# Практическая работа №8 Классы и объекты в Visual Basic for Application

## Тема и цель работы

Знакомство с объектами VBA. На практике рассмотреть основные операции и функции.

## Теоретический материал, для освоения темы

## Объектная модель

Объектная модель MS Excel представляет собой иерархию объектов, подчиненных одному объекту Application, который соответствует самому приложению Office. Многие из этих объектов собраны в библиотеке объектов конкретного приложения, но некоторые из них, например, объект Assistant, входят в библиотеку объектов Office, которая является общей для всех офисных приложений.

## Полная и неявная ссылка на объект

Полная ссылка на объект состоит из ряда имен вложенных последовательно друг в друга объектов. Разделителями имен объектов в этом ряду являются точки, ряд начинается с объекта Application и заканчивается именем самого объекта. Например, полная ссылка на ячейку А1 рабочего листа Продажи рабочей книги с именем Архив имеет вид:

Application.Workbooks("Архив").Worksheets("Продажи").Range("Al")

Приводить каждый раз полную ссылку на объект совершенно не обязательно. Обычно достаточно ограничиться только неявной ссылкой на объект. В неявной ссылке, в отличие от полной, объекты, которые активны в данный момент, как правило, можно опускать. В рассмотренном случае, если ссылка на ячейку А1 дана в программе, выполняемой в среде Excel, то ссылка на объект Application может быть опущена, т.е. достаточно привести относительную ссылку:

Workbooks("Архив").Worksheets("Продажи").Range("Al")

Если в этом примере ссылки рабочая книга Архив является активной, то ссылку можно еще сократить:

Worksheets("Продажи").Range("Al")

Если и рабочий лист Продажи активен, то в относительной ссылке вполне достаточно ограничиться упоминанием только диапазона А1:

Range("Al")

## Основные объекты VBA

Объект Application (приложение) является главным в иерархии объектов Excel и представляет само приложение Excel. Он имеет более 120 свойств и методов и методов. Эти свойства и методы предназначены для установки общих параметров приложения Excel. Кроме того, объект Application позволяет вызывать более 400 встроенных функций рабочего листа при помощи конструкции вида:

Application.ФункцияРабочегоЛиста (Аргументы)

Например, для определения постоянных периодических платежей при постоянной процентной ставке можно воспользоваться следующей функцией:

Application.Pmt (Аргументы)

## Свойства объекта Application

|  |  |
| --- | --- |
| ActiveWorkbook | Возвращает активный объект: рабочую книгу, лист, ячейку, диаграмму диалоговое окно. Свойство ActiveCell содержится в ActiveSheet, а свойства ActiveChart и ActiveDialog в ActiveWorkbook.  Например:  ActiveCell.Value = “Привет!”  Здесь в активную ячейку вводится фраза Привет! |
| ActiveSheet |
| ActiveCell |
| ActiveChart |
| ActiveDialog |
| ThisWorkbook | Возвращает рабочую книгу, содержащую выполняющийся в данный момент макрос. |
| Calculation | Устанавливает режим вычислений |
| Caption | Возвращает текст в строке имени главного окна Excel. Установка свойства равным Empty возвращает заголовок, используемый по умолчанию |
| DisplayAlerts | Значения: True – отображаются встроенные предупреждения о работе программы, False – не отображаются предупреждения |
| DisplayFormulaBar | Значения: True – строка формул выводится в окне Excel, False – не выводится |
| DisplayScrollBars | Значения: True – полоса прокрутки выводится в окне Excel, False – полосы прокрутки не отображаются |
| Height | Высота приложения в пунктах |
| Width | Ширина приложения в пунктах |
| Right | Расстояние в пунктах от правой границы окна приложения до правого края окна |
| Left | Расстояние в пунктах от левой границы окна приложения до левого края окна |
| Top | Расстояние в пунктах от верхней границы приложения |
| WindowState | Устанавливает размер окна |

## Методы объекта Application

|  |  |
| --- | --- |
| Calculate | Вызывает принудительное вычисление во всех открытых рабочих книгах. |
| Run | Запускает на выполнение программу или макрос. |
| OnKey | Устанавливает сочетание клавиш для выполнение выбранной процедуры.  OnKey(Key, Procedure) |
| Quit | Закрывает приложение. |
| OnTime | Назначает выполнение процедуры на определенное время. |

## События объекта Application

|  |  |
| --- | --- |
| NewWorkbook | При создании новой рабочей книги |
| WorkbookActivate | При активизации рабочей книги |
| WorkbookBeforeClose | Перед закрытием рабочей книги |
| WorkbookBeforePrint | Перед печатью рабочей книги. |
| WorkbookBeforeSave | Перед сохранением рабочей книги. |
| WorkbookNewSheet | При добавлении нового листа |
| WorkbookOpen | При открытии рабочей книги |

## Объект Workbook (книга)

## Свойства объекта Workbook

|  |  |
| --- | --- |
| ActiveSheet | Возвращает активный лист книги. |
| ActiveDialog | Возвращает активное диалоговое окно |
| ActiveChart | Возвращает активную диаграмму  Например: MsgBox “Название диаграммы” & ActiveChart.Name  выводит в диалоговом окне имя активной диаграммы |
| Saved | True – если не производились изменения в документе со времени его последнего сохранения, False – в противном случае. Например:  If WorkBook.Saved=True Then  MsgBox “ удачно сохранились! “  End If |
| WriteReserved | True – если книгу закрыть для записи, False – в противном случае |

