

ПЕРВОЕ ВЫСШЕЕ ТЕХНИЧЕСКОЕ УЧЕБНОЕ ЗАВЕДЕНИЕ РОССИИ



**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**
федеральное государственное бюджетное образовательное учреждение
высшего образования
САНКТ-ПЕТЕРБУРГСКИЙ ГОРНЫЙ УНИВЕРСИТЕТ

СОГЛАСОВАНО

УТВЕРЖДАЮ

Проректор по международной
деятельности
ФГБОУ ВО «Санкт-Петербургский
горный университет»
_____ Борзенков В.Т.
« » _____ 2022 г.

Проректор по образовательной
деятельности
ФГБОУ ВО «Санкт-Петербургский
горный университет»
_____ Петраков Д.Г.
« » _____ 2022 г.

**МЕЖДУНАРОДНАЯ КРАТКОСРОЧНАЯ ОБРАЗОВАТЕЛЬНАЯ ПРОГРАММА ПО
ОСВОЕНИЮ ОБУЧАЮЩИМИСЯ ДОПОЛНИТЕЛЬНЫХ
ПРОФЕССИОНАЛЬНЫХ КОМПЕТЕНЦИЙ**

**«ЭЛЕКТРОННЫЕ ТАБЛИЦЫ MS EXCEL:
ПРОДВИНУТОЕ ВЛАДЕНИЕ»**

Уровень программы: профессиональный

Форма обучения: очная (с применением дистанционных образовательных технологий – ДОТ)

Модель использования ДОТ: полностью дистанционное обучение

Объем программы: 50 часов

Руководитель программы: _____ к.т.н. Грищенко Е.Н.

Составитель программы: _____ к.т.н. Грищенко Е.Н.

**САНКТ-ПЕТЕРБУРГ
2022**

1. Общие положения

1.1. Цель программы:

Цель программы – приобретение знаний и формирование практических навыков работы с программой Microsoft Excel.

1.2. Основные задачи программы:

- получение дополнительных знаний и практических навыков в области информационных технологий для обработки и статистического анализа больших объемов данных;
- получение дополнительных знаний и практических навыков в области автоматизации решения вычислительных задач разного рода;
- получение дополнительных знаний и практических навыков в области программирования и разработки пользовательских инструментов для расчетов, визуализации и аналитики.

1.3. Категория слушателей:

Лица, получающие высшее образование (студенты, магистранты, аспиранты) в высших технических образовательных организациях минерально-сырьевого комплекса по различным направлениям подготовки, за исключением направления 09.03.02 «Информационные системы и технологии», с уровнем владения английским языком – не ниже В2.

1.4. Планируемые результаты обучения

Перечень дополнительных профессиональных компетенций, качественное изменение которых осуществляется в результате реализации программы обучения:

- Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности;
- Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий.

1.5. Требования к результатам освоения программы:

С целью достижения указанных в пункте 1.4 дополнительных профессиональных компетенций, слушатели в процессе освоения краткосрочной программы должны:

Знать:

- современные инструментальные среды, программно-технические платформы и программные средства, в том числе отечественного производства, используемые для решения задач профессиональной деятельности, и принципы их работы;
- процессы, методы поиска, сбора, хранения, обработки, предоставления, распространения информации и способы осуществления таких процессов и методов (информационные технологии);
- логику построения и принципы функционирования современных языков программирования и языков работы с базами данных, сред разработки информационных систем и технологий, принципы разработки алгоритмов и компьютерных программ;
- современные языки программирования и языки работы с базами данных, среды разработки информационных систем и технологий.

Уметь:

- выбирать и использовать современные информационно-коммуникационные и интеллектуальные технологии, инструментальные среды, программно-технические платформы и программные средства, в том числе отечественного производства, для решения задач профессиональной деятельности;

– анализировать профессиональные задачи, выбирать и использовать подходящие ИТ-решения;

– применять современные языки программирования для разработки оригинальных алгоритмов и компьютерных программ, пригодных для практического применения, вести базы данных и информационные хранилища, применять современные программные среды разработки информационных систем и технологий;

– читать коды программных продуктов, написанных на освоенных языках программирования, и вносить требуемые изменения

Владеть:

– навыками работы с лежащими в основе ИТ-решений данными;

– навыками применения современных информационно-коммуникационных и интеллектуальных технологий, инструментальных сред, программно-технических платформ и программных средств, в том числе отечественного производства, для решения задач профессиональной деятельности;

– навыками разработки оригинальных алгоритмов и компьютерных программ, пригодных для практического применения

– навыками отладки и тестирования прототипов программно-технических комплексов задач.

1.6. Календарный учебный график проведения дистанционных занятий

Условные обозначения:

Теоретическое обучение	час
Итоговая аттестация	ИА

Форма обучения	Дни недели/ауд.час							
	1	2	3	4	5	6	7	8
Очная (с применением дистанционных образовательных технологий)	4	4	4	4	4	4	4	4, ИА

1.7. Учебный план:

№	Наименование модуля	Всего часов	В том числе					
			к	ц	ск	е	ст.	на
1	Введение. Обзор программного обеспечения Microsoft Excel	2	2	-	-	-	-	-
2	Модуль 1. Вычисления в Microsoft Excel. Таблицы. Диаграммы. Функции.	16	-	10	6	-	-	-
3	Модуль 2. Анализ данных. Сводные таблицы и диаграммы.	12	-	6	6	-	-	-
4	Модуль 3. Макросы. Разработка инструментов вычислений в VBA	16	-	10	6	-	-	-
5	Итоговая аттестация	4	-	-	-	-	-	4
	Всего	50	2	26	18	4	4	4

1.8. Объем программы и виды учебной работы:

Вид учебной работы	Часы
Лекционные занятия	2
Практические занятия	26
Итоговая аттестация	4
Всего учебных занятий	32
Самостоятельная работа, включая подготовку к итоговой аттестации	18
Общий объем программы	50

2. Содержание обучения:

2.1. Содержание обучения по программе:

Наименование разделов профессионального модуля, тем	Содержание учебного материала	Объем часов
Введение. Обзор программного обеспечения Microsoft Excel	Общие сведения о Microsoft Excel. Эффективные приёмы работы с книгами Excel.	2
Модуль 1. Вычисления в Microsoft Excel. Таблицы. Диаграммы. Функции.	<ul style="list-style-type: none">– Создание таблиц данных, форматирование таблиц, сортировка, фильтры, поиск и замена данных;– Формулы и функции, математические, логические статические, текстовые и другие функции;– Графическое представление данных, создание диаграмм, спарклайнов.	16
Модуль 2. Анализ данных. Сводные таблицы и диаграммы.	<ul style="list-style-type: none">– Выделение тренда, подбор параметров, поиск решения;– Анализ данных (корреляция, регрессия), анализ «что – если»;– Сводные таблицы, настройка полей, группировка, преобразование;– Сводные диаграммы.	12
Модуль 3. Макросы. Разработка инструментов вычислений в VBA	<ul style="list-style-type: none">– Работа с макрорекордером (быстрое создание макросов);– Основы программирования в VBA;– Программирование линейных вычислительных процессов;– Программирование разветвляющегося вычислительного процесса;– Программирование циклов;– Создание и дизайн рабочих окон (форм).	16

2.2. Рабочие программы модулей – представлены в Приложении 1 к образовательной программе.

2.3. Формы аттестаций по программе:

Для оценки качества усвоения знаний и умений предусмотрены текущий и итоговый виды контроля.

Текущий контроль успеваемости осуществляется на основе тестов, которые содержат контрольные вопросы по каждому изучаемому модулю и должны быть сданы обучающимися в ходе программы.

Тесты составляются преподавателем с помощью Google Forms; ссылка на тест высылается слушателям программы по электронной почте.

Форма итоговой аттестации по программе – зачет (часть 1 – демонстрация навыков работы в Microsoft Excel по индивидуальным вопросам преподавателя, часть 2 – сдача итогового теста).

Часть 1 зачета проводится на платформе Cisco Webex; ссылка для подключения к видеосвязи высылается слушателям программы по электронной почте.

Часть 2 зачета проводится в системе Google Forms; ссылка на тест высылается слушателям программы по электронной почте.

К зачету допускаются только те слушатели, которые успешно сдали все домашние задания.

На аттестации предусмотрена система прокторинга, позволяющая подтвердить личность слушателя и объективно оценить его знания. Этому способствуют включенная веб-камера и микрофон на протяжении всей аттестации. При обрыве связи слушателю будет назначено новое время для сдачи итоговой аттестации.

2.4. Оценочные материалы:

Примеры тестовых вопросов для зачета:

1. Какую формулу нужно использовать для получения псевдослучайного вещественного числа в диапазоне от 0 до 1?
 - 1) СЛУЧМЕЖДУ;
 - 2) СЛМЕЖДУ;
 - 3) СЛЧИС;
 - 4) СЛУЧЧИС
2. Что делает функция СРЗНАЧЕСЛИ?
 - 1) Возвращает среднее среди всех ячеек, которые соответствуют условию
 - 2) Возвращает среднее, если оно соответствует условию
 - 3) Возвращает значения ячеек, которые соответствуют условию
3. В каком из равенств присутствует ошибка?
 - 1) R2C5 = \$E\$2
 - 2) R[1]C6 = \$F1
 - 3) R[2]C[2] = B3
4. В чем отличие между "Поиском решения" и "Подбором параметра"?
 - 1) Смысл "Подбора параметра" - в определении минимума функции
 - 2) В "Поиске решения" можно указывать ограничивающие условия
 - 3) В "Подборе параметра" можно выбрать метод решения
5. Какая ошибка возникнет при вводе следующего кода: `If True Then MsgBox ("Hello")`
 - 1) Ошибка компиляции
 - 2) Ошибка выполнения из-за бесконечного цикла
 - 3) Логическая ошибка
 - 4) Нет ошибки
6. Что произойдет, если прописать строку `UserForm1.Hide` в событии `MouseDown` формы `UserForm1`?
 - 1) Форма закроется, как только пользователь отпустит кнопку мыши
 - 2) Форма закроется, как только пользователь зажмет кнопку мыши
 - 3) Форма закроется, как только пользователь кликнет по форме

Примерная шкала оценивания знаний в рамках зачета в тестовой форме

Количество правильных ответов, %	Оценка
0-59	Не зачтено
60-100	Зачтено

2.5. Учебно-методические материалы (в том числе конспекты лекций) – представлены в Приложении 2 к образовательной программе.

2.6. Вид документа, подтверждающий прохождение обучения:

После успешного окончания обучения выдается сертификат федерального государственного бюджетного образовательного учреждения высшего образования «Санкт-Петербургский горный университет» о прохождении международной краткосрочной образовательной программы «Электронные таблицы MS Excel: продвинутое владение».

В случае невыполнения требований по посещаемости и/или итоговой аттестации слушателю выдается справка об обучении.

3. Организационно-педагогические условия реализации программы:

3.1. Материально-технические условия реализации программы:

Рабочее место преподавателя оборудовано персональным компьютером (ноутбуком) с веб-камерой, микрофоном, доступом к сети Интернет. На компьютере установлено программное обеспечение Cisco Webex, MS Excel.

Удаленное рабочее место слушателя снабжено персональным компьютером (ноутбуком) с веб-камерой, микрофоном, доступом к сети Интернет. На компьютере установлено программное обеспечение Cisco Webex, MS Excel.

3.2. Кадровое обеспечение образовательного процесса по программе:

Фамилия, Имя, Отчество	Образование (вуз; год окончания; специальность)	Должность, ученая степень, звание, стаж работы в данной или аналогичной области, лет	Количество научных и учебно- методических публикаций
Руководитель программы			
Грищенко Екатерина Николаевна	Донецкий национальный технический университет (2015), специальность – инженерная геодезия; профессиональная переподготовка по программе «Преподаватель информатики» (542 часа, 2020).	Международный центр компетенций в горнотехническом образовании под эгидой ЮНЕСКО, начальник отдела, кандидат технических наук, 3 года	17 научных статей, 3 свидетельства о регистрации программ для ЭВМ.
Профессорско-преподавательский состав			
Грищенко Екатерина Николаевна	Донецкий национальный технический университет (2015), специальность – инженерная геодезия; профессиональная переподготовка по программе «Преподаватель информатики» (2020).	Международный центр компетенций в горнотехническом образовании под эгидой ЮНЕСКО, начальник отдела, кандидат технических наук, 3 года	17 научных статей, 3 свидетельства о регистрации программ для ЭВМ.

ПРИЛОЖЕНИЕ 1
к образовательной программе по освоению
обучающимися дополнительных профессиональных компетенций
«Электронные таблицы MS Excel:
продвинутое владение»

Рабочая программа модуля

«Введение. Обзор программного обеспечения Microsoft Excel»

1. Структура модуля

Наименование модуля / наименование тем модуля	Всего, час	В том числе			Форма контроля
		Лекционные занятия	Практические занятия	Самостоятельная работа	
Введение. Обзор программного обеспечения Microsoft Excel	2	2	–	–	–
Установочная лекция «Microsoft Excel. Эффективные приёмы работы с книгами Excel»	2	2	–	–	–

2. Матрица формирования профессиональных компетенций

Наименование тем модуля	Кол-во часов	Профессиональные компетенции
Введение. Обзор программного обеспечения Microsoft Excel	2	Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности

3. Содержание модуля

Введение. Обзор программного обеспечения Microsoft Excel (2 час.)

Модуль включает 2 часа лекции руководителя программы.

Содержание лекции.

Введение. Актуальность программного обеспечения MS Excel в современном мире. Изучение Excel в рамках курса информатики в минерально-сырьевых университетах.

История Excel. Первые электронные таблицы VisiCalc. Электронные таблицы компании Lotus. Первые версии Excel и набор доступных функций. Постепенное совершенствование программного обеспечения Excel, внедрение новых возможностей. Включение VBA в состав Excel.

Excel в современном мире. Возможности для профессионалов и новичков. Наглядное представление данных. Сферы применения Excel.

Содержание предстоящей программы. Порядок работы на занятиях. Порядок сдачи домашних заданий. Итоговая аттестация и сдача курсового проекта.

