SF EDUCATION

Программирование на VBA

Вводный урок. Функции и процедуры (Вводные) Часть 3

Содержание

1. Элемент «Сортировка» Sort
2. Создание рабочей программы
3. Поэтапный алгоритм процедуры сортировки

1. Элемент «Сортировка» Sort

Особенность элемента «Сортировка» Sort – наличие большого количества параметров.

Sort

expression.**Sort** (*Key1*, *Order1*, *Key2*, *Type*, *Order2*, *Key3*, *Order3*, *Header*, *OrderCustom*, *SortMethod*, *DataOption1*, *DataOption2*, *DataOption3*)

Разберем эти параметры и применим их на практике

Имеем в Excel рандомные данные из трех портфелей. Нам нужно их отсортировать и в дальнейшем использовать.

Процедура сортировки:

- Выберем ячейку 1 - **Range (“A 1”). Select** для того, чтобы после этого выбрать всю таблицу – ActivtCell. CurrentRegion. Select;
- Таким образом мы сможем выбрать весь необходимый объем таблиц, которые хотим сортировать;
- Получили формулу

```
Sub sort()
Range("A1").Select
ActiveCell.CurrentRegion.Select
End Sub
```

После включения макроса будем сортировать таблицу:

- Напишем строку в формулу: Selection. Sort;
- **Key 1**

  Key 1 **–** это обязательный параметр.

  Тип этого параметра – Range, то есть по сути - это ячейка. Параметров **Key 1 у** нас всего три. Это означает, что значения, возможно, могут совпадать. И чтобы ранжировать один параметр от другого, в строке формулы мы можем указать также **Key 2 и Key 3 –** то есть те ключи, на которые будет ссылаться **vba** при сортировке;
- **“D 1”**

  “D 1” – ячейка D 1 - этот параметр создается для сортировки таблицы.
  По этой ячейке по столбцу D будет производиться сортировка;
  Сортировка бывает, как по столбцам, так и по строкам. Поэтому мы выбираем ячейку – первую ячейку по строке и по столбцу;
- **Order**

  Указываем, каким образом мы будем сортировать – по возрастанию или по убыванию. Применяем здесь параметр Order, их также 3 типа -
  Order 1, Order 2 и Order 3. Каждый из этих типов относится к Key.
  Order бывает двух типов – сортировка по возрастанию и сортировка по убыванию. Если не указываем ордер, ставится значение по умолчанию – сортировка по возрастанию – Order 1 : = x 1 Ascending. В нашем случае ставим по убыванию, пишем - Order 1 : = x 1 Descending;
- **Header**

  Это заголовок таблицы или наименование каждого из столбцов. Так его и прописываем в нашу формулу. Если в нашей таблице есть строка заголовков, то в формуле запишем Yes. Если в строке заголовков не

подразумевается – пишем No.

Мы можем в нашей строке формулы написать - Header : = x 1 Guess. Это будет обозначать, что Excel сам должен определить наличие заголовка и записать его;

- **OrderCustom**

  Пропишем OrderCustom : = 1. Это для того, чтобы была произведена сортировка именно по возрастанию;

- **MatchCase**
- Это сортировка, учитывающая регистр: MatchCase : = False (или True);

Далее по характеристике сортировок:

- **Orientation**

  Orientation – это сортировка по столбцу или по строке.

  Если сортируем по строке, пишем Orientation : = x 1 LeftToRight.

  Если сортируем по столбцу, пишем Orientation : = x 1 TopToBottom;

- **DataOption**

  Это один из необязательных параметров, который задает способ сортировки текста.

  SortNormal – это текстовые и числовые значения.

  SortTextAsNumbers – берет текст и присваивает ему значения, которые сортируются как числа.

  Запишем:

  DataOption 1 : = x 1 SortTextAsNumbers - если мы хотим, чтобы наши данные по сортировке рассматривались как числовые.

  DataOption 1 : = x 1 SortNormal – если текстовые данные отсутствуют;

- **Запускаем наш макрос:**

```
Sub sort()
Range("A1").Select
ActiveCell.CurrentRegion.Select
Selection.sort Key1:=Range("D1"), Order1:=xlDescending, Header:=xlGuess, _
    OrderCustom:=1, MatchCase:=False, Orientation:=xlTopToBottom, _
    DataOption1:=xlSortNormal
End Sub
```

Процедура отсортировала наш макрос.

Такой способ сортировки с записью макроса очень часто применяется.

Существует еще один способ сортировки.

