

EXCEL И ЕГО ОБЪЕКТЫ

Основные понятия языка VBA Excel

На занятии рассматриваются следующие вопросы:

- **объекты VBA;**
- **свойства, методы и события;**
- **элементы языка VBA;**
- **структура объектов VBA.**

Объекты

- Объект - основной элемент VBA Excel. В VBA объектами являются рабочая книга, рабочий лист и его составляющие.

Примеры объектов:

- **Sheet** - лист Excel;
 - **Cell** - ячейка;
 - **Range** - диапазон ячеек;
 - **Application** - приложение;
 - **UserForm** - пользовательская форма.
-
- Доступ к объекту возможен через его методы и свойства

Методы

- Над объектами можно совершать различные действия. **Действия, совершаемые над объектами, называются методами.** Например, ячейку можно очистить (Clear), приложение закрыть (Quit), пользовательскую форму показать (Show) или скрыть (Hide).
- Название метода отделяется от названия объекта точкой: **объект.метод.**

Примеры использования методов:

- **Range("B2:E2").Select** - выбрать диапазон ячеек B2:E2;
- **Range("C1:C5").Clear** - очистить диапазон ячеек C1:C5;
- **UserForm2.Hide** - скрыть форму № 2;
- **UserForm5.Show** - показать форму № 5;
- **Application.Quit** - выйти из приложения.

Свойства

- Свойствами описываются характеристики объектов. Например, размер и цвет шрифта, положение формы на экране или состояние объекта (доступность, видимость). Чтобы изменить характеристику объекта, надо просто изменить значение свойства, т.е. присвоить ему определенные значения.
- Синтаксис установки значения свойства: **Объект.Свойство = ЗначениеСвойства**,
- где Объект обозначает имя объекта, Свойство - имя свойства, которому присваивается значение.
- Имя объекта отделяется от имени свойства точкой.

Примеры свойств:

- `Range("D1").Value = 2005` - поместить в ячейку D1 значение 2005.
- `Range("C1:C10").Text = "Информатика"` - поместить в диапазон ячеек C1:C10 текст Информатика.
- `Range("B2").Font.Size = 14` - в ячейке B2 установить размер шрифта 14.

События

- Событие представляет собой действие, распознаваемое объектом (например, щелчок мышью или нажатие клавиши, перемещение мыши или выход из программы), для которого можно запрограммировать отклик, т.е. реакцию объекта на произошедшее событие.
- В языке VBA для каждого объекта определен набор стандартных событий.
- **Стандартное событие для объекта “кнопка” (CommandButton) - щелчок мышью (Click).**
- Если пользователь нажимает на кнопку, то это событие. На это событие должен быть отклик, т.е. выполнение какой-либо программы. Такая программа называется процедурой обработки событий и имеет стандартное имя. Если такой отклик не создан (не написана соответствующая программа), то система никак не будет реагировать на это событие

Упрощённая структура приложения Excel

- **Application** (приложение)
 - **Workbooks** (коллекция рабочих книг)
 - **Workbook** (объект, представляющий файл рабочей книги Excel)
 - **Worksheets** (коллекция рабочих листов)
 - **Worksheet** (объект, представляющий рабочий лист книги Excel)
 - **Range** (ячейка, диапазон)

.....

Коллекция `Workbooks`
Объект `Workbook`

Создание рабочей книги

```
Sub Первый_пример()
```

```
Workbooks.Add
```

```
End Sub
```

Открытие существующей на диске рабочей книги

```
Sub Второй_пример()
```

```
Workbooks.Open _
```

```
Filename:="E:\Инструкц карты Excel\Новые файлы\Диаграмма.xls"
```

```
End Sub
```

- `Filename` – обязательный параметр задает путь и имя открываемого файла

Сохранение рабочих книг

```
Sub Третий_пример()  
Workbooks(2).SaveAs _  
Filename:="Моя рабочая книга"  
End Sub
```

- SaveAs – метод для сохранения новой рабочей книги в первый раз с присвоением ей имени или для сохранения книги под другим именем
- Save – метод для сохранения рабочей книги, которой уже было присвоено имя

```
Sub Четвертый_пример()  
Workbooks("Моя рабочая книга.xls").SaveCopyAs  
–  
Filename:="Моя вторая рабочая книга.xls"  
End Sub
```

- SaveCopyAs – метод для сохранения копии рабочей книги под другим именем, не изменяя при этом имя текущей книги

Заккрытие рабочих книг

```
Sub Пятый_пример()
```

```
Workbooks("Моя рабочая книга.xls").Close True
```

```
End Sub
```

- Close – метод для закрытия рабочей книги.