5. Учебно-методическое обеспечение модуля

1. Справка и обучение по Excel [Электронный ресурс]: служба поддержки Microsoft. URL: <https://support.microsoft.com/ru-ru/excel> (дата обращения: 16.09.2021).

2. Alexander M., Kusleika R. Excel 2019 Bible. – John Wiley & Sons, Inc., 2019. – pp. 1074.

3. DeMarco, J. Pro Excel 2007 VBA. – Springer (March 16, 2009). – pp. 388.
 4. Чиртик А.А., Гладкий А.А. Excel. Трюки и эффекты. – Питер, 2006. – 368 с.
 5. Комолова Н.В., Клименко А.В. Программирование на VBA в Excel 2019. Самоучитель. – BHV, 2020. – 496 с.

Рабочая программа модуля

«Вычисления в Microsoft Excel. Таблицы. Диаграммы. Функции»

1. Структура модуля

Наименование модуля/наименование тем модуля	Всего, час	В том числе			Форма контроля
		Лекционные занятия	Практические занятия	Самостоятельная работа	
Модуль 1. Вычисления в Microsoft Excel. Таблицы. Диаграммы. Функции.	16	–	10	6	текущий
Базовые операции в Excel	2	–	2	–	–
Математические и статистические функции Excel	2	–	2	–	–
Логические и текстовые функции Excel	2	–	2	–	–
Применение математических, статистических, логических и текстовых функций	2	–	–	2	–
Работа с одномерными и двумерными массивами. Ссылки	2	–	2	–	–
Применение средств работы с массивами и ссылками	2	–	–	2	–
Диаграммы и спарклайны в Excel	2	–	2	–	–
Графическое представление данных	2	–	–	2	–

2. Матрица формирования профессиональных компетенций

Наименование тем модуля	Кол-во часов	Профессиональные компетенции
Вычисления в Microsoft Excel. Таблицы. Диаграммы. Функции.	16	Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности.

3. Содержание модуля

Модуль 1. Вычисления в Microsoft Excel. Таблицы. Диаграммы. Функции (16 час.)

Модуль включает 10 часов практических занятий и 6 часов самостоятельной работы.

В данный раздел включены:

- практические занятия «Базовые операции в Excel», «Математические и статистические функции Excel», «Логические и текстовые функции Excel», «Работа с одномерными и двумерными массивами. Ссылки», «Диаграммы и спарклайны в Excel»;
- самостоятельные занятия «Применение математических, статистических, логических и текстовых функций», «Применение средств работы с массивами и ссылками», «Графическое представление данных».

Содержание лекций, практических занятий, и самостоятельных занятий.

Основные характеристики Excel. Интерфейс, параметры, вкладки, функции, свойства, основы работы.

Ячейки, листы, книги, рабочие области. Адрес ячейки. Выделение ячеек, столбцов, строк, областей. Ввод, редактирование и удаление данных. Специальный адрес ячейки.

Форматирование ячеек. Числовой формат, выравнивание, шрифт, границы, заливка, защита. Системный разделитель целой и дробной части числа. Рассмотрение отдельных типов данных. Объединение ячеек, перенос текста. Индексы

Вставка строк и столбцов. Копирование ячеек.

Создание формул. Множественные расчеты по формулам. Инструмент "растаскивания". Автосуммирование и прочие автоматические расчеты. Сортировка, фильтры.

Операции с рабочими листами, копирование, перемещение.

Сохранение документов. Печать документов.

Математические функции: абсолютное значение, число пи, перевод в градусы и радианы, тригонометрические функции, экспонента, логарифмы, корень числа, перевод в арабские и римские числа, определение целого, функции округления, определение факториала, наибольшего общего делителя и кратного, случайных значений, суммы чисел, произведения.

Статистические функции: среднее значение, подсчет значений, в том числе, с условием, определение максимального и минимального значения, медианы и моды.

Логические функции: вычисление с указанием условия, логические операции И, ИЛИ и НЕ.

Текстовые функции: определение числового кода символов, сцепление строк, сравнение строк, поиск текста в строке, отсечение символов с начала или конца строки, определение числа символов в строке, перевод букв в прописные или строчные, замена текста в строке, повтор текста.

Работа со ссылками и массивами значений. Использование функций ВПР и ГПР для поиска значений в таблицах. Определение номера строки или столбца, адреса ячейки, указание ссылок на ячейки, подсчет числа строк и столбцов, поиск значения в диапазоне ячеек.

Работа с матрицами: создание двумерных массивов, сложение матриц, вычитание матриц, умножение матриц, транспонирование матриц, нахождение обратной матрицы, нахождение определителя матрицы.

Графическое представление данных. Создание диаграмм, упорядочение данных для диаграмм. Всевозможные виды диаграмм. Гистограмма, линейчатая диаграмма, график, график с областями, поверхностная и лепестковая диаграммы, круговая диаграмма, кольцевая диаграмма, точечная и пузырьковая диаграммы, биржевая диаграмма.

Добавление данных в диаграмму. Изменение ряда данных. Фильтрация данных на диаграмме. Добавление линии тренда.

Спарклайны. Указание диапазона данных для спарклайна и расположения. Создание спарклайна. Отметка точек данных на спарклайнах. Изменение стиля. Обработка пустых ячеек и нулевых значений. Удаление спарклайнов.

4. Перечень занятий семинарского типа

№ темы	Наименование занятия семинарского типа	Вид занятия	Кол-во час.
1	Базовые операции в Excel	практическое занятие	2
2	Математические и статистические функции Excel	практическое занятие	2
3	Логические и текстовые функции Excel	практическое занятие	2
4	Работа с одномерными и двумерными массивами. Ссылки	практическое занятие	2
5	Диаграммы и спарклайны в Excel	практическое занятие	2

5. Учебно-методическое обеспечение модуля

1. Справка и обучение по Excel [Электронный ресурс]: служба поддержки Microsoft. URL: <https://support.microsoft.com/ru-ru/excel> (дата обращения: 16.09.2021).
2. Alexander M., Kusleika R. Excel 2019 Bible. – John Wiley & Sons, Inc., 2019. – pp. 1074.
3. DeMarco, J. Pro Excel 2007 VBA. – Springer (March 16, 2009). – pp. 388.
4. Чиртик А.А., Гладкий А.А. Excel. Трюки и эффекты. – Питер, 2006. – 368 с.

Рабочая программа модуля

«Анализ данных. Сводные таблицы и диаграммы»

1. Структура модуля

Наименование модуля/наименование тем модуля	Всего, час	в том числе			Форма контроля
		Лекционные занятия	Практические занятия	Самостоятельная работа	
Модуль 2. Анализ данных. Сводные таблицы и диаграммы.	12	–	6	6	текущий
Решение нелинейных уравнений в Excel	2	–	2	–	–
Практическое применение инструментов подбора параметра и поиска решения	2	–	–	2	–
Анализ данных в Excel	2	–	2	–	–
Практическое применение инструментов анализа данных	2	–	–	2	–
Сводные таблицы и диаграммы в Excel	2	–	2	–	–
Практическое применение инструментов сводных таблиц и диаграмм	2	–	–	2	–

2. Матрица формирования профессиональных компетенций

Наименование тем модуля	Кол-во часов	Профессиональные компетенции
Анализ данных. Сводные таблицы и диаграммы.	12	Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности

3. Содержание модуля

Модуль 2. Анализ данных. Сводные таблицы и диаграммы. (12 час.)

Модуль включает 6 часов практических занятий и 6 часов самостоятельной работы. В данный раздел включены:

- практические занятия «Решение нелинейных уравнений в Excel», «Анализ данных в Excel», «Сводные таблицы и диаграммы в Excel»;
- самостоятельные занятия «Практическое применение инструментов подбора параметра и поиска решения», «Практическое применение инструментов анализа данных», «Практическое применение инструментов сводных таблиц и диаграмм».

Содержание лекций, практических занятий и самостоятельных занятий.

Математические основы решения нелинейных уравнений. Применение инструментов "Подбора параметра" и "Поиска решения" для решения уравнений.

Процесс работы инструмента "Подбор параметра". Порядок решения задания и уточнения корня уравнения.

Процесс работы инструмента "Поиск решения". Порядок решения задания и уточнения корня уравнения. Целевая функция. Максимум и минимум целевой функции.

Инструментарий Excel для анализа данных. Инструмент разбиения текста по столбцам. Инструмент мгновенного заполнения ячеек. Определение закономерностей для мгновенного заполнения. Примеры мгновенного заполнения. Консолидация данных в Excel. Построение однотипных таблиц и применение инструмента консолидации. Использование диспетчера сценариев для однотипной работы. Построение таблиц данных. Прогноз данных (в Excel версии 2019 года). Указание временной шкалы и определение прогнозируемых значений. Структурирование данных. Группировка. Получение промежуточных итогов. Инструменты статистического анализа: корреляция, регрессия.

Сводные таблицы и диаграммы, и область их применения. Создание сводных таблиц. Настройка сводных таблиц. Изменение исходных данных сводной таблицы. Упорядочение полей сводной таблицы с помощью списка полей. Использование срезов для фильтрации данных. Создание временной шкалы сводной таблицы для фильтрации дат. Использование временной шкалы для фильтрации по периоду времени.

Сводные диаграммы и их виды. Создание сводной диаграммы. Создание диаграммы на основе сводной таблицы. Виды диаграмм. Изменение данных сводных таблиц. Удаление данных сводных таблиц.

4. Перечень занятий семинарского типа

№ темы	Наименование занятия семинарского типа	Вид занятия	Кол-во час.
1	Решение нелинейных уравнений в Excel	практическое занятие	2
2	Анализ данных в Excel	практическое занятие	2
3	Сводные таблицы и диаграммы в Excel	практическое занятие	2

5. Учебно-методическое обеспечение модуля

1. Справка и обучение по Excel [Электронный ресурс] : служба поддержки Microsoft. URL: <https://support.microsoft.com/ru-ru/excel> (дата обращения: 16.09.2021).

2. Alexander M., Kusleika R. Excel 2019 Bible. – John Wiley & Sons, Inc., 2019. – pp. 1074.

3. DeMarco, J. Pro Excel 2007 VBA. – Springer (March 16, 2009). – pp. 388.

4. Чиртик А.А., Гладкий А.А. Excel. Трюки и эффекты. – Питер, 2006. – 368 с.

Рабочая программа модуля

«Макросы. Разработка инструментов вычислений в VBA»

1. Структура модуля

Наименование модуля/наименование тем модуля	Всего, час	в том числе			Форма контроля
		Лекционные занятия	Практические занятия	Самостоятельная работа	
Модуль 3. Макросы. Разработка инструментов вычислений в VBA.	16	–	10	6	текущий
Макросы и введение в программирование	2	–	2	–	–
Применение инструмента записи макросов для решения задач	2	–	–	2	–

Наименование модуля/наименование тем модуля	Всего, час	в том числе			Форма контроля
		Лекционные занятия	Практические занятия	Самостоятельная работа	
Программирование линейных вычислительных процессов в VBA	2	–	2	–	–
Программирование разветвляющегося вычислительного процесса в VBA	2	–	2	–	–
Программирование циклов в VBA	2	–	2	–	–
Алгоритмы линейных вычислений, ветвления и циклов	2	–	–	2	–
Разработка форм (рабочих окон) в VBA	2	–	2	–	–
Дизайн интерфейса собственного приложения и программирование его функций	2	–	–	2	–

2. Матрица формирования профессиональных компетенций

Наименование тем модуля	Кол-во часов	Профессиональные компетенции
Макросы. Разработка инструментов вычислений в VBA.	16	Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий.

3. Содержание модуля

Модуль 3. Макросы. Разработка инструментов вычислений в VBA. (16 час.)

Модуль включает 10 часов практических занятий и 6 часов самостоятельной работы.

В данный раздел включены:

- практические занятия «Макросы и введение в программирование», «Программирование линейных вычислительных процессов в VBA», «Программирование разветвляющегося вычислительного процесса в VBA», «Программирование циклов в VBA», «Разработка форм (рабочих окон) в VBA»;
- самостоятельные занятия «Применение инструмента записи макросов для решения задач», «Алгоритмы линейных вычислений, ветвления и циклов», «Дизайн интерфейса собственного приложения и программирование его функций».

Содержание лекций, практических занятий и самостоятельных занятий.

Подключение вкладки "Разработчик". Инструменты записи макросов в Excel. Запись макроса. Основы и инструкции по записи макросов в Excel. Работа с макросами, записанными в Excel. Корректировка кода записанных макросов. Запуск макросов. Удаление макросов.

Введение в программирование. Автоматизация и повторяемость задач. Расширение возможностей взаимодействия с пользователем. Использование кода для выполнения операций приложениями. Обращение к ячейкам. Использование переменных. Объявление переменных. Уровни доступа: публичный, частный и другие. Типы данных: целочисленные, вещественные, строковые, даты, финансовые, логические. Использование массивов. Динамические массивы. Ключевые слова. Построение диаграмм с помощью кода.

Программирование линейных вычислительных процессов в VBA. Стандартные математические функции. Математические функции модуля Math. Математические функции модуля WorksheetFunction.

Программирование разветвляющегося вычислительного процесса в VBA.

Однострочная и блочная форма записи условных операторов.

Программирование циклов в VBA. Циклы с предусловием и с постусловием. Циклы со счетчиком. Цикл While Wend, цикл Do While Loop, цикл Do Loop While, цикл Do Until Loop. Цикл For. Ранний выход из цикла.

Формы (рабочие окна) в VBA. Добавление формы UserForm. Свойства форм: заголовок, высота, ширина, отступ, фон, цвет границы, шрифт, состояние активности, картинка, полоса прокрутки. События при взаимодействии с формой: при инициализации, активации и деактивации, нажатии и двойном нажатии кнопки мыши, нажатии клавиш клавиатуры, прокрутки с помощью колеса мыши, изменении размера окна.

Элементы панели инструментов: метка с текстом, поле для ввода информации, выпадающий список, список с перечнем позиций, поле для отметки, переключатель, кнопка с фиксированным состоянием, граница для расположения элементов, обычная кнопка, диспетчер с коллекцией вкладок, диспетчер с коллекцией страниц, полоса прокрутки, двунаправленный счетчик, объект изображения, редактор ссылок. Окно свойств. Общие свойства. Отображение и закрытие окна. Пример разработки собственного приложения.

