SF EDUCATION

Программирование на VBA

Свойства и методы. Часть 1

Содержание

1. Пример работы с ячейками и объектами VBA
2. Получить таблицу

1. Пример работы с ячейками и объектами VBA

Перейдем к небольшому заданию.

У нас на рабочем столе есть папка «Дубликат» с определенным документом. Нам нужно:

1. создать копию этого документа;
2. сохранить эту копию в эту же папку;
3. открыть эту копию документа;
4. в этой копии на соответствующих местах – от ячейки B 3 до ячейки F 12 – должна располагаться такая же таблица.

2. Получить таблицу

Характеристики и значения в скопированной таблице должны быть такие же, как и в начальной таблице.
То есть, без заполнения самой таблицы, в скопированной таблице будут располагаться те же самые значения.

Дата	Итог	Компания	Валюта	Операция
08.08.2020	208,8	ASOS	$	*Списание*
07.08.2020	510	Перекресток	₽	*Списание*
07.08.2020	60	CoFix	₽	*Списание*
07.08.2020	150	Пекарня	₽	*Списание*
06.08.2020	300	FixPrice	₽	*Списание*
06.08.2020	400	Лента	₽	*Списание*
06.08.2020	300	McDonald's	₽	*Списание*
06.08.2020	500	Яндекс Такси	₽	*Списание*
05.08.2020	55000	ЗП	₽	*Поступление*

SF EDUCATION

Программирование на VBA

Свойства и методы. Часть 2

Содержание

1. Открытие книги. Пример копирования таблицы с сохранением параметров
2. Сохранение копии продукта
3. Закрытие книги

1. Открытие книги. Пример копирования таблицы с сохранением параметров

У нас на рабочем столе есть папка «Дубликат» с определенным документом. Нам нужно:

1. создать копию этого документа;
2. сохранить эту копию в эту же папку;
3. открыть эту копию документа;
4. в этой копии на соответствующих местах – от ячейки B 3 до ячейки F 12, должна располагаться такая же таблица.

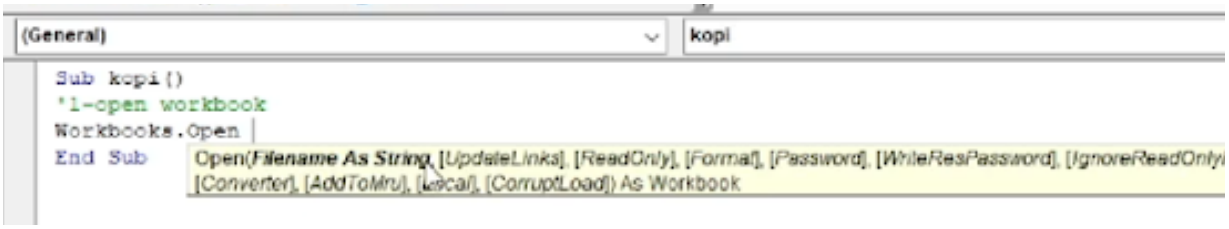
Характеристики и значения в скопированной таблице должны быть такие же, как и в начальной таблице.

То есть, без заполнения самой таблицы, в скопированной таблице будут располагаться те же самые значения.

Приступим к выполнению задания.

Открыть книгу с рабочего стола:

- Заходим в Разработчик → Visual Basic → Insert → Module;
- Создадим наш макрос – Module 1 → Назовем имя макроса Sub copi (). Наш макрос сам все дальше допишет;
- Записываем ходы: ‘ 1 open workbook → Workbooks. Open;
- Excel поможет нам определиться, какую выбрать книгу.



Что из подсказки Excel нужно нам прописать, чтобы открылась книга? Что обязательно прописывать, а что необязательно?

Если мы видим, что запрашиваемый элемент находится в формуле в квадратных скобках, то эта информация считается дополнительной.

Если запрашиваемая информация заключена в круглые скобки, то она считается обязательной.

Например, из формулы видно, что **Filename As String** – обязательный элемент для прописания, так как он находится внутри круглых скобок.

Итак, нам нужно прописать путь и имя файла, к которому мы обращаемся.

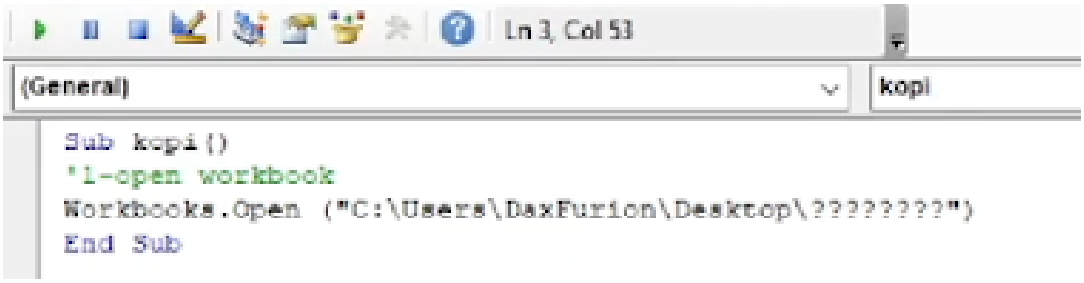
Сначала разберемся с путями:

- Наша книга Workbook располагается на рабочем столе.

