContinuous

.ColorIndex = 0

.TintAndShade = 0

.Weight = xlThin

End With

End Sub 'конец подпрограммы (макроса)

Вот перечень обозначений и команд, который можно вынести из этого кода:

|  |  |
| --- | --- |
| Sub  ...  End Sub | Подпрограмма (процедура, макрос) на VBA |
| Range("A13") | Диапазон ячеек (можно указать в скобках любой адрес или имя ячейки) |
| ActiveSheet | Текущий лист |
| Range("A13:C13").Select | Выделить указанный диапазон |
| .FormulaR1C1 | Ввод формулы относительно текущей ячейки (R – row, C – Column)  Например, "R[-1]C0" или "R[-1]C" – ячейка на одну строку выше и в том же столбце  "R26C5"– ячейка на пересечении 26 строки и 5 столбца (E26) |
| Selection | Текущее выделение |
| .Copy | Команда копирования. Можно использовать для Selection, можно непосредственно для Range() |
| .Paste | Команда вставки. Используется для листа в целом, а не диапазона ячеек |

В VBA, также как и в большинстве других языков программирования присутствуют циклы For и While, оператор ветвления If. Нам достаточно цикла For:

**For** *переменная* = начальное\_значение **To** конечное\_значение

...

**Next** 'конец цикла

'ввести в каждую первую ячейку строк с 15 по 100 ее номер

**For** i = 15 **To** 100

Cells(i, 3).Value = i

**Next**

Обратите внимание на конструкцию Cells(i, 3).Value. В дальнейшем нам придется заменить Range() на Cell(), поскольку мы знаем номер строки и столбца, а не адрес ячейки.

Переменные в VBA объявляются по ключевому слову Dim. Тип переменных не указывается:

**Dim** i

**Dim** x, y, z

Осталось узнать, сколько строк нужно заполнить. Это число введено в ячейку D8, нужно просто записать его в переменную. Получим следующую конструкцию:

**Dim** r, rmax

rmax = Range("D8").Value + 11 'считываем докуда заполнять строки (с 12 до нужного кол-ва)

'ячейки заполняются с 13 строки до rmax, 12 строка уже заполнена

**For** r = 13 **To** rmax

'здесь выполняем все операции из кода выше

**Next**

Осталось добавить еще один момент – очистку ячеек. Поскольку мы не знаем, сколько строк было заполнено ранее, просто очистим достаточно большой диапазон.

Range("A13:C1000").Clear

В итоге получим следующий код:

Sub Табулирование() 'начало подпрограммы (макроса)

'

' Табулирование Макрос

' Автор: Коробецкая А.А.

'

' Очистка ячеек

Range("A13:C1000").Clear

Dim r, rmax

rmax = Range("D8").Value + 11

'ячейки заполняются с 13 строки до rmax, 12 строка уже заполнена

For r = 13 To rmax

'================= ВВОД ФОРМУЛ В ЯЧЕЙКИ ==================

Cells(r,1).Select

ActiveCell.FormulaR1C1 = "=R[-1]C+1"

Cells(r,2).Select

ActiveCell.FormulaR1C1 = "=R[-1]C+R9C6"

Range("C12").Select 'это ячейка – источник формулы, она не меняется

Selection.Copy

Cells(r,3).Select

ActiveSheet.Paste

'================= НАСТРОЙКА ФОРМАТА ЯЧЕЕК ==================

Range(Cells(r,1), Cells(r,3)).Select'выделение диапазона через первую и последнюю ячейку

Application.CutCopyMode = False

Selection.Borders(xlDiagonalDown).LineStyle = xlNone

Selection.Borders(xlDiagonalUp).LineStyle = xlNone

With Selection.Borders(xlEdgeLeft)

.LineStyle = xlContinuous

.ColorIndex = 0

.TintAndShade = 0

.Weight = xlThin

End With

With Selection.Borders(xlEdgeTop)

.LineStyle = xlContinuous

.ColorIndex = 0

.TintAndShade = 0

.Weight = xlThin

End With

With Selection.Borders(xlEdgeBottom)

.LineStyle = xlContinuous

.ColorIndex = 0

.TintAndShade = 0

.Weight = xlThin

End With

With Selection.Borders(xlEdgeRight)

.LineStyle = xlContinuous

.ColorIndex = 0

.TintAndShade = 0

.Weight = xlThin

End With

With Selection.Borders(xlInsideVertical)

.LineStyle = xlContinuous

.ColorIndex = 0

.TintAndShade = 0

.Weight = xlThin

End With

With Selection.Borders(xlInsideHorizontal)

.LineStyle = xlContinuous

.ColorIndex = 0

.TintAndShade = 0

.Weight = xlThin

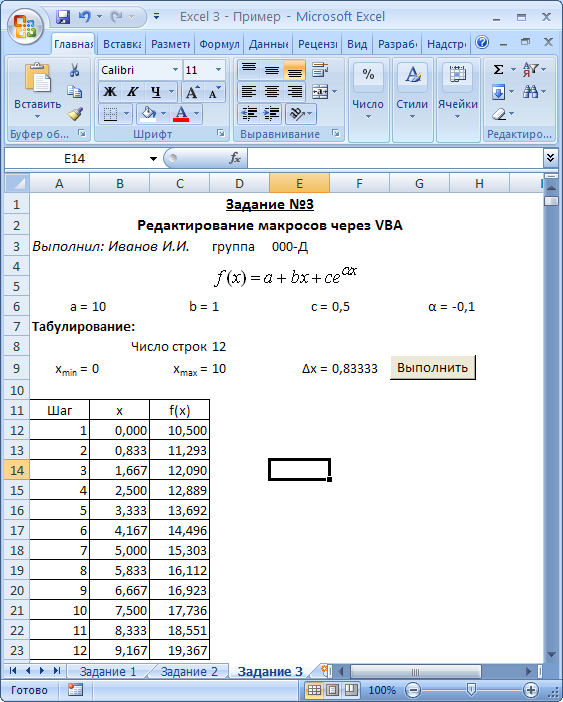
End With

Next

End Sub

Вы можете просто вставить этот код вместо исходного.

Осталось прикрепить макрос к кнопке и настроить оформление листа.



Задания для продвинутых

1. Добавьте в код выделение первой ячейки таблицы после завершения цикла.
2. Сократите код, убрав лишние действия. Формулы можно вводить и не выделяя ячейку, формат применять ко всему диапазону целиком.
3. Измените код так, чтобы значение Δx не изменялось после ввода другого числа ячеек на листе.
4. Добавьте проверку на число строк в таблице. Оно должно быть больше 1 и меньше числа очищаемых ячеек. Если пользователь ввел неправильное число, выведите сообщение в какую-нибудь ячейку.

Для этого используйте оператор ветвления:

**If** *условие* **Then**

*команды\_если\_выполняется*

**Else** *команды\_иначе*