4. Перечень занятий семинарского типа

№ темы	Наименование занятия семинарского типа	Вид занятия	Кол-во час.
1	Макросы и введение в программирование	практическое занятие	2
2	Программирование линейных вычислительных процессов в VBA	практическое занятие	2
3	Программирование разветвляющегося вычислительного процесса в VBA	практическое занятие	2
4	Программирование циклов в VBA	практическое занятие	2
5	Разработка форм (рабочих окон) в VBA	практическое занятие	2

5. Учебно-методическое обеспечение модуля

1. Справка и обучение по Excel [Электронный ресурс]: служба поддержки Microsoft. URL: <https://support.microsoft.com/ru-ru/excel> (дата обращения: 16.09.2021).

2. Alexander M., Kusleika R. Excel 2019 Bible. – John Wiley & Sons, Inc., 2019. – pp. 1074.

3. DeMarco, J. Pro Excel 2007 VBA. – Springer (March 16, 2009). – pp. 388.

4. Чиртик А.А., Гладкий А.А. Excel. Трюки и эффекты. – Питер, 2006. – 368 с.

5. Комолова Н.В., Клименко А.В. Программирование на VBA в Excel 2019. Самоучитель. – BHV, 2020. – 496 с.

ПРИЛОЖЕНИЕ 2

**к образовательной программе по освоению
обучающимися дополнительных профессиональных компетенций
«Электронные таблицы MS Excel: продвинутое владение»**

**Методические указания для обучающихся по освоению программы
с применением ДОТ**

Процесс изучения материала программы предусматривает активное использование современных инновационных образовательных технологий. Формы обучения: индивидуальные и групповые. Методы обучения:

- работа с преподавателем;
- самостоятельная работа.

При освоении курса используются следующие виды форм обучения для достижения запланированных результатов обучения и формирования компетенций:

- совместное погружение в проблемное поле;
- обсуждение сложных вопросов и проблем;
- разборы конкретных ситуаций и т.д.

Процесс освоения курса предусматривает следующие работы:

1. Лекционные, практические занятия;
2. Самостоятельная работа;
3. Контрольные мероприятия (итоговая аттестация).

**Методические указания для обучающихся по лекционным занятиям
по модулю с применением ДОТ**

Лекционные занятия проводятся посредством видеоконференцсвязи. Преподаватель заранее обеспечивает слушателей ссылкой на подключение. Занятие проходит при включенным веб-камерах, динамиках и микрофонах компьютеров преподавателя и слушателей.

Взаимодействие со слушателями осуществляется посредством видеосвязи или текстового чата в системе.

Лекция является наиболее экономичным способом передачи учебной информации, т.к. при этом обширный материал излагается концентрировано, в логически выдержанной форме, с учетом характера профессиональной деятельности обучаемых. Лекция закладывает основы научных знаний в обобщенной форме. На лекционных занятиях преподаватель:

- знакомит обучающихся с общей методикой работы над курсом;
- дает характеристику учебников и учебных пособий, знакомит слушателей со списком литературы;
- рассказывает о требованиях к аттестации;
- рассматривает основные теоретические положения курса;
- разъясняет вопросы, которые возникли у обучающихся в процессе изучения курса.

Лекционное занятие преследует 5 основных дидактических целей:

- информационную (сообщение новых знаний);
- развивающую (систематизация и обобщение накопленных знаний);
- воспитывающую (формирование взглядов, убеждений, мировоззрения);
- стимулирующую (развитие познавательных и профессиональных интересов);
- координирующую с другими видами занятий.

Методические указания для обучающихся по практическим занятиям по модулю с применением ДОТ

Практические занятия проводятся посредством видеоконференцсвязи. Преподаватель заранее обеспечивает слушателей ссылкой на подключение. Занятие проходит при включенным веб-камерах, динамиках и микрофонах компьютеров преподавателя и слушателей.

Для того чтобы практические занятия приносили максимальную пользу, необходимо помнить, что упражнение и решение заданий проводятся по вычитанному на лекциях материалу и связаны, как правило, с детальным разбором отдельных вопросов лекционного курса. Следует подчеркнуть, что только после усвоения лекционного материала с определенной точки зрения (а именно с той, с которой он излагается на лекциях) он будет закрепляться на практических занятиях как в результате обсуждения и анализа лекционного материала, так и с помощью решения проблемных ситуаций, задач. При этих условиях обучающийся не только хорошо усвоит материал, но и научится применять его на практике, а также получит дополнительный стимул (и это очень важно) для активной проработки лекции.

При самостоятельном решении заданий нужно обосновывать каждый этап решения, исходя из теоретических положений курса. Если обучающийся видит несколько путей решения проблемы, то нужно сравнить их и выбрать самый рациональный. Полезно до начала вычислений составить краткий план решения проблемы. Решение проблемных заданий или примеров следует излагать подробно, вычисления располагать в строгом порядке, отделяя вспомогательные вычисления от основных. Решения при необходимости нужно сопровождать комментариями, схемами, чертежами и рисунками.

Следует помнить, что решение каждого учебного задания должно доводиться до окончательного логического ответа, которого требует условие, и по возможности с выводом. Полученный ответ следует проверить способами, вытекающими из существа данного задания. Полезно также (если возможно) решать несколькими способами и сравнить полученные результаты. Решение заданий данного типа нужно продолжать до приобретения твердых навыков в их решении.

При подготовке к практическим занятиям следует использовать литературу из представленного списка, а также руководствоваться приведенными указаниями и рекомендациями. На практических занятиях приветствуется активное участие в обсуждении конкретных ситуаций, способность на основе полученных знаний находить наиболее эффективные решения поставленных проблем, уметь находить полезный дополнительный материал по тематике занятий.

Домашние задания сдают преподавателю в срок и в полном объеме, высылая по электронной почте.

В процессе подготовки изучают рекомендованные преподавателем источники литературы, а также самостоятельно осуществляют поиск релевантной информации.

Методические указания для обучающихся по самостоятельной работе по модулю с применением ДОТ

Достижение целей эффективной подготовки обучающихся и развитие профессиональных компетенций невозможно без их целеустремленной самостоятельной работы. Самостоятельная работа обучающихся является составной частью учебной работы и имеет целью закрепление и углубление полученных знаний и навыков, поиск и приобретение новых знаний, в том числе с использованием автоматизированных обучающих систем, а также выполнение учебных заданий, подготовку к предстоящим занятиям, текущему контролю и итоговой аттестации.

Основная цель данного вида занятий состоит в обучении методам самостоятельной работы с учебным материалом, научной литературой, с ситуационными задачами, развитие способности самостоятельно повышать уровень профессиональных знаний,

реализуя специальные средства и методы получения нового знания, и использовать приобретенные знания и умения в практической деятельности. Состав самостоятельной работы:

1. Подготовка к лекционным и практическим занятиям:

- чтение текста (учебника, первоисточника, литературы и т.д.);
- подготовка вопросов для самостоятельного изучения.

2. Подготовка к итоговой аттестации:

- повторение всего учебного материала модуля;
- аналитическая обработка текста; периодического, продолжающегося издания или сборника как составная часть его основного текста.

Методические указания для обучающихся по итоговой аттестации по модулю с применением ДОТ

В период подготовки к итоговой аттестации обучающихся вновь обращаются к пройденному учебному материалу. При этом они не только закрепляют полученные знания, но и получают новые. Подготовка обучающегося к аттестации включает в себя три этапа:

- самостоятельная работа в течение программы;
- непосредственная подготовка в дни, предшествующие итоговой аттестации по темам программы;
- подготовка к ответам на вопросы.

Подготовка к аттестации осуществляется на основании конспектов лекций, учебников и учебных пособий, информации среды интернет. Литература для подготовки к аттестации рекомендуется преподавателем. Для полноты учебной информации и ее сравнения лучше использовать не менее двух источников. Обучающийся вправе сам придерживаться любой из представленных в литературе точек зрения по спорной проблеме (в том числе отличной от преподавателя), но при условии достаточной научной аргументации.

Аттестация проводится в форме зачета по темам, охватывающим весь пройденный материал. Зачет включает две части: часть 1 – демонстрация навыков работы в Excel по индивидуальным вопросам преподавателя; часть 2 – сдача итогового теста. Предметом оценивания являются знания, умения и практический опыт обучающихся.

Учебно-методические материалы (в том числе конспекты лекций)

Модуль «Вычисления в Microsoft Excel. Таблицы. Диаграммы. Математические и логические функции»

Конспект лекции №1

«Microsoft Excel. Эффективные приёмы работы с книгами Excel»

Microsoft Excel – одна из самых популярных программ для работы с электронными таблицами, а также функциональный инструмент визуализации и анализа данных.

Первая электронная таблица, явившаяся прототипом программы Excel, появилась в 1979 году благодаря студенту из Гарварда Даниэлю Бриклину. Как известно, студентам приходится делать множество математических расчетов, и Бриклин счел это занятие слишком утомительным и трудозатратным. Придумав первую в мире электронную таблицу, Бриклин, тем не менее, доказал, что студенты Гарварда – действительно лучшие из лучших.

Электронная таблица получила название VisiCalc и сразу обрела популярность. Ее первая версия предназначалась для компьютеров Apple. Да – да, оказывается, они уже

существовали в то время и, кстати, именно появлению VisiCalc Apple обязан значительным скачком продаж.

Следующим этапом развития Excel стала электронная таблица, представленная компанией Lotus и предназначенная для компьютеров IBM. И снова, благодаря электронной таблице, оборот компании вскоре достиг 50 миллионов долларов в год.

Версия Excel, которая в большей степени уже знакома пользователям, была представлена в 1987 году компанией Майкрософт. Программа имела усовершенствованный интерфейс и широкий набор функций.

В сравнении с первыми табличными процессорами Excel представляет множество новых функций пользовательского интерфейса, но суть остается прежней: как и в программе-родоначальнике, VisiCalc, организованные в строки и столбцы клетки-ячейки могут содержать данные или формулы с относительными или абсолютными ссылками на другие клетки.

Excel был первым табличным процессором, позволявшим пользователю менять внешний вид таблицы на экране: шрифты, символы и внешний вид ячеек. Он также первым представил метод умного пересчёта ячеек – обновления только ячеек, зависящих от изменённых ячеек: раньше табличные процессоры пересчитывали все ячейки; это делалось либо после каждого изменения (что на больших таблицах долго), либо по команде пользователя (что могло вводить пользователя в заблуждение не пересчитанными значениями).

Будучи впервые объединёнными в Microsoft Office в 1993 году, Microsoft Word и Microsoft PowerPoint получили новый графический интерфейс для соответствия Excel, главного стимула модернизации ПК в то время.

Начиная с 1993 года, в состав Excel входит Visual Basic для приложений (VBA), язык программирования, основанный на Visual Basic, позволяющий автоматизировать задачи Excel. VBA является мощным дополнением к приложению и в более поздних версиях Excel доступна полнофункциональная интегрированная среда разработки. Можно создать VBA-код, повторяющий действия пользователя и таким образом автоматизировать простые задачи. VBA позволяет создавать формы для общения с пользователем.

Excel в современном мире

Интеллектуальные возможности для профессионалов и новичков

Excel запоминает типичные действия и упорядочивают данные, тем самым экономя ваше время. Без труда создавайте электронные таблицы на базе шаблонов или самостоятельно и выполняйте вычисления с помощью новейших формул.

Более наглядное представление данных

Новые диаграммы и графики помогут вам представить данные убедительно и наглядно. Используйте различные варианты форматирования, спарклайны и таблицы, чтобы облегчить понимание данных. Создавайте прогнозы и отображайте тенденции одним щелчком.

Сферы, где применяется Excel

– Администрирование. Работа со списками и базами данных в приложении позволит быстро вносить и сортировать сведения в таблицах, делать выборки и отчёты.

– Бухгалтерия. Excel может работать в комплексе с другими специализированными разработками. Приложение даёт возможность делать различные формы документов, производить расчёты с коэффициентами, создавать шаблоны для повторяемых операций.

– Экономика и финансовая аналитика. Функционал приложения включает формулы для расчётов пени, неустойки, ссуды, цен с учётом инфляции. Реализует анализ данных любой сложности, выводит прогнозы и отчёты.

– Банковское дело. Excel имеет целый комплексный подход к банковским подсчётам – выводит кредитные процентные ставки, считает доход банка для кредита с заданной суммой и ежемесячные выплаты и сроки погашения.

– Торговля. Софт визуально формирует клиентскую базу, группирует ассортимент товаров, осуществляет создание прайс-листов, накладных, ведёт учёт и рассчитывает программы лояльности.

– Статистика. Сравнение и обработка данных. Глубокий и специализированный анализ огромных объёмов информации.

– Везде, где пользуются компьютером – лаборатории, медицина, социология и прочее, требуется работа с электронными таблицами. Но занимательный excel стал довольно популярным среди обычных пользователей. Электронные таблицы помогают вести семейную бухгалтерию и контролировать бюджет семьи.

На сегодняшний день Excel используют более 750 миллионов человек по некоторым оценкам.

Содержание программы

- Модуль 1. Вычисления. Таблицы. Диаграммы. Формулы и функции.
 - Создание таблиц данных, форматирование таблиц, сортировка, фильтры, поиск и замена данных;
 - Формулы и функции, математические, логические статические, текстовые и другие функции;
 - Графическое представление данных, создание диаграмм, спарклайнов.
- Модуль 2. Анализ данных. Сводные таблицы и диаграммы.
 - Выделение тренда, подбор параметров, поиск решения;
 - Анализ данных (корреляция, регрессия), анализ «что – если»;
 - Сводные таблицы, настройка полей, группировка, преобразование;
 - Сводные диаграммы.
- Модуль 3. Макросы. Разработка инструментов вычислений в VBA.
 - Работа с макрорекордером (быстрое создание макросов);
 - Основы программирования в VBA;
 - Программирование линейных вычислительных процессов;
 - Программирование разветвляющегося вычислительного процесса;
 - Программирование циклов;
 - Создание и дизайн рабочих окон (форм).

Практическое занятие № 1

«Базовые операции в Excel»

Excel – это электронная таблица, которая входит в пакет Microsoft Office. Появление электронных таблиц было одним из важных событий в развитии компьютерных систем, так как позволило производить быстро и качественно расчеты на персональных ЭВМ. По оценкам западных экспертов применение электронных таблиц для расчетов по сравнению с калькулятора повысило производительность в десятки раз.