  Здесь мы видим ее путь;



- Скопируем его и вставим в наш макрос:



  Сразу заметно, что как только встречаются русские буквы, то все они заменяются на вопросительные знаки «???»;

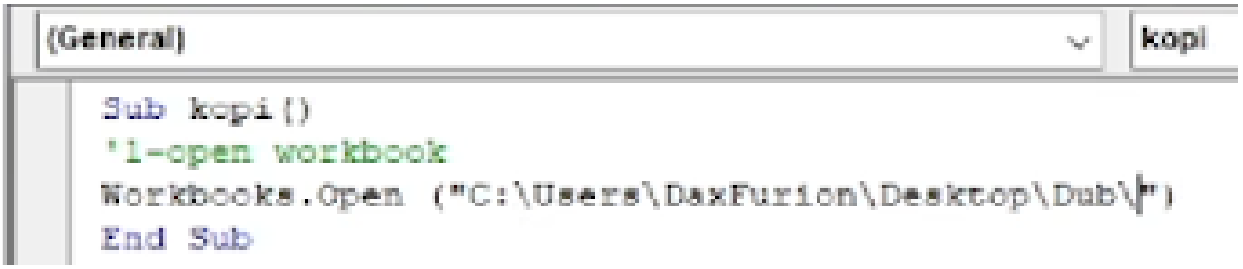
- Проблема заключается в том, что юникод - метод шифровки русских букв в Excel и в компьютере различаются. Поэтому Excel их просто не распознает. Здесь придется изменять вручную все папки с русскими буквами. Поэтому меняем шрифт на русский и пишем «Дубликат».

```
Sub kopi()
'1-open workbook
Workbooks.Open ("C:\Users\DaxFurion\Desktop\Дубликат")
End Sub
```

Русский язык в виде алгоритмического кода здесь абсолютно бессмыслен и считываться не будет. Но если слова на русском записаны в путях, то все отлично считается.

Второй способ обхода данной проблемы – замена имени папки на английский. Этот способ не подходит в случае, когда путь к папкам не зависит от нас. Например, если папка находится в базе данных на сервере, и мы не имеем возможности прямого доступа к ней и соответственно, не можем изменить ее название.

В этом случае мы меняем название нашей папки и записываем его английскими буквами:



После того, как мы прописали путь, нам необходимо обратиться к самому воркбуку (Workbook):

- Workbook в нашей папке называется **Book 1. xlsx.** Мы точно также должны его прописать;



- Протестируем строчку, которую мы прописали. Мы видим, что открылась книга Book 1 с таблицей, формат которой нам нужно скопировать;

- Теперь нам нужно пересохранить эту книгу с названием New Copi.

2. Сохранение копии продукта

И так, второй этап нашей работы заключается в сохранении новой копии нашего продукта.

Для этого проводим следующие действия:

- Обращаемся к книге Workbooks и прописываем полную формулу пути;
- При этом при сохранении книги, надо придумать ей имя;
- Таким образом, мы сохраним путь, который был у Книги 1 с добавлением нового имени.

```
'2-save newcopy
Workbooks("Book1.xlsx").SaveCopyAs (Workbooks("Book1.xlsx").Path & "\" & "NewCopy.xlsx")
End Sub
```

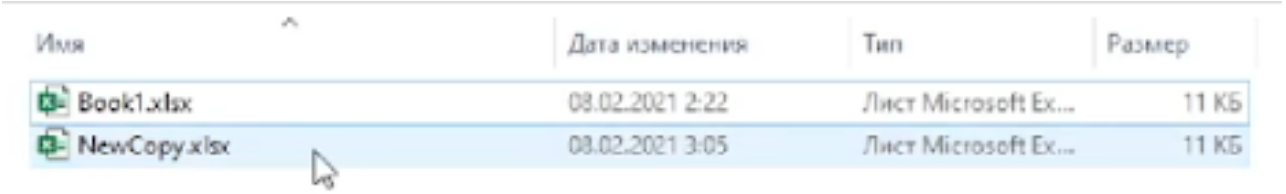
Workbooks(“Book1.xlsx”).SaveCopyAs(Workbooks(“Book1.xlsx”).Path &“&“NewCopy.xlsx”)

Таким образом, мы сохранили книгу в тех же самых путях, в которых находилась Книга 1 с названием New Copy.

