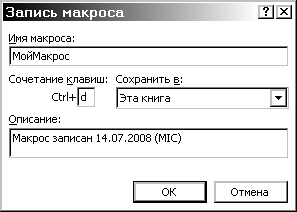
## VBA в Excel

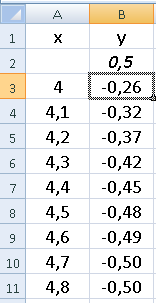
Теперь, когда вы уже знакомы с записью и редактированием макросов в среде Word, познакомимся с возможностями VBA в Excel. Не будем делать мелкие упражнения. Поставим более объёмную задачу, в процессе решения которой, можно увидеть основные принципы работы в электронных таблицах.

##### Особенности окна записи макроса.

Работа с макрорекодером в электронных таблицах Excel имеет небольшие отличия по сравнению с работой в Word. Диалоговое окно записи предлагает задать сочетание клавиш и определить место сохранения макроса. Будем сохранять макросы вместе с книгой. Таким образом, в отличие от приложения Word не предлагается лишь создавать кнопку.

Пока мы совершаем действия, которые записывает макрорекодер, видимой становится панель Остановить Запись с одноимённой кнопкой. На этой панели также есть кнопка-переключатель с относительной на абсолютную запись. Об этой возможность речь пойдёт позже. В Excel 07 эти возможности только в меню Макросы.

### Задача построения графика функции

Электронные таблицы дают нам возможность делать всевозможные расчёты, используя большой набор встроенных функций, и представлять данные в удобном текстовом и графическом виде. Для первоначального знакомства с возможностями использования VBA в Excel поставим задачу: **автоматизировать построение графика функции** из определённого нами набора. Конкретизируем, как именно мы будем это делать.

На рисунке показан фрагмент данных, подготовленных для построения графика функции. Это получится в результате трёх шагов.

Обычно построение графика состоит из следующих шагов:

1. Ввести имена переменных, например, X и Y.
2. Ввести два значения переменной X и с помощью автозаполнения арифметической прогрессией (пометка двух значений и протяжка за черный крест в правом нижнем углу) заполнить столбец необходимой длины.
3. Ввести формулу (на рисунке в ячейку B3). Если у функции есть параметры, рекомендуется оставить для них место над столбцом значений Y (ячейка B2). Копировать формулу по столбцу (протяжка за черный крест).
4. Пометить значения введённых переменных и подписи. Построить с помощью мастера функций стандартную точечную диаграмму со значениями, соединёнными сглаживающими линиями.

Для того чтобы легко было разобраться в записях макрорекодера, не будем выполнять все операции сразу. Начнем запись макроса с первого пункта.

##### Ячейки, диапазоны, строки и столбцы.

#### Запись первого шага.

Введем в ячейки A1и B1 соответственно имена переменных, например, X и Y. Посмотрим, что у нас получилось:

Sub Макрос()

Range("A1").Select ‘ выбрали ячейку A1

ActiveCell.FormulaR1C1 = "X" ‘ ввели значение X

Range("B1").Select ‘ выбрали ячейку B1

ActiveCell.FormulaR1C1 = "Y" ‘ ввели значение Y

Range("B2").Select ‘ выбрали ячейку B2

End Sub

Из этого можно сделать такие заключения:

1. Range – это диапазон (в частности, состоящий из одной ячейки, который может быть выбран Select, т.е. помечен).
2. ActiveCell – активная клетка, в которую можно ввести формулу FormulaR1C1. Текст и число являются частным видом формул, в этом мы убедимся в дальнейшем.

Задания для самостоятельной работы

1. Запишите макрос, который будет изменять ширину нескольких колонок.
2. Запишите макрос, который будет изменять высоту нескольких строк.
3. Попробуйте записать макросы, изменяющие параметры (цвет, узор или размеры ячеек) прямоугольного диапазона.

Замечания. После выполнения этих заданий в редакторе Вы увидите использование не только объекта Range, но и Columns – колонки, Rows – строчки при выборе объекта. Например:

Range("B14:C17").Select ‘ диапазон

Columns("F:F").Select ‘ колонка F

Rows("10:13").Select ‘ строки с 10 по 13

Далее будет следовать конструкция, подобная приведённой ниже:

With Selection.Interior

.ColorIndex = 41

.Pattern = xlVertical

End With

Мы уже знакомы с конструкцией With … End With, она позволяет задать внутри этих «скобок» сразу несколько свойств объекта. Объект Selection.Interior – интерьер или внутренняя область выделенного. Свойства в данном случае – это ColorIndex (номер цвета) и Pattern (узор). Узор устанавливается из меню Формат🡪 Вид🡪 Узор. Вместо номера цвета можно использовать свойство цвет и определить его традиционно с помощью палитры RGB, например:

With Selection.Interior

.Color = RGB(255, 255, 0) ‘ желтый цвет

End With

##### Автозаполнение

#### Запись второго шага

Введём два значения переменной X: в ячейку A3 – число 4, в ячейку A4 – значение 4.1. Затем с помощью автозаполнения арифметической прогрессией (помечаем два введённых значения и протягиваем за черный крест в правом нижнем углу) заполнить столбец до двадцатой строки. Excel определяет шаг прогрессии автоматически как разницу двух последовательных значений. В данном случае он равен 0,1.

Итак, должен получиться следующей макрос:

Sub Автозаполнение()

Range("A3").Select

ActiveCell.FormulaR1C1 = "4"

Range("A4").Select

ActiveCell.FormulaR1C1 = "4.1"

Range("A3:A4").Select

Selection.AutoFill Destination:=Range("A3:A20"), \_

Type:=xlFillDefault

Range("A3:A20").Select

End Sub

Первые пять строк макроса аналогичны уже рассмотренным выше. Метод AutoFill (автозаполнение) имеет один важный для нас параметр: Destination – назначение, т.е. диапазон, на который происходит автозаполнение.

Замечание: строку

Selection.AutoFill Destination:=Range("A3:A24"), \_ Type:=xlFillDefault

можно упростить. Макрорекодер записывает полную команду. Её структура такова:

Объект.Метод Параметр1:=значениеПараметра1, \_ Параметр2:=значениеПараметра2

У разных методов может быть различное число параметров. При краткой форме записи название параметра можно упустить, оставив только значение. Если какой-то параметр вы не указываете, то берётся его значение по умолчанию. Итак, в нашем случае команда может приобрести вид:

Selection.AutoFill Range("A3:A24")