Ячейки, листы, книги, рабочие области

В любой структуре есть минимальная единица информации. Минимальной единицей информации в Excel является ячейка. Ячейки собираются в листы. Лист в Excel – это набор ячеек.

Книга – это набор листов.

Рабочая область – набор файлов или книг. Для удобства работы несколько книг со связанными рабочими областями можно объединить в рабочую область и у нас будет ощущение одного файла.

Адрес ячейки

Каждая ячейка в Excel имеет адрес. Систем адресации несколько. Адрес ячейки складывается из номера строки и название колонки.

Выделение ячеек

При работе с Excel, одна из ячеек обычно активна. Активная ячейка обведена черным квадратом. Колонка и строка активной ячейки подсвечивается.

Выделять можно и несколько ячеек. Выделяются они, используя левую кнопку мыши и две клавиши на клавиатуры. Первый способ – это выделение с использованием клавиши Shift.

Второй способ – это использование клавиши Ctrl. Она позволяет выделять сложные диапазоны.

Для того чтобы выделить колонку или строку необходимо щелкнуть по имени колонки или строки.

Для того, чтобы выделить всю таблицу сразу, необходимо щелкнуть по пересечению имен строк и колонок.

Ввод, редактирование и удаление данных

Вводить информацию можно прямо в ячейку и в строку формул. Для того, чтобы ввести что-то в ячейку, нужно выделить ее и начать вводить символы с клавиатуры.

Для удаления необходимо выделить одну ячейку или несколько, и нажать Delete.

Специальный адрес ячейки (имя)

Есть ряд ситуаций, когда пользоваться адресом ячейки неудобно или невозможно. Ячейке можно присвоить специальное имя. Для этого нужно нажать на окно с адресом, ввести имя ячейки и нажать Enter, чтобы подтвердить имя. Теперь, при перемещении на эту ячейку, в окне адреса показывается ее имя.

Формат ячейки

Каждая ячейка имеет формат. Это понятие включает много параметров.

- цвет текста и фона;
- выравнивание;
- обрамление;
- отображение.

Для того, чтобы изменить формат ячейки, или просмотреть текущий нужно выделить ячейку. После этого нужно перейти в меню ячейки «Формат», и появится диалоговое окно с настройками формата ячейки.

Вкладки:

- «Число», как будет отображаться информация в ячейке;
- «Выравнивание», как будет располагаться информация в ячейке;
- «Шрифт», каким шрифтом будет отображаться информация в ячейке;
- «Граница», как будет отображаться граница;
- «Заливка», каким будет фон;
- «Защита», для установки защиты ячейки.

После любых изменений окно настройки формата ячейки нужно закрыть.

Объединение ячеек

Для того, что объединить несколько ячеек их надо выделить, перейти в меню формат ячейки, и установить галочку рядом с надписью объединенные ячейки.

После объединения серия ячеек будет вести себя словно одна ячейка. Для нее можно устанавливать формат и рамку. В любой момент есть возможность разъединить объединенные ячейки.

Вставка строк и столбцов

Вставка строк и столбцов производится выше и левее текущей ячейки. Необходимо выбрать ячейку, выше которой будет производиться вставка, перейти в меню

«Вставка» и выбрать строки или столбцы.

Копирование ячеек

Копирование ячейки осуществляется с помощью комбинации клавиш Ctrl + C, вставка – с помощью комбинации клавиш Ctrl + V.

Помимо обычной вставки есть еще и специальная вставка, которая позволяет более детально контролировать результат вставки.

Создание формул

Одна из существенных возможностей Excel – это создание формул. Формула начинается со знака равно (=). В формулах используются адреса ячеек. В формулах можно использовать скобки по правилам математических операций.

Множественные расчеты по формулам

Именно эта возможность электронных таблиц увеличила производительность в десятки раз. Ведь при ручном расчете нужно каждый раз вводить не только цифры, но и математические операции.

Растаскивание

Создание таблиц связано с массовым вводом некоторых чередующихся последовательностей. В Excel есть возможность автоматизации подобных операций. Нужно в соседних ячейках ввести две цифры. Выделить их, и потянуть за квадрат, появившийся в правом нижнем углу выделения ячеек.

Автосумма

В Excel предусмотрена специальная кнопка на панели инструментов, позволяющая автоматически выделять диапазон и подводить итог в виде суммы – «Автосумма».

Пользуясь автосуммированием, необходимо иметь в виду, что Excel выделит только непрерывный диапазон.

Рабочие листы

Ярлычки листов находятся в самом низу книги. Можно переключаться между ними, нажимая левой кнопкой мыши на соответствующем листе.

При необходимости листы можно добавлять. Добавление производится справа от текущего листа.

При работе можно выделять несколько листов для удаления или перетаскивания или печати. Можно использовать клавиши Shift или Ctrl. Выделенные листы окрасятся белым цветом.

Листы можно копировать и из одной открытой книги в другую. Правой кнопкой необходимо щелкнуть по ярлыку листа и выбрать пункт меню «Переместить -> Скопировать». Появится диалоговое окно с выбором, в какую книгу нужно перемещать или копировать.

Для того, чтобы поместить файл на диск и задать ему имя, нужно выбрать пункт меню «Сохранить как».

Практическое занятие №2

«Математические и статистические функции Excel»

В рамках занятия рассматриваются основные математические и статистические функции.

Математические функции:

- =ABS(число) - Возвращает модуль (абсолютную величину) числа.
- =ПИ() - Возвращает число 3,14159265358979 — математическую константу "пи" с точностью до 15 цифр. У функции ПИ нет аргументов.
- =ГРАДУСЫ(угол) - Преобразует радианы в градусы.

- =РАДИАНЫ(угол) - Преобразует градусы в радианы.
- =SIN(число) - Возвращает синус заданного угла. Число задается в радианах.
- =COS(число) - Возвращает косинус заданного угла. Число задается в радианах.
- =TAN(число) - Возвращает тангенс заданного угла. Число задается в радианах.
- =COT(число) - Возвращает котангенс заданного угла. Число задается в радианах.
- =ASIN(число) - Возвращает арксинус числа. Угол определяется в радианах в диапазоне от $-\pi/2$ до $\pi/2$.
- =ACOS(число) - Возвращает арккосинус числа. Угол определяется в радианах в интервале от 0 до "пи".
- =ATAN(число) - Возвращает арктангенс числа. Угол определяется в радианах в диапазоне от $-\pi/2$ до $\pi/2$.
- =ACOT(число) - Возвращает арккотангенс числа. Угол определяется в радианах в интервале от 0 до "пи".
- =SINH(число) - Возвращает гиперболический синус числа.
- =COSH(число) - Возвращает гиперболический косинус числа.
- =TANH(число) - Возвращает гиперболический тангенс числа.
- =COTH(число) - Возвращает гиперболический котангенс числа.
- =ASINH(число) - Возвращает гиперболический арксинус числа.
- =ACOSH(число) - Возвращает гиперболический арккосинус числа.
- =ATANH(число) - Возвращает гиперболический арктангенс числа.
- =ACOTH(число) - Возвращает гиперболический арккотангенс числа.
- =SEC(число) - Возвращает секанс угла.
- =CSC(число) - Возвращает косеканс угла.
- =SECH(число) - Возвращает гиперболический секанс угла. Секанс - отношение длины гипотенузы к длине катета, прилежащего к данному острому углу в прямоугольном треугольнике.
- =CSCH(число) - Возвращает гиперболический косеканс угла. Косеканс - отношение длины гипотенузы к длине катета, противолежащего данному острому углу в прямоугольном треугольнике.
- =EXP(число) - Возвращает число e , возведенное в указанную степень. Число e равно 2,71828182845904 и является основанием натурального логарифма.
- =LOG(число;[основание]) - Возвращает логарифм числа по заданному основанию.
- =LN(число) - Возвращает натуральный логарифм числа. Натуральный логарифм – это логарифм по основанию e (2,71828182845904).
- =LOG10(число) - Возвращает десятичный логарифм числа.
- =ЗНАК(число) - Определяет знак числа. Возвращает значение 1, если число положительное, 0 (ноль), если число равно 0, и -1, если число отрицательное.
- =КОРЕНЬ(число) - Возвращает положительное значение квадратного корня.
- =АРАБСКОЕ(текст) - Преобразует римское число в арабское. Текст - строка, ограниченная кавычками, пустая строка (""), или ссылка на ячейку, содержащую текст.
- =РИМСКОЕ(число;[форма]) - Преобразует арабское число в римское в текстовом формате.
- =СТЕПЕНЬ(число;степень) - Возвращает результат возведения числа в степень.
- =ЦЕЛОЕ(число) - Округляет число до ближайшего меньшего целого.
- =ЧАСТНОЕ(числитель;знаменатель) - Возвращает целую часть результата деления с остатком. Эта функция используется, когда нужно отбросить остаток от деления.
- =ОСТАТ(число;делитель) - Возвращает остаток от деления аргумента "число" на значение аргумента "делитель". Результат имеет тот же знак, что и делитель.
- =ОТБР(число;[число_разрядов]) - Усекает число до целого, отбрасывая дробную часть.

- =ОСНОВАНИЕ(число;основание;[минимальная_длина]) - Преобразует число в текстовое представление с указанным основанием системы счисления.
- =ОКРУГЛТ(число;точность) - ОКРУГЛТ возвращает число, округленное до нужного кратного.
- =ОКРВВЕРХ(число;точность) - Возвращает результат округления с избытком до ближайшего числа, кратного значению точности. Например, если в значениях цен не должны использоваться рубли, а товар стоит 442 рубля, формула =ОКРВВЕРХ(442;10) позволяет округлить цену с точностью до 10 рублей.
- =ОКРВНИЗ(число;точность) - Округляет число до кратного заданной точности с недостатком.
- =ЧИСЛКОМБ(число;число_выбранных) - Возвращает количество комбинаций для заданного числа элементов. Функция ЧИСЛКОМБ используется для определения общего числа всех групп, которые можно составить из элементов данного множества.
- =ЧЁТН(число) - Возвращает число, округленное до ближайшего четного целого. Эту функцию можно использовать при обработке объектов, которые поступают парами. Например, в ящик можно упаковывать предметы по два в ряд. Ящик будет заполнен, если количество предметов, округленное до ближайшего четного числа, будет равно вместимости ящика.
- =НЕЧЁТ(число) - Возвращает число, округленное до ближайшего нечетного целого.
- =ФАКТР(число) - Возвращает факториал числа. Факториал числа – это значение, равное $1*2*3*...*$ число.
- =НОД(число1;[число2];...) - Возвращает наибольший общий делитель двух или более целых чисел. Наибольший общий делитель — это наибольшее целое число, на которое делятся число1 и число2 без остатка.
- =НОК(число1;[число2];...) - Возвращает наименьшее общее кратное целых чисел. Наименьшее общее кратное — это наименьшее положительное целое, которое кратно всем целым аргументам "число1", "число2" и т. д. Функция НОК используется для сложения дробей с различными знаменателями.
- =ПРОИЗВЕД(число1;[число2];...) - Функция ПРОИЗВЕД перемножает все числа, переданные как аргументы, и возвращает произведение.
- =СЛЧИС() - Функция СЛЧИС возвращает равномерно распределенное случайное вещественное число, большее или равное 0, но меньшее 1. При каждом пересчете листа возвращается новое случайное вещественное число.
- =СЛУЧМЕЖДУ(нижн_граница;верхн_граница) - Возвращает случайное целое число, находящееся в диапазоне между двумя заданными числами. При каждом вычислении листа возвращается новое случайное целое число.
- =СУММ(число1;[число2];...) - Функция СУММ суммирует значения. Вы можете складывать отдельные значения, диапазоны ячеек, ссылки на ячейки или данные всех этих трех видов.
- =СУММЕСЛИ(диапазон; условие; [диапазон_суммирования]) - Функция СУММЕСЛИ используется, если необходимо просуммировать значения диапазон, соответствующие указанному критерию. Предположим, например, что в столбце с числами необходимо просуммировать только значения, превышающие 5. Для этого можно использовать следующую формулу: =СУММЕСЛИ(B2:B25;"> 5")
- =СУММКВ(число1;[число2];...) - Возвращает сумму квадратов аргументов.

Статистические функции:

- =СРЗНАЧ(число1;[число2];...) - Возвращает среднее арифметическое аргументов. Например, если диапазон A1:A20 содержит числа, формула =СПБ(A1:A20) возвращает среднее из этих чисел.

- =СРЗНАЧЕСЛИ(диапазон, условия, [диапазон_усреднения]) - Возвращает среднее значение (среднее арифметическое) всех ячеек в диапазоне, которые соответствуют данному условию.

- =СЧЁТ(значение1;[значение2];...) - Функция СЧЁТ подсчитывает количество ячеек, содержащих числа, и количество чисел в списке аргументов.

- =СЧЁТЕСЛИ(диапазон;критерий) - С помощью статистической функции СЧЁТЕСЛИ можно подсчитать количество ячеек, отвечающих определенному условию (например, число клиентов в списке из определенного города).

- =МАКС(число1;[число2];...) - Возвращает наибольшее значение из набора значений.

- =МИН(число1;[число2];...) - Возвращает наименьшее значение в списке аргументов.

- =МЕДИАНА(число1;[число2];...) - Возвращает медиану заданных чисел. Медиана — это число, которое является серединой множества чисел.

- =МОДА(число1;[число2];...) - Возвращает наиболее часто повторяющееся значение в массиве или диапазоне данных.

Практическое занятие №3

«Логические и текстовые функции Excel»

В рамках занятия рассматриваются основные логические и текстовые функции.

Логические функции:

- =ЕСЛИ(лог_выражение; значение_если_истина; [значение_если_ложь]) - Функция ЕСЛИ — одна из самых популярных функций в Excel. Она позволяет выполнять логические сравнения значений и ожидаемых результатов. Поэтому у функции ЕСЛИ возможны два результата. Первый результат возвращается в случае, если сравнение истинно, второй — если сравнение ложно. Например, функция =ЕСЛИ(С2="Да";1;2) означает следующее: ЕСЛИ(С2="Да", то вернуть 1, в противном случае вернуть 2).