Запускаем сохраненную книгу, смотрим открывшуюся таблицу:

Дата	Итог	Компания	Валюта	Операция
08.08.2020	208,8	ASOS	$	*Списание*
07.08.2020	510	Перекресток	₽	*Списание*
07.08.2020	60	CoFix	₽	*Списание*
07.08.2020	150	Пекарня	₽	*Списание*
06.08.2020	300	FixPrice	₽	*Списание*
06.08.2020	400	Лента	₽	*Списание*
06.08.2020	300	McDonald's	₽	*Списание*
06.08.2020	500	Яндекс Такси	₽	*Списание*
05.08.2020	55000	ЗП	₽	*Поступление*

Открываем папку и видим, что у нас есть новая копия, точно такая же, как и Book 1:



Операция по пересохранению книги нужна для того, чтобы все время не обращаться к книгам, которые находятся на серверах или в локальном доступе, и вручную не переносить их на рабочий стол. Таким способом можно автоматизировать и ускорить процесс работы.

Если был проработан макрос и книга сохранена, то ее никак нельзя вернуть на предыдущую версию.

3. Закрытие книги

На данный момент у нас еще открыта книга Book 1.

Книгу закроем аналогично тому, как открывали:

- Надо обратиться к книге Book 1. xlsx и закрыть ее
  Workbooks (“Book 1. xlsx”). Close;
- Смотрим строку для обязательного описания - ни одна из методик не обязательна к написанию;



- Это означает, что по факту книга не закроется, потому что к нам будет запрос – хотим ли мы сохранить старую копию книги в нашей папке.
  Поэтому мы не будем сохранять изменения в книге.
  Пишем:
  Workbooks (“Book 1. xlsx”). Close Save Changes: = False;
- Запускаем наш макрос → В папке появилась копия книги NewCopy. xlsx;

- Откроем эту копию. Она идентична таблице в нашей книге;
- Теперь в Книгу Book 1. xlsx нам нужно скопировать формат из книги NewCopy. xlsx.

Как скопировать формат из копии в книгу:

- Для того, чтобы начать копировать формат, откроем 4-ю книгу и скопируем туда название нашей первоначальной операции

```
'4-open workbooks
Workbooks.Open ("C:\Users\DaxFurion\Desktop\Dub\NewCopy.xlsx")
```

Запишем формулу: в книгу Workbooks, которая называется “Книга 1. xlsx” мы должны скопировать данные таблицы в ячейках от B 3 до F 12

14 – 26. Прописываем макросы.

Мы сделали оболочку нашей таблицы:

```
'4-open workbooks
Workbooks.Open ("C:\Users\DaxFurion\Desktop\Dub\NewCopy.xlsx")
Workbooks("Книга1.xlsx").Worksheets("Лист1").Range("B3:F12").Borders(xlEdgeBottom).LineStyle = x
Workbooks("Книга1.xlsx").Worksheets("Лист1").Range("B3:F12").Borders(xlEdgeTop).LineStyle = xlCo
Workbooks("Книга1.xlsx").Worksheets("Лист1").Range("B3:F12").Borders(xlEdgeLeft).LineStyle = xlC
Workbooks("Книга1.xlsx").Worksheets("Лист1").Range("B3:F12").Borders(xlEdgeRight).LineStyle = xl

End Sub
```

Теперь нам нужно прописать формулы для внутреннего содержания таблицы:

```
\Desktop\Dub\NewCopy.xlsx")
("Лист1").Range("B3:F12").Borders(xlEdgeBottom).LineStyle = xlContinuous
("Лист1").Range("B3:F12").Borders(xlEdgeTop).LineStyle = xlContinuous
("Лист1").Range("B3:F12").Borders(xlEdgeLeft).LineStyle = xlContinuous
("Лист1").Range("B3:F12").Borders(xlEdgeRight).LineStyle = xlContinuous
("Лист1").Range("B3:F12").Borders(xlEdgeBottom).LineStyle = xlContinuous
```

```
op\Dub\NewCopy.xlsx")
1").Range("B3:F12").Borders(xlEdgeBottom).LineStyle = xlContinuous
1").Range("B3:F12").Borders(xlEdgeTop).LineStyle = xlContinuous
1").Range("B3:F12").Borders(xlEdgeLeft).LineStyle = xlContinuous
1").Range("B3:F12").Borders(xlEdgeRight).LineStyle = xlContinuous
1").Range("B3:F12").Borders(xlInsideHorizontal).LineStyle = xlContinuous
1").Range("B3:F12").Borders(xlInsideVertical).LineStyle = xlContinuous
```

ИТОГ

Закрытие книги происходит посредством создания копии продукта