Если вы забыли какие-либо параметры, вы можете обратиться на официальный сайт Microsoft: docs.microsoft.com в разделе VBA. Здесь прописаны все возможные объекты, которыми можно воспользоваться, все сортировочные параметры.

2. Создание рабочей программы

Здесь мы дополнительно изучим такую важную функцию, как **Coll,** то есть вызов. Мы запишем несколько процедур и в некоторых процедурах будем вызывать другие процедуры.

Сдвиг при добавлении столбца и строк

Запишем структуру формулы:

- Нам нужно добавить одну колонку (столбец):

  Columns (“A”). Select

  Selection. Insert Shift : = xltoRight;

- Добавим несколько строк:

  Rows (“1 : 6”) . Select

  Selection. Insert Shift : = xlDown;

```
Sub InsRowCol()

Columns("A").Select
Selection.Insert shift:=xlToRight
Rows("1:6").Select
Selection.Insert shift:=xlDown
End Sub
```

- Пропишем процедуру, которая будет редактировать слова

  Sub redText ()

  Range (“A1”) = “Купленные компании”

  Range (“A2”) = “Цена акций”

Range (“B4”) = “Портфель”

Range (“B6”) = “Name”

Range (“C6”) = “Цена открытия”

Range (“D6”) = “Цена закрытия”

Range (“E6”) = “Net”

- Пропишем процедуру, редактирующую шрифт;

  Прописываем ячейки и колонки, редактирующие параметры шрифта:

```
End With
Columns("B:E").Select
Selection.Style = "Currency"
Columns("B:E").Select
Selection.Columns.AutoFit
End Sub
```

Итак, у нас получилось 4 пакета данных. Они не отредактированы.

Как правильно вызвать выполнение процедуры:

- Пропишем самую главную процедуру Name. Она будет вызывать все оставшиеся процедуры;

```
Sub main()
For x = 1 To Worksheets.Count - 1
Worksheets(x).Select
Call sort
Call InsRowCol
Call redText
Call redText2
Range("B6").Select
Selection.CurrentRegion.Select
Selection.Copy

Worksheets("All").Select

Range("B10000").Select
Selection.End(xlUp).Select
ActiveCell.Offset(2, 0).Select
ActiveCell.Value2 = Worksheets(x).Name & " Портфель"
ActiveCell.Offset(2, 0).Select
ActiveSheet.Paste
Next x
```

- Теперь изменяем наши колонки по количеству элементов в ячейке:

```
Columns("B:E").Select
Selection.Columns.AutoFit
End Sub
```

3. Поэтапный алгоритм процедуры сортировки

- Запускаем наш макрос;
- Начинается подсчет X;
- Активируется первый лист;
- Из процедуры Col перескакиваем на процедуру Sort;
- Запускаем процедуру InsRpwCol - добавляем строки;
- Запускаем RedText – вставляем необходимый текст;
- Запускаем RedText 2 – изменяем форматы текста;
- Выбрали всю нашу таблицу, скопировали ее;
- Активируем Worksheets (“All”). Select;
- Выбираем ячейку B 10 000;
- Поднимаемся наверх, используя Ctrl;

- Получили самую последнюю строчку → сдвигаемся от нее на две строки вниз;
- Вставляем в командную строку (наверху) текст «А Портфель»;
- Сдвигаемся еще вниз (на 5-ю строчку) и вставляем туда все, что скопировали в следующий Next;
- Запускаем Sort → отсортировали;
- Запускаем добавление нескольких строк (6 строк и 1 столбец);
- Отредактируем текст – Coll redText;
- Второе редактирование формата текста;
- Подбираем регион, который соответствует нашей таблице, копируем его;
- Открываем нашу таблицу → CTRL Shift → закидываемся наверх;
- Активируем третий лист → отсортируем, запустив Sort;
- Запускаем форматирование – добавление 1 столбца и 6 строк;
- Вставляем нужный текст;
- Запускаем редактирование форматов самого шрифта;
- Отредактируем колонки по ширине и длине;
- Скопировали таблицу отфармотированную;
- Закидываемся наверх → B 1000, спускаемся на 2 ячейки вниз;
- Вставляем имя листа и слово Портфель;
- Спускаемся на 2 ячейки вниз → вставляем весь Портфель4
- Выделяем все наши столбцы и делаем autofit, чтобы они были читабельные;
- Получаем окончательный лист с таблицами.

ИТОГ

Таким образом, мы сделали на одном листе таблицу. У вас может быть тысячи таких листов, и вы можете всю эту информацию расположить на одном листе, используя относительные ссылки.

Мы каждую отдельную процедуру записали в какое-то определенное действие и просто вызывали их тогда, когда они нам были нужны.