- =И(логическое_значение1;[логическое_значение2];...) - Функция И возвращает значение ИСТИНА, если в результате вычисления всех аргументов получается значение ИСТИНА, и значение ЛОЖЬ, если вычисление хотя бы одного из аргументов дает значение ЛОЖЬ.

- =ИЛИ(логическое_значение1;[логическое_значение2];...) - Функция ИЛИ возвращает значение ИСТИНА, если в результате вычисления хотя бы одного из ее аргументов получается значение ИСТИНА, и значение ЛОЖЬ, если в результате вычисления всех ее аргументов получается значение ЛОЖЬ.

- =НЕ(логическое_значение) - Функция НЕ меняет значение своего аргумента на обратное. Обычно функция НЕ используется для расширения возможностей других функций, выполняющих логическую проверку.

- =ИСТИНА() - Возвращает логическое значение ИСТИНА. Эту функцию можно использовать, если значение ИСТИНА должно возвращаться при выполнении условия.

- =ЛОЖЬ() - Возвращает логическое значение ЛОЖЬ.

- =ЕСЛИОШИБКА(значение;значение_если_ошибка) - Функцию ЕСЛИОШИБКА можно использовать для перебора и обработки ошибок в формуле. Если же формула возвращает значение, определяемую формулой, возвращается ошибка; в противном случае возвращается результат формулы.

Текстовые функции:

- =СИМВОЛ(число) - Возвращает знак с заданным кодом. Функция СИМВОЛ позволяет преобразовать числовые коды знаков, которые получены из файлов с других компьютеров, в знаки данного компьютера.

- =КОДСИМВ(текст) - Возвращает числовой код первого знака в текстовой строке. Возвращаемый код соответствует набору знаков, используемому на данном компьютере.

- =СЦЕПИТЬ(текст1;[текст2];...) - Используйте функцию СЦЕПИТЬ, относящуюся к текстовым функциям, для объединения двух или нескольких текстовых строк в одну.

- =РУБЛЬ(число;[число_знаков]) - Функция РУБЛЬ, одна из функций ТЕКСТ, преобразует число в текст с использованием формата валюты с округленными десятичными числами до замещенного количества мест.

- =СОВПАД(текст1;текст2) - Сравнивает две строки текста и возвращает значение ИСТИНА, если они в точности совпадают, и ЛОЖЬ — в противном случае. Функция СОВПАД учитывает регистр, но игнорирует различия в форматировании. Функция СОВПАД позволяет проверить текст, вводимый в документ.

- =НАЙТИ(искомый_текст;просматриваемый_текст;[нач_позиция]) - Функция НАЙТИ находит вхождение одной текстовой строки в другую и возвращают начальную позицию искомой строки относительно первого знака второй строки. Функции НАЙТИ и НАЙТИБ работают с учетом регистра и не позволяют использовать подстановочные знаки. Если необходимо выполнить поиск без учета регистра или использовать подстановочные знаки, воспользуйтесь функцией ПОИСК или ПОИСКБ.

- =ПОИСК(искомый_текст;просматриваемый_текст;[начальная_позиция]) - Функция ПОИСК находит одну текстовую строку в другой и возвращают начальную позицию первой текстовой строки (считая от первого символа второй текстовой строки). Функции ПОИСК и ПОИСКБ не учитывают регистр. Если требуется учитывать регистр, используйте функции НАЙТИ и НАЙТИБ.

- =ЛЕВСИМВ(текст;[число_знаков]) - Функция ЛЕВСИМВ возвращает первый символ или несколько первых символов текстовой строки на основе заданного числа символов.

- =ПРАВСИМВ(текст;[число_знаков]) - Функция ПРАВСИМВ возвращает последний символ или несколько последних символов текстовой строки на основе заданного числа символов.

- =ДЛСТР(текст) - Функция ДЛСТР возвращает число символов в текстовой строке.

- =СТРОЧН(текст) - Преобразует знаки в текстовой строке из верхнего регистра в нижний.

- =ПРОПИСН(текст) - Делает все буквы в тексте прописными.

- =ПРОПНАЧ(текст) - Первая буква в строке текста и все первые буквы, следующие за знаками, отличными от букв, преобразуются в прописные (верхний регистр). Все прочие буквы в тексте преобразуются в строчные (нижний регистр).

- =ПСТР(текст;начальная_позиция;число_знаков) - Функция ПСТР возвращает заданное число знаков из текстовой строки, начиная с указанной позиции.

- =ЗАМЕНИТЬ(стар_текст;начальная_позиция;число_знаков;нов_текст) - Функция ЗАМЕНИТЬ заменяет указанное число символов текстовой строки другой текстовой строкой.

- =ПОДСТАВИТЬ(текст;стар_текст;нов_текст;[номер_вхождения]) - Подставляет значение аргумента "нов_текст" вместо значения аргумента "стар_текст" в текстовой строке. Функция ПОДСТАВИТЬ используется, когда нужно заменить определенный текст в текстовой строке; функция ЗАМЕНИТЬ используется, когда нужно заменить любой текст начиная с определенной позиции.

- =ПОВТОР(текст;число_повторений) - Повторяет текст заданное число раз. Функция ПОВТОР используется для заполнения ячейки заданным количеством вхождений текстовой строки.

- =СЖПРОБЕЛЫ(текст) - Удаляет из текста все пробелы, за исключением одиночных пробелов между словами. Функция СЖПРОБЕЛЫ используется для обработки текстов, полученных из других прикладных программ, если эти тексты могут содержать лишние пробелы.

- =ТЕКСТ(значение; формат) - С помощью функции ТЕКСТ можно изменить представление числа, применив к нему форматирование с кодами форматов. Это полезно в ситуации, когда нужно отобразить числа в удобочитаемом виде либо объединить их с текстом или символами.

Самостоятельное занятие № 1

«Применение математических, статистических, логических и текстовых функций»

Цель: освоить на практике математические, статистические, логические и текстовые функции.

Задание:

1. Математические функции:

- Решить два примера из выданных преподавателем.
- Продемонстрировать работу одной из функций округления.
- Сформировать ряд случайных чисел по определенной тематике (список цен, оценок в университете, дней в месяце / году или другое).
- Продемонстрировать суммирование значений с условием (среди ранее сформированных).

2. Статистические функции:

- Продемонстрировать поиск среднего значения с условием (среди ранее сформированных).
- Продемонстрировать подсчет количества значений с условием (среди ранее сформированных).

3. Логические функции:

- Решить два примера (согласно варианту из журнала) из выданных преподавателем.
- Придумать пример совместного использования формул ЕСЛИ и И (или ЕСЛИ и ИЛИ). Другими словами, выполнить проверку двух условий, или одного из двух.

4. Текстовые функции:

- Привести пример использования формулы ТЕКСТ (дата, время, проценты, пользовательский формат или другое).
- Записать в ячейку любимую цитату и заменить одно слово на другое.
- Записать предложение. Проверить, входит ли определенное слово в это предложение, и если да, выдать в другую ячейку какое-нибудь сообщение (если нет, выдать другое сообщение).

Практическое занятие № 4

«Работа с одномерными и двумерными массивами. Ссылки»

В рамках занятия рассматриваются функции для работы со ссылками, а также одномерными и двумерными массивами.

- =ВПР(искмое_значение, таблица, номер_столбца, [интервальный_просмотр])

Функция ВПР используется для поиска значения в таблице. Для построения синтаксиса функции ВПР потребуется следующая информация:

1. Значение, которое вам нужно найти, то есть искомое значение.
2. Диапазон, в котором находится искомое значение. Помните, что для правильной работы функции ВПР искомое значение всегда должно находиться в первом столбце

диапазона. Например, если искомое значение находится в ячейке С2, диапазон должен начинаться с С.

3. Номер столбца в диапазоне, содержащий возвращаемое значение. Например, если в качестве диапазона указать В2:D11, следует посчитать В первым столбцом, С — вторым и так далее.

4. При желании вы можете указать слово ИСТИНА, если вам достаточно приблизительного совпадения, или слово ЛОЖЬ, если вам требуется точное совпадение возвращаемого значения. Если вы ничего не указываете, по умолчанию всегда подразумевается вариант ИСТИНА, то есть приблизительное совпадение.

- =ГПР(искомое_значение;таблица;номер_строки;[интервальный_просмотр])

Выполняет поиск значения в первой строке таблицы или массив значений и возвращает значение, находящееся в том же столбце в заданной строке таблицы или массива. Функция ГПР используется, когда сравниваемые значения расположены в первой строке таблицы данных, а возвращаемые — на несколько строк ниже. Если сравниваемые значения находятся в столбце слева от искомых данных, используйте функцию ВПР.

- =СТРОКА([ссылка]) - Возвращает номер строки, определяемой ссылкой.

• =СТОЛБЕЦ([ссылка]) - Функция СТОЛБЕЦ возвращает номер столбца заданного ссылка на ячейку.

• =АДРЕС(номер_строки;номер_столбца;[тип_ссылки];[a1];[имя_листа]) - Функцию АДРЕС можно использовать для получения адреса ячейки на листе по номерам строки и столбца.

• =ДВССЫЛ(ссылка_на_текст;[a1]) - Возвращает ссылку, заданную текстовой строкой. Ссылки немедленно вычисляются для вывода их содержимого. Функция ДВССЫЛ используется, если требуется изменить ссылку на ячейку в формуле без изменения самой формулы.

- =ЧСТРОК(массив) - Возвращает количество строк в ссылке или массив.

- =ЧИСЛОСТОЛБ(массив) - Возвращает количество столбцов в массив или ссылке.

- =СМЕЩ(ссылка;смещ_по_строкам;смещ_по_столбцам;[высота];[ширина]) -

Данная функция возвращает ссылку на диапазон, отстоящий от ячейки или диапазона ячеек на заданное число строк и столбцов. Возвращаемая ссылка может быть отдельной ячейкой или диапазоном ячеек. Можно задавать количество возвращаемых строк и столбцов.

- =ПОИСКПОЗ(искомое_значение;просматриваемый_массив;[тип_сопоставления])—

Функция ПОИСКПОЗ выполняет поиск указанного элемента в диапазоне ячеек и возвращает относительную позицию этого элемента в диапазоне.

Работа с матрицами

1. Создание двумерного массива (матрицы) осуществляется простым заполнением области ячеек.

2. Сложение матриц.

Для сложения матриц необходимо ввести формулу =ДИАПАЗОН1 + ДИАПАЗОН2, выделяя диапазон мышкой или ввода значения с клавиатуры. Диапазон обозначается верхней левой и нижней правой ячейками через двоеточие.

Если вы введете формулу с диапазонами в одну ячейку и нажмете Enter, то получите значение #ЗНАЧ!.

Чтобы отобразить матрицу с результатом сложения, необходимо действовать в следующей последовательности.

Выделить диапазон ячеек соответственно количеству строк и столбцов в результирующей матрице от левой верхней ячейки на правой нижней, затем ввести формулу (сразу или используя строку формул), ввести формулу и нажать Shift + Ctrl + Enter.

3. Вычитание матриц.

Алгоритм вычитания подобен алгоритму сложения, за исключением знака самой операции.

Необходимо выделить поле ячеек для результирующей таблицы, ввести формулу (в ячейку или в строке формул) и нажать Shift + Ctrl + Enter.