## Методы объекта Workbook

|  |  |
| --- | --- |
| Activate | Активизирует рабочую книгу |
| Add | Создает новую книгу |
| Protect | Защищает рабочую книгу от внесения в неё изменений.  Protect(Password, Structure, Windows)  Например: ActiveWorkbook.Protect Password:= “Не влезай!” |
| UnProtect | Снятие защиты с листа  UnProtect(Password) |
| Close | Закрытие рабочей книги |
| Open | Открытие существующей книги |
| Save | Сохранение рабочей книги |
| SaveAs | Сохранение книги с другими параметрами |
| PrintPreview | Предварительный просмотр |

## События объекта Workbook

|  |  |
| --- | --- |
| BeforeClose | При закрытии книги |
| BeforePrint | Перед печатью рабочей книги |
| BeforeSave | Перед сохранением рабочей книги |
| NewSheet | При добавлении нового листа. |
| Open | При открытии рабочей книги. |
| SheetActivate | При активизации любого рабочего листа. |
| SheetDeactivate | Когда рабочий лист теряет фокус. |

## Объект Worksheet (лист)

## Свойства объекта Worksheet

|  |  |
| --- | --- |
| Name | Возвращает имя рабочего листа. |
| Visible | Отображает «видимость» или «невидимость» листа (True, False – соответственно) |
| ActiveCell | Возвращает активную ячейку активного листа. |
| StandartHeight | Возвращает стандартную высоту всех строк рабочего листа |
| UsedRange | Возвращает диапазон (объект Range), который содержит данные. |
| Union | Возвращает диапазон, являющийся объединением нескольких диапазонов. |

## Методы Worksheet

|  |  |
| --- | --- |
| Activate | Активизирует указанный рабочий лист. Worksheets(1).Activate |
| Add | Создает новый рабочий лист  Add(Before, After, Count, Type) |
| Delete | Удаляет рабочий лист |
| Protect | Защищает рабочий лист от внесения в него изменений |
| Unprotect | Снятие защиты с рабочего листа |
| Copy | Копирование рабочего листа в другое место  Copy (Before, After). Допустимо использование только одного аргумента |
| Move | Перемещение рабочего листа в другое место |
| Evaluate | Преобразует выражение в объект или значение. Используется при вводе формул и ячеек из диалоговых окон |

## Объекты Range (диапазон)

При работе с объектом Range необходимо помнить, как в Excel ссылаются на ячейку рабочего листа (см. относительные и абсолютные ссылки).

Так как ячейка является частным случаем диапазона, состоящим только из единственной ячейки, объект Range также позволяет работать с ней. Объект Cells(ячейки) – это альтернативный способ работы с ячейкой. Например, ячейка A2 как объект описывается Range(“A2”) или Cells(1,2).

## Свойства объекта Range

|  |  |
| --- | --- |
| Value | Возвращает значение из ячейки или в ячейки диапазона. Например: h=Range (“C1”).Value |
| Name | Возвращает имя диапазона. |
| WrapText | Позволяет переносить текст при вводе в диапазон.  With Range (“F3”)  .Value= “Привет всем!”  .WrapText=True  End With |
| Comment | Добавление комментария к данным в ячейки. |
| Font | Возвращает объект Шрифт с указанными параметрами. |
| Formula | Возвращает формулу в формате А1. |
| Text | Возвращает содержание диапазона в текстовом формате. |

## Наиболее часто используемые методы Range

|  |  |
| --- | --- |
| Clear | Очистка диапазона.  Range (“A1:B1).Clear |
| Copy | Копирует диапазон в другой или буфер обмена. |
| Cut | Копирует диапазон в другой или буфер обмена с удалением. |
| Delete | Удаляет диапазон. |
| Insert | Вставка ячейки или диапазона ячеек.  WorkSheets (“Лист1”).Rows(4).Insert |
| Select | Выделение диапазона. |

Заполнение произвольного диапазона данных по столбцам можно осуществить следующим образом для перебора адреса ячейки с А3 по А5:

For i = 1 To 3

A = "A" & i + 2

Range([A]).Value = InputBox("Введи данные " & i)

Next

## Задания на лабораторную работу

**Задание 1.** Создать приложение в VBA, позволяющее определить была ли сохранена открытая рабочая книга.

**Задание 2.**

Создать приложение в VBA, позволяющее произвести расчет амортизации различными методами. Данные для расчета брать с одного листа текущей книги, а выводить таблицу начисления амортизации в отдельную сохраненную книгу.

I. Применение функции АПЛ(нач\_стоимость;ост\_стоимость;время\_эксплуатации)

Стоимость актива составляет 10 000 руб., срок эксплуатации актива – 5 лет, а ликвидационная стоимость – 2000 руб. Создайте таблицу начисления износа, в которой бы указывались суммы ежегодной амортизации и значения остаточных стоимостей в конце каждого года, применяя линейный метод.

II. Применение функции АСЧ(нач\_стоимость;ост\_стоимость;время\_эксплуатации; период)

Стоимость актива составляет 46 000 руб., срок эксплуатации актива – 7 лет, а ликвидационная стоимость – 9000 руб. Создайте таблицу начислений износа, в которой бы указывались суммы ежегодной амортизации и значения остаточных стоимостей в конце каждого года, применяя метод «суммы (годовых) чисел».