Аргументы для данного метода:

- SaveChanges задает вариант действий для случая, когда рабочая книга содержит несохраненные изменения (True, false), если этот аргумент опущен, пользователю будет предложено сохранить изменения.
- FileName – это имя файла, в который сохраняется рабочая книга, если этот аргумент опущен, используется текущее имя, если такового не существует, пользователю будет предложено его ввести.

Коллекция WorksheetS
Объект Worksheet

Добавление рабочих листов

```
Sub Шестой_и_седьмой_пример()  
Dim mysheet As Excel.Worksheet  
Set mysheet = Worksheets.Add  
mysheet.Name = "Моя страница"  
End Sub
```

- Аргументы являются необязательными. Если они опущены, создается один рабочий лист, сразу после активного. Аргументы Before и After используются для определения существующего рабочего листа, до или после которого нужно вставить новый. Для того чтобы добавить более одного рабочего листа, задайте нужное количество в аргументе Count. Если этот аргумент больше единицы, метод Add возвращает ссылку на последний из добавленных рабочих листов. Как всегда, желательно новым рабочим листам назначить имена, чтобы можно было впоследствии к ним обращаться в программе.

Удаление рабочих листов

```
Sub Восьмой_пример()
```

```
ActiveWorkbook.Worksheets("Моя страница").Delete
```

```
End Sub
```



```
Sub Девятый_пример()
```

```
Application.DisplayAlerts = False
```

```
ActiveWorkbook.Worksheets("Моя  
страница").Delete
```

```
Application.DisplayAlerts = True
```

```
End Sub
```

- Аргумент SheetName является именем удаляемого рабочего листа. Когда пользователь в программе пытается удалить рабочий лист, Excel обычно выводит ему запрос на подтверждение удаления. Для того чтобы этот запрос не выводился, нужно установить свойство DisplayAlerts объекта Application в значение False. После того как рабочий лист удален, важно этому свойству вернуть значение True

Копирование рабочих ЛИСТОВ

Sub Десятый_пример()

Application.Sheets.Copy

End Sub

Sub Одиннадцатый_пример()

**Application.Worksheets("Моя страница").Copy _
after:=Worksheets(Worksheets.Count)**

End Sub

- Если нужно скопировать рабочий лист в ту же рабочую книгу, используются аргументы Before или After для указания того существующего рабочего листа, до или после которого требуется вставить копию. Если эти аргументы опущены, рабочие листы копируются в новую книгу. В следующем фрагменте рабочий лист SalesData копируется в последнюю позицию текущей рабочей книги

Перемещение рабочих ЛИСТОВ

```
Sub Двенадцатый_пример()  
Application.Worksheets("Моя страница").Select  
Application.Worksheets("Моя страница").Move _  
after:=Sheets(1)  
End Sub
```

Объект Range

```
Sub Шестнадцатый_пример()  
Range("a1:c5").Select  
Range("c6").Activate  
End Sub
```

- ActiveSheet.Range ("A1").Value = 7
- ActiveSheet.Cells (1,1).Value = 7

Система Excel имеет два различных стиля обозначений для ссылок на диапазоны рабочей страницы. Обозначения стиля **A1** использует букву колонки, дополненную номером строки, где абсолютная ссылка обозначается знаком доллара (\$). Обозначения стиля **R1C1** имеют форму $RxCy$, где x и y являются номерами строки и столбца, соответственно. X и Y могут быть опущены, если нет связанного с ними смещения. Относительные ссылки в этом стиле заключаются в квадратные [] скобки. Следующие пары ссылок в стилях A1 и R1C1 являются эквивалентными:

\$A\$1	R1C1
A1	RC
A2	R[1]C
A1:C3	RC:R[2]C[2]
A2:C3	R[1]C:R[2]C[2]
\$A\$2:\$C\$3	R2C1:R3C2