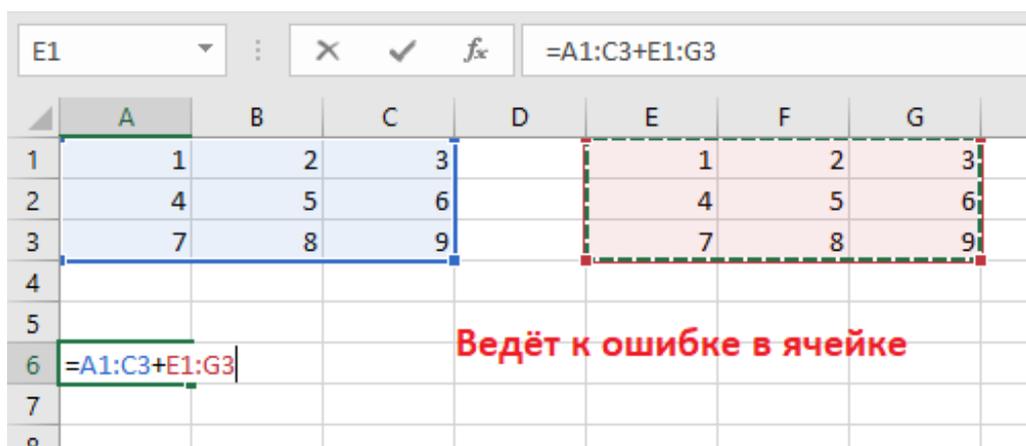


Рис. 1. Ошибка при сложении матриц

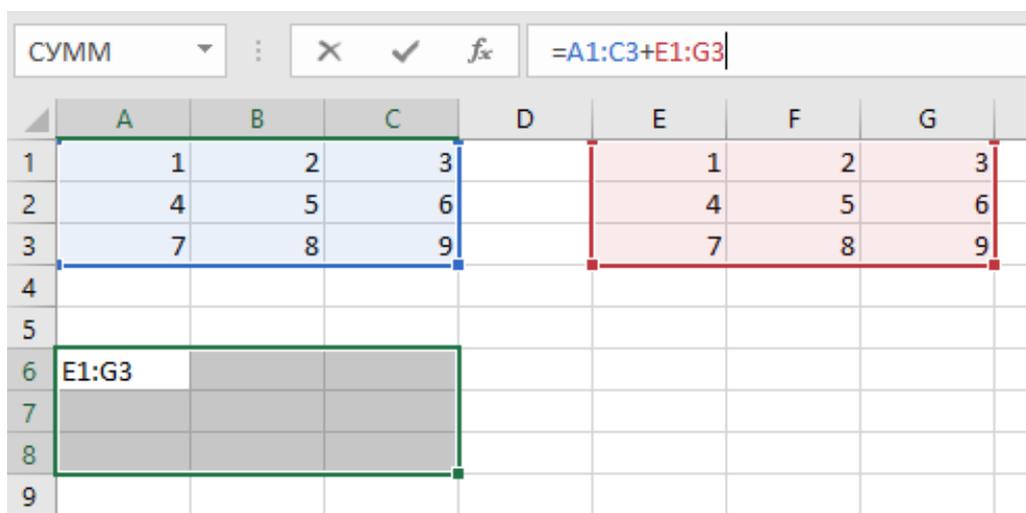


Рис. 2. Сложение матриц

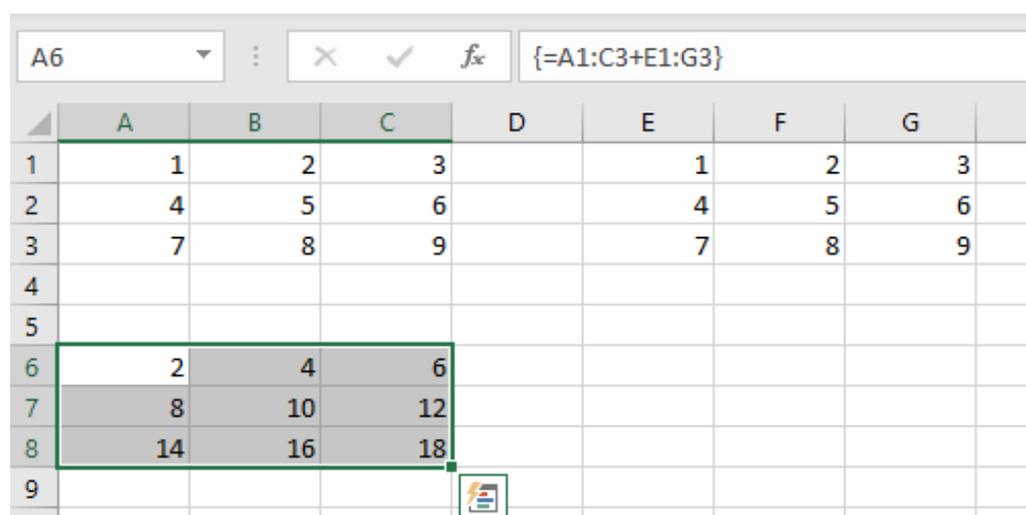


Рис. 3. Результат сложения матриц

4. Умножение матриц.

Чтобы умножить матрицу на число, нужно каждый ее элемент умножить на это число. Формула в Excel: =A1*\$E\$3 (ссылка на ячейку с числом должна быть абсолютной).

Умножим матрицу на матрицу разных диапазонов. Найти произведение матриц можно только в том случае, если число столбцов первой матрицы равняется числу строк второй.

Для умножения матриц используется формула =МУМНОЖ(массив1; массив2).

Функция МУМНОЖ возвращает матричное произведение двух массивов. Результатом является массив с таким же числом строк, что и массив1, и с таким же числом столбцов, что и массив2.

В результирующей матрице количество строк равняется числу строк первой матрицы, а количество колонок – числу столбцов второй.

5. Транспонирование матриц.

Транспонировать матрицу – поменять строки и столбцы местами.

Это можно сделать двумя способами.

Способ №1. Выделить матрицу (все ячейки), нажать Ctrl + C для копирования, выбрать ячейку для размещения транспонированной матрицы, нажать правую кнопку мыши и выбрать “Специальная вставка”. Там отметить опцию “транспонировать” и нажать ОК.

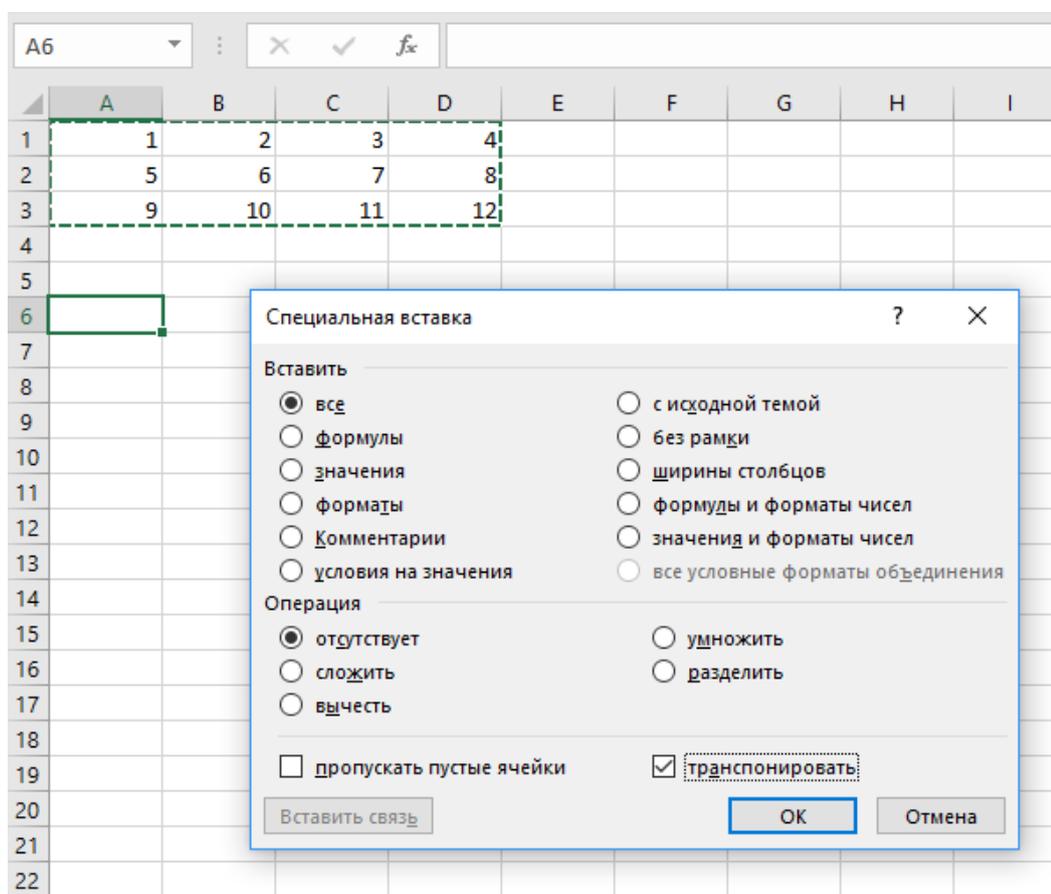


Рис. 4. Специальная вставка для транспонирования матриц

Способ №2. Использование функции ТРАНСП.

Необходимо выделить мышкой поле ячеек под транспонированную матрицу, ввести формулу =ТРАНСП(массив), и нажать Shift + Ctrl + Enter.

6. Нахождение обратной матрицы.

Ее имеет смысл находить, если мы имеем дело с квадратной матрицей (количество строк и столбцов одинаковое).

Размерность обратной матрицы соответствует размеру исходной.

Для нахождения обратной матрицы используется формула МОБР.

=МОБР(массив) - Функция МОБР возвращает обратную матрицу для матрицы, хранящейся в массиве.

Порядок такой же, как и в предыдущих случаях. Необходимо выделить поле ячеек под обратную матрицу, ввести формулу и нажать Shift + Ctrl + Enter.

7. Нахождение определителя матрицы.

Определитель матрицы можно вычислить только для квадратных матриц, т.е. у которых количество строк равно количеству столбцов.

Для вычисления определителя есть специальная функция МОПРЕД(массив). В аргументе функции необходимо указать ссылку на диапазон ячеек (массив), содержащий элементы матрицы.

Самостоятельное занятие №2

«Применение средств работы с массивами и ссылками»

Цель: освоить на практике работу с массивами и ссылками.

Задание:

1. Матрицы:

- Решить три примера по матрицам согласно своему варианту по журналу.

2. Ссылки и массивы:

- Решить задание по функции ВПР согласно своему варианту по журналу.

Внедрить в задание использование списков (через “Данные” - “Проверка данных”).

- Продемонстрировать использование функций АДРЕС, ДВССЫЛ и ПОИСКПОЗ.

Практическое занятие №5

«Диаграммы и спарклайны в Excel»

Диаграммы позволяют наглядно представить данные, чтобы произвести наибольшее впечатление на аудиторию. Можно начать работу с рекомендуемой диаграммы или выбрать один из встроенных шаблонов диаграмм.

Создание диаграммы

Чтобы создать диаграмму, необходимо выбрать хотя бы одну ячейку в диапазоне данных (наборе ячеек). На вкладке “Вставка” нажмите кнопку “Рекомендуемые диаграммы”. На вкладке “Рекомендуемые диаграммы” выберите диаграмму для предварительного просмотра. Выберите диаграмму, нажмите кнопку ОК.

Упорядочение данных для диаграмм

Excel может порекомендовать диаграммы для ваших данных. Предлагаемые диаграммы зависят от того, как вы расположили данные. Кроме того, у вас могут быть собственные диаграммы.

Гистограмма, линейчатая, график, с областями, поверхностная или лепестковая

Упорядочение данных осуществляется по столбцам или по строкам.

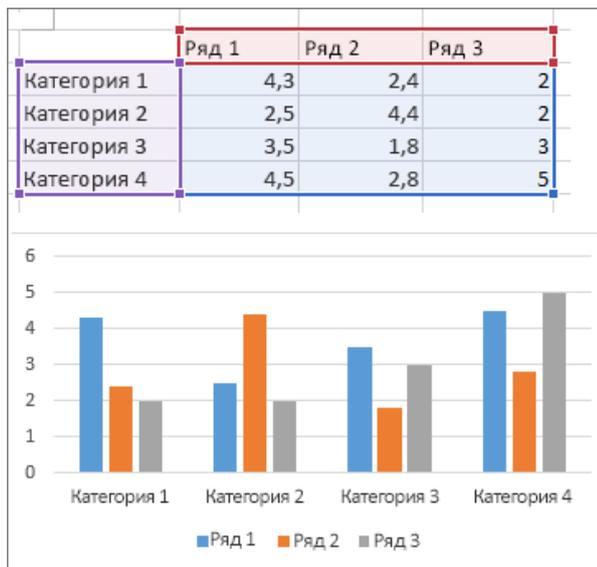


Рис. 5. Гистограмма

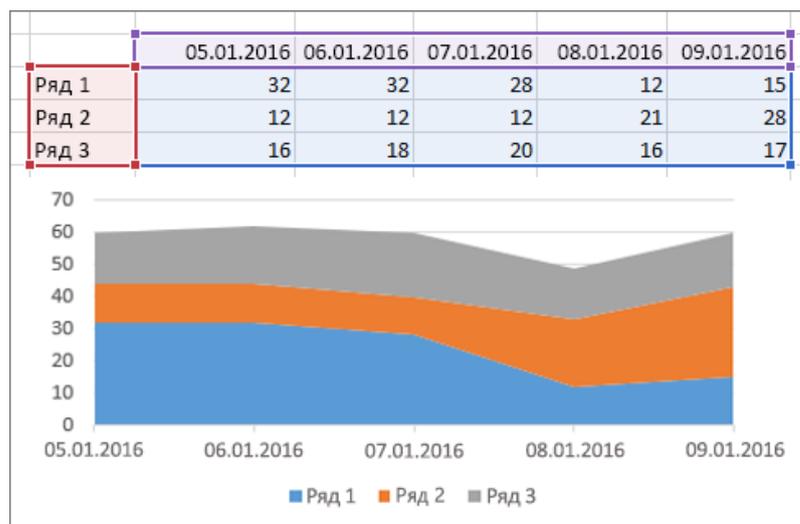


Рис. 6. Диаграмма с областями

Круговая диаграмма

Эта диаграмма использует один набор значений ("ряд данных").

Упорядочение данных осуществляется по одному столбцу или строке и одному столбцу.

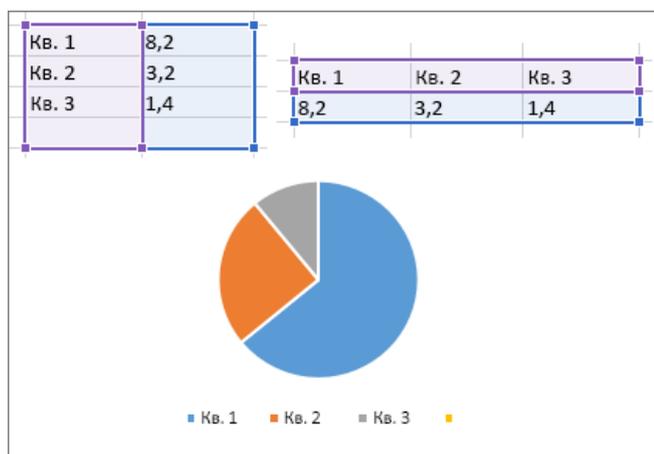


Рис. 7. Круговая диаграмм

Кольцевая диаграмма

На этой диаграмме может отображаться один или несколько рядов данных. Упорядочение данных осуществляется по одному или нескольким столбцам или строкам данных.

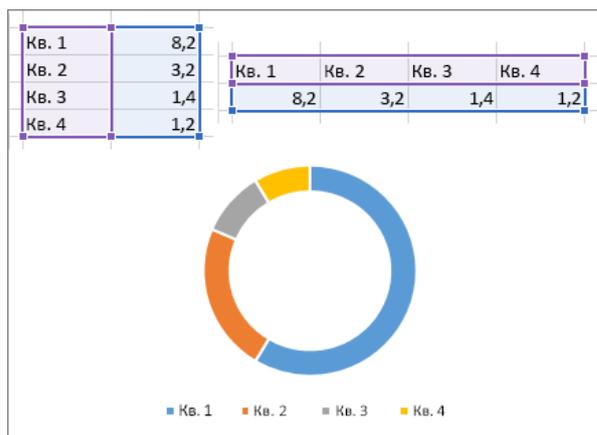


Рис. 8. Кольцевая диаграмма

Точечная или пузырьковая диаграмма

Данные расположены по столбцам, причем значения x – в первом столбце, а значения y – в следующем.

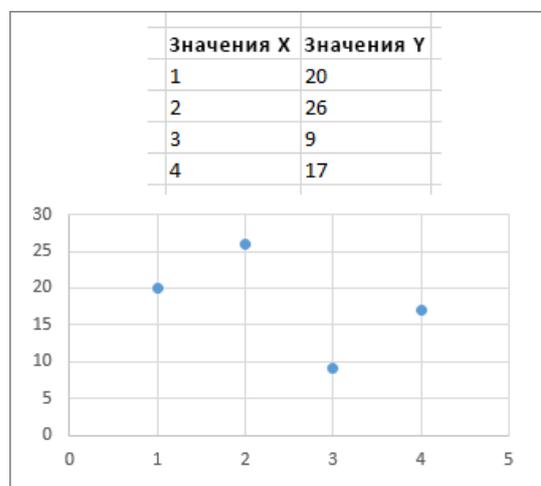


Рис. 9. Точечная диаграмма

В пузырьковой диаграмме добавляется третий столбец для указания размера пузырьков, используемых для представления точек данных в рядах данных.



Рис. 10. Пузырьковая диаграмма

Биржевая диаграмма

Упорядочение данных осуществляется по столбцам или по строкам, используя комбинацию курса открытия, самого высокого курса, самого низкого курса, курса закрытия, а также имена или даты в качестве меток данных в правильном порядке.

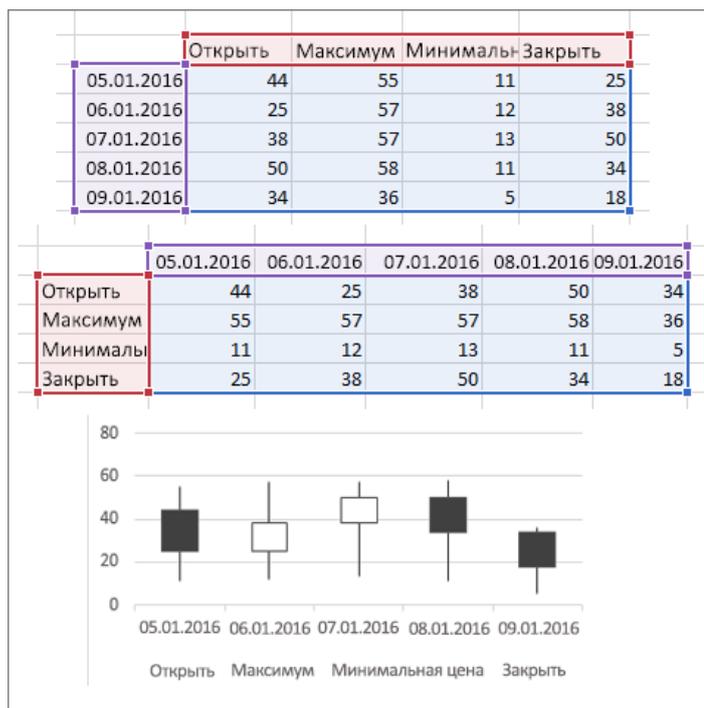


Рис. 11. Биржевая диаграмма

Добавление ряда данных в диаграмму

После создания диаграммы может потребоваться добавить в нее еще один ряд данных. Ряд данных – это строка или столбец чисел, которые вводятся на листе и отображаются на диаграмме, например список значений прибыли за квартал.

Можно ввести новый ряд данных, который нужно добавить, в ячейки на листе, расположенные рядом с исходными данными диаграммы или под ними.

Отображаемые исходные данные выделяются на листе, и появляются маркеры изменения размера. Необходимо перетащить маркеры изменения размера на листе, чтобы включить новые данные. Диаграмма обновится автоматически и отобразит новый добавленный ряд данных.

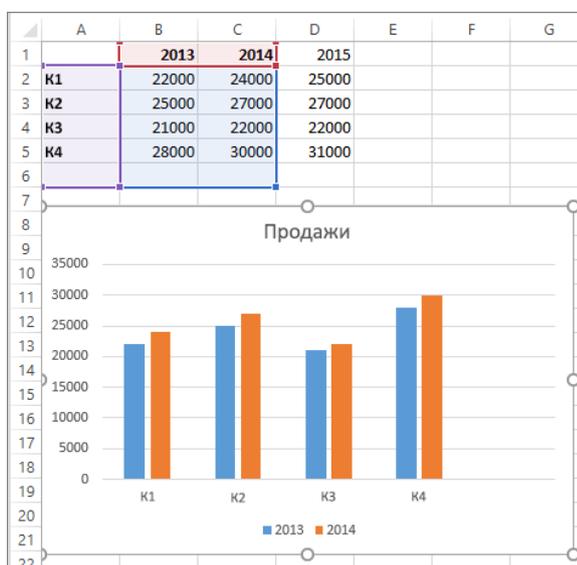


Рис. 12. Добавление ряда данных

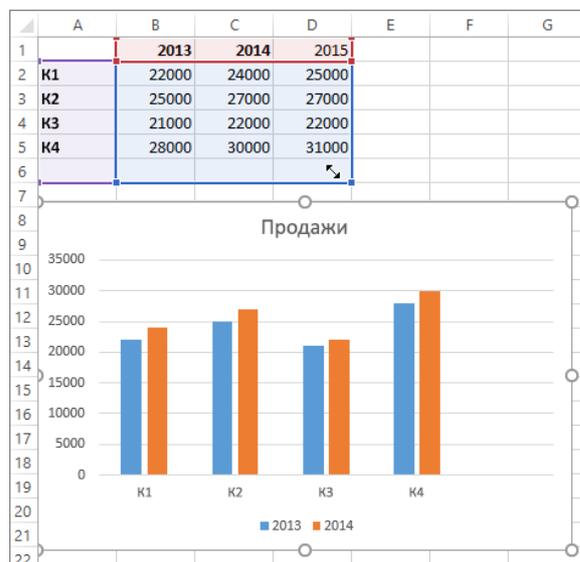


Рис. 13. Добавление ряда данных (готовый вид)

Изменение ряда

Чтобы изменить ряд данных, нужно щелкнуть по диаграмме правой кнопкой мыши и выбрать «Выбрать данные». В поле «Элементы легенды (ряды)» выбрать ряд, который нужно изменить. Нажать кнопку «Изменить», внести изменения и нажать кнопку ОК.

Вносимые изменения могут привести к разрыву связей с исходными данными на листе.

Фильтрация данных на диаграмме

Чтобы произвести фильтрацию данных на диаграмме, нужно нажать в любом месте диаграммы, затем нажать кнопку «Фильтры диаграммы» рядом с диаграммой, на вкладке «Значения» выбрать ряды или категории, которые нужно отобразить или скрыть.

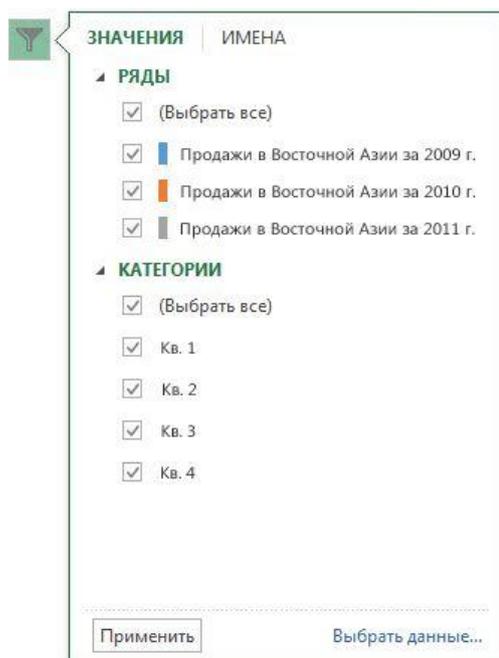


Рис. 14. Фильтрация данных

Добавление линии тренда

Чтобы добавить линию тренда, нужно выбрать диаграмму. На вкладке «Конструктор» необходимо нажать кнопку «Добавить элемент диаграммы», выбрать пункт «Линия тренда», а затем указать тип линии тренда: Линейная, Экспоненциальная, Линейный прогноз или Скользящее среднее.

Спарклайны

Спарклайны – это небольшие диаграммы внутри отдельных ячеек на листе. Благодаря своей компактности спарклайны выразительно и наглядно показывают закономерности в больших наборах данных. С помощью спарклайнов можно демонстрировать тренды в рядах значений (например, сезонные увеличения и уменьшения или экономические циклы) либо выделять максимальные и минимальные значения. Спарклайн выглядит понятнее всего, когда он расположен рядом с соответствующим набором данных. Чтобы создать спарклайн, нужно выделить диапазон значений, которые требуется проанализировать, а затем выбрать место для спарклайна.

	A	B	C	D	E
1	Магазин	Январь	Февраль	Март	Доход с начала года
2	Саратов	р. 4 873,00	р. 11 766,00	р. 7 712,00	
3	Пермь	р. 9 575,00	р. 7 135,00	р. 5 575,00	
4	Санкт-Петербург	р. 12 011,00	р. 9 373,00	р. 3 386,00	

Рис. 15. Диапазон данных для спарклайнов

Вставка спарклайнов

Выберите диапазон данных для спарклайнов:
B2:D4

Укажите, где следует разместить спарклайны:
\$E\$2:\$E\$4

[Подробнее о спарклайнах](#) Отмена OK

Рис. 16. Расположение спарклайнов

	A	B	C	D	E
1	Магазин	Январь	Февраль	Март	Доход с начала года
2	Саратов	р. 4 873,00	р. 11 766,00	р. 7 712,00	▬ ▬ ▬
3	Пермь	р. 9 575,00	р. 7 135,00	р. 5 575,00	▬ ▬ ▬
4	Санкт-Петербург	р. 12 011,00	р. 9 373,00	р. 3 386,00	▬ ▬ ▬

Рис. 17. Пример спарклайнов столбцов

Создание спарклайнов

Чтобы создать спарклайн, нужно выбрать диапазон данных. На вкладке «Вставка» нужно нажать кнопку «Спарклайны» и выбрать нужный тип спарклайна.

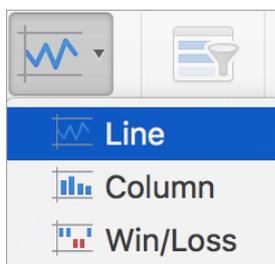


Рис. 18. Виды спарклайнов

В диалоговом окне «Вставка спарклайнов» нужно обратить внимание на то, что первое поле уже заполнено с учетом того, что выбрано на шаге 1.

Нужно выделить на листе ячейку или диапазон ячеек, куда нужно поместить спарклайн, и нажать кнопку ОК.

Отметка точек данных на спарклайнах

С помощью маркеров можно выделять отдельные точки данных, такие как максимальные и минимальные значения.

На вкладке «Конструктор спарклайнов» можно выбрать нужные маркеры и настроить цвета.

Изменение стиля спарклайна

На вкладке «Конструктор спарклайнов» можно выбрать нужный стиль.



Рис. 19. Стили спарклайнов

Удаление спарклайнов

На вкладке «Конструктор спарклайнов» нужно нажать на стрелку рядом с кнопкой «Очистить» и выберите нужный вариант.

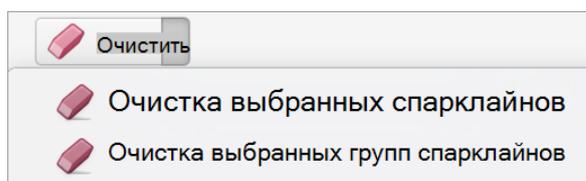


Рис. 20. Удаление спарклайна

Самостоятельное занятие №3

«Графическое представление данных»

Цель: освоить на практике работу с диаграммами и спарклайнами.

Задание:

1. Построить графики функций (3 графика), согласно номеру в журнале. Для всех графиков использовать различное форматирование на свой вкус. Постарайтесь внести максимум различий.

2. Продемонстрировать использование диаграмм трех других видов (кроме обычного графика). Для этого создать 3 простые таблицы (можно использовать функции СЛУЧМЕЖДУ и СЛЧИС). Обязательно произведите форматирование диаграмм.

3. Создать таблицу с данными за разные годы (минимум 3 строки) или используйте другую тематику. Показать на всех строках разные спарклайны («график», «гистограмма», «выигрыш/проигрыш»). Выполнить форматирование.

Практическое занятие №6

«Решение нелинейных уравнений в Excel»

Программа MS Excel имеет встроенные средства, с помощью которых можно без программирования решать нелинейные уравнения и осуществлять поиск максимального и минимального значений функции одной переменной в заданных границах.

В общем случае решение нелинейного уравнения проводится численно в два этапа (здесь речь идет лишь о вещественных корнях уравнения). На первом этапе производится поиск интервалов, в которых содержится только по одному корню. Второй этап решения связан с уточнением корня в выбранном интервале (определением значения корня с заданной точностью). Известно, что корень уравнения (уравнение записано в виде $f(x)=0$) – это такое значение аргумента, при котором значение функции равно нулю. В графическом представлении – это может быть точка пересечения или касания графика функции с осью абсцисс.

При решении уравнения не надейтесь никогда найти точное значение корня и добиться обращения функции в нуль при использовании компьютера, где сами числа представлены ограниченным числом знаков. Здесь критерием может служить приемлемая абсолютная или относительная погрешность корня. Если, например, относительная погрешность равна 0,000001 ($=0,000001$), то искомым результатом будет иметь 6 верных (значащих) цифр после запятой ($n=\lg(1/)$).

В настоящей работе решение уравнений сводится к выполнению второго этапа, то есть к уточнению корня на заданном отрезке.

Решение нелинейных уравнений в таблицах MS Excel возможно осуществить следующими способами:

- посредством «Подбора параметра»;
- с помощью «Поиска решения».

Подбор параметра

При подборе параметра Excel использует итерационный (циклический) процесс. Количество итераций и точность (относительная погрешность) устанавливаются через Файл -> Параметры -> Формулы.

При подборе параметра Excel изменяет значение аргумента функции в одной конкретной ячейке до тех пор, пока значения функции, вычисляемые по формуле, ссылающейся на эту ячейку, не станут соответствовать установленным параметрам вычислений.

Уточнение корня уравнения этим способом сводится к следующим действиям.

1. Заданное уравнение преобразовать к виду $f(x)=0$. Левая часть уравнения и будет той функцией, нуль которой необходимо найти. Например, задано уравнение $\text{tg}(x)=1/x$. Приводим его к виду $x*\text{tg}(x)=1$, переносим единицу в левую часть уравнения и получаем $x*\text{tg}(x)-1=0$. Тогда функция, нуль которой предстоит найти, имеет вид $f(x)=x*\text{tg}(x)-1$.

2. В выбранную ячейку рабочего листа (например, B5) ввести текст $x=$.

3. В соседнюю справа ячейку (например, в ячейку C5) ввести любое начальное приближение к корню из заданного отрезка (можно использовать значение левой или правой границы).

4. В ячейку строкой ниже (например, B6) ввести текст $f(x)=$.

5. В соседнюю ячейку (справа от предыдущей) ввести выражение для вычисления значений функции, в качестве которой использовать левую часть уравнения (в нашем случае в ячейку C6 ввести формулу $=C5*\text{TAN}(C5)-1$. Ссылка в формуле вводится щелчком мыши по ячейке с начальным значением аргумента, то есть по ячейке C5.

6. Щёлкнуть мышью по ячейке с формулой для вычисления значений функции (C6).

7. Выбрать вкладку «Данные» -> «Анализ «что-если»» -> «Подбор параметра».

8. В появившемся диалоговом окне «Подбор параметра» удалить адрес текущей ячейки в окне «Установить в ячейке:», если он не соответствует адресу ячейки с

выражением для вычисления значений функции, и щёлкнуть мышью по ячейке с формулой (С6), в окне «Значение:» ввести 0 (нуль). Щелкнуть мышью в окне «Изменяя значение ячейки:», а затем щёлкнуть мышью по ячейке со значением x (С5).

9. Щёлкнуть мышкой по кнопке ОК. Результат получен.

Поиск решения

Задачи, которые можно решать с помощью Поиска решения, в общей постановке формулируются так.

Найти:

- x_1, x_2, \dots, x_n такие, что $F(x_1, x_2, \dots, x_n) \rightarrow (\text{Max}; \text{Min}; = \text{Value})$
- при ограничениях: $G(x_1, x_2, \dots, x_n) \rightarrow (\text{Value}; \text{Value}; = \text{Value})$, где Value - это значение.

Искомые переменные x_1, x_2, \dots, x_n - ячейки рабочего листа - называются регулируемыми ячейками.

Целевая функция $F(x_1, x_2, \dots, x_n)$ должна задаваться в виде формулы в ячейке рабочего листа. Эта формула может содержать функции, определённые пользователем, и должна зависеть (ссылаться) от регулируемых ячеек. В момент постановки задачи определяется, что делать с целевой функцией. Возможен выбор одной из следующих задач:

- найти максимум целевой функции;
- найти минимум целевой функции;
- добиться того чтобы целевая функция принимала фиксированное значение: $F(x_1, x_2, \dots, x_n) = a$.

Функции $G(x_1, x_2, \dots, x_n)$ называются ограничениями. Их можно задать как в виде равенств, так и неравенств. На регулируемые ячейки можно наложить и другие дополнительные ограничения.

Для уточнения корня с помощью Поиска решения необходимо выполнить следующие действия:

1. в ячейку (например, I5) ввести текст $x =$;
2. в ячейку справа от I5 (K5) ввести значение начальной границы (число) заданного отрезка;
3. в соседнюю ячейку снизу (I6) ввести текст $f(x) =$;
4. в расположенную справа от I6 ячейку (K6) ввести формулу для вычисления значений функции, в качестве которой использовать левую часть преобразованного уравнения (в нашем случае формула имеет вид $=K5 * \text{TAN}(K5) - 1$;
5. щелкнуть мышью по ячейке с целевой функцией (K6);
6. выбрать вкладку Данные - Поиск решения, если этого пункта нет, тогда необходимо выбрать кнопку Файл, в открывшемся меню щелкнуть мышью по кнопке Параметры выбрать пункт Надстройки и нажать Перейти, в открывшемся окне установить флажок Поиск решения и щелкнуть мышью по кнопке ОК. После этого повторить запуск Поиска решения;
7. в появившемся диалоговом окне Поиск решения выполнить следующие установки:
 - в окне Установить целевую ячейку: щелчком мыши по ячейке K6 установить абсолютный адрес ячейки с целевой функцией ($\$K\6);
 - установить переключатель варианта в положение значению: (используется значение по умолчанию - нуль);
 - в окне Изменяя ячейки: щелчком мыши по ячейке K5 установить абсолютный адрес ячейки со значением x ($\$K\5);
 - щелкнуть мышью по кнопке Выполнить;
 - в появившемся диалоговом окне Результаты поиска решения щёлкнуть по кнопке ОК, после чего полученное решение (уточненный корень) будет записано в изменяемую ячейку K5.

Самостоятельное занятие №4

«Практическое применение инструментов подбора параметра и поиска решения»

Цель: освоить на практике работу с инструментами для решения нелинейных уравнений.

Задание:

1. Решить заданное нелинейное уравнение согласно своему варианту методом подбора параметра. Отобразить график.
2. Решить заданное нелинейное уравнение согласно своему варианту методом поиска решения.

Практическое занятие №7

«Анализ данных в Excel»

Текст по столбцам

С помощью мастера распределения текста по столбцам текст, содержащийся в одной ячейке, можно разделить на несколько.

1. Выделите ячейку или столбец с текстом, который вы хотите разделить.
2. На вкладке Данные нажмите кнопку Текст по столбцам.
3. В мастере распределения текста по столбцам установите переключатель с разделителями и нажмите кнопку Далее.
4. Выберите разделители для своих данных. Например, запятую и пробел. Данные можно предварительно просмотреть в окне Образец разбора данных.
5. Нажмите кнопку Далее.
6. В поле Поместить в выберите место на листе, где должны отображаться разделенные данные.
7. Нажмите кнопку Готово.

Мгновенное заполнение

Функция мгновенного заполнения автоматически подставляет данные, когда обнаруживает закономерность. Например, с помощью мгновенного заполнения можно разделять имена и фамилии из одного столбца или объединять их из двух разных столбцов.

Предположим, что столбец А содержит имена, столбец В — фамилии, а вы хотите заполнить столбец С сочетаниями имен и фамилий. Если ввести полное имя в столбец С, функция мгновенного заполнения заполнит остальные ячейки соответствующим образом.

1. Введите полное имя в ячейке С2 и нажмите клавишу ВВОД.
2. Начните вводить следующее полное имя в ячейке С3. Excel определит закономерность и отобразит предварительное изображение остальной части столбца, заполненной объединенным текстом.
3. Для подтверждения предварительного просмотра нажмите клавишу ВВОД.
4. Объединение данных с помощью функции мгновенного заполнения

Если вариант заполнения не выводится, вероятно, эта функция не включена. Вы можете выбрать Данные > Мгновенное заполнение, чтобы применить заполнение вручную или нажать клавиши CTRL+E. Чтобы включить мгновенное заполнение, выберите Сервис > Параметры > Дополнительно > Параметры правки и установите флажок Автоматически выполнять мгновенное заполнение.

Консолидация данных

При выполнении ряда работ у пользователя Microsoft Excel может быть создано несколько однотипных таблиц в одном файле или в нескольких книгах.

Данные необходимо свести воедино. Собрать в один отчет, чтобы получить общее представление. С такой задачей справляется инструмент «Консолидация».

Сначала проверим, чтобы:

- макеты всех таблиц были одинаковыми;
- названия столбцов – идентичными (допускается перестановка колонок);
- нет пустых строк и столбцов.

Далее нам нужно создать новую книгу или лист, в которую будет выводиться результирующая таблица с объединенными данными. Открываем его (или ее, если это книга), после чего нажимаем левой кнопкой мыши по левой верхней ячейке (если она не была выделена сама). Важно убедиться, что под ней и справа от нее нет абсолютно никакой информации, потому что консолидированная таблица будет занимать столько места, сколько ей требуется. После этого переходим на вкладку «Данные», на которой есть кнопка с соответствующим названием. Нажимаем на «Консолидация», после чего открывается диалоговое окно, которое выглядит следующим образом.

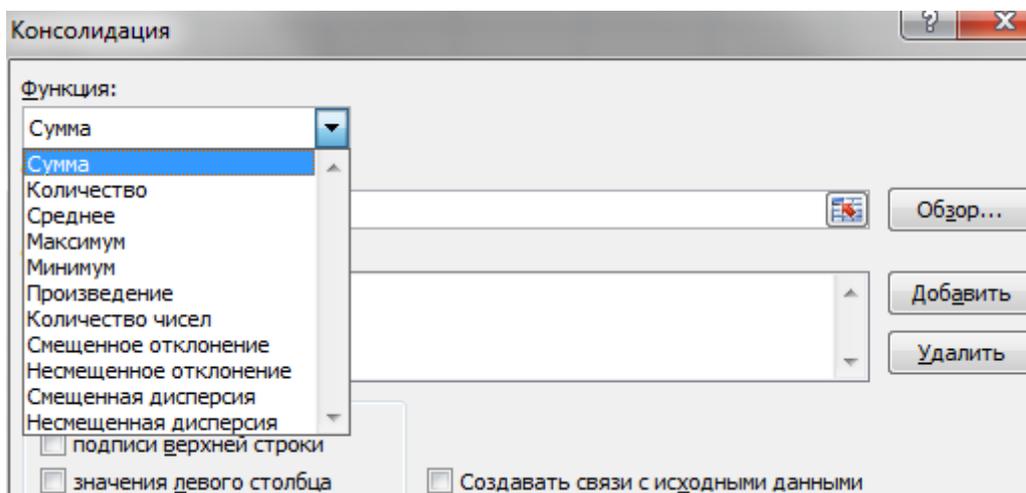


Рис. 21. Консолидация

Видим, что слева находится перечень функций, которые могут использоваться при объединении. Это самые разные вычисления, такие как сумма значений, несмещенная дисперсия, среднее арифметическое и другое. Мы выберем первый вариант, потому что он наиболее часто используемый и простой для понимания. В этом случае ячейки в соответствующих местах будут просто складываться между собой.

После этого начинаем заполнять следующее поле – «Ссылка». Для этого нужно поставить в этом поле курсор. После этого открываем первый лист (в нашем случае это данные за первый квартал). Все данные выделяем (в том числе, и шапку), после чего там появится первый диапазон, который будет суммироваться с дальнейшим. После этого нажимаем кнопку «Добавить».

Далее добавляем такое количество листов, которое требуется. В нашем случае это данные за все остальные кварталы. Просто нажимаем «Добавить», выделяем требуемые диапазоны, а потом повторяем цикл до тех пор, пока не будет добавлено такое количество, которое нам необходимо.

Все диапазоны, которые были добавлены для объединения, отображаются в соответствующем поле, возле которого есть надпись «Список диапазонов». Также можно более гибко задавать настройки, указывая флажочками возле подходящих по смыслу пунктов «Подписи верхней строки» или «Значения левого столбца». С помощью этих параметров можно определять, что использовать в качестве имен. Также есть кнопка «Создавать связи с исходными данными», которая позволяет автоматически изменять данные в консолидированной таблице, если в исходном материале что-то редактируется.

Диспетчер сценариев

Сценарий – это набор значений, которые Excel сохраняет и может автоматически подставлять на листе. Вы можете создать и сохранить различные группы значений в виде

сценариев, а затем переключаться на любой из них, чтобы просматривать различные результаты.

Если у нескольких пользователей есть определенные данные, которые вы хотите использовать в сценариях, то можно добавить их в отдельные книги и объединить сценарии из нескольких книг.

После подготовки всех нужных сценариев можно создать сводный отчет, в который включаются данные из всех сценариев.

Управление сценариями выполняется с помощью диспетчера сценариев в группе Анализ "что если" на вкладке Данные.

Предположим, вам требуется создать бюджет, но доходы точно не известны. С помощью сценариев можно определить различные возможные значения дохода, а затем переключаться между сценариями для выполнения анализа "что если".

Таблица данных

Таблица данных – это диапазон ячеек, в котором можно изменять значения в некоторых ячейках и вы можете найти разные ответы на вопрос. Экспериментировать с различными значениями, чтобы увидеть соответствующий вариант результатов, — это распространенная задача при анализе данных.

Таблицы данных с двумя переменными используются в том случае, если требуется проследить, как изменение значений двух переменных в одной формуле повлияет на результаты этой формулы. Например, таблицу данных с двумя переменными можно использовать, чтобы узнать, как разные комбинации процентных ставок и сроков ссуды повлияют на размер ежемесячного платежа.

В таблице данных с двумя переменными используется формула, содержащая два списка входных значений. Формула должна ссылаться на две разные ячейки ввода.

Исходные данные			Параметр 1				
Параметр 1	10		10	11	12	13	14
Параметр 2	250	Параметр 2	250				
Параметр 3	3000		251				
			252				
			253				
			254				

Рис. 22. Таблица данных

1. Ввод исходных данных в желтые ячейки.
2. Исходные данные могут изменяться. Диапазон значений вводится в зеленые ячейки.
3. В оранжевую ячейку вводится формула, которая ссылается на исходные данные (желтые ячейки).
4. С помощью кнопки “Таблица данных” и выделения всего диапазона (оранжевой ячейки, зеленых ячеек и голубых ячеек) происходит расчет всей таблицы.

Прогноз данных

Если у вас есть статистические данные с зависимостью от времени, вы можете создать прогноз на их основе. При этом в Excel создается новый лист с таблицей, содержащей статистические и предсказанные значения, и диаграммой, на которой они отражены. С помощью прогноза вы можете предсказывать такие показатели, как будущий объем продаж, потребность в складских запасах или потребительские тенденции.

1. На листе введите два ряда данных, которые соответствуют друг другу:
 - ряд значений даты или времени для временной шкалы;
 - ряд соответствующих значений показателя.

Эти значения будут предсказаны для дат в будущем.

2. Выделите оба ряда данных.
3. На вкладке Данные в группе Прогноз нажмите кнопку Лист прогноза.
4. В окне Создание прогноза выберите график или гограмму для визуального представления прогноза.
5. В поле Завершение прогноза выберите дату окончания, а затем нажмите кнопку Создать.

В Excel будет создан новый лист с таблицей, содержащей статистические и предсказанные значения, и диаграммой, на которой они отражены.

Этот лист будет находиться слева от листа, на котором вы ввели ряды данных (то есть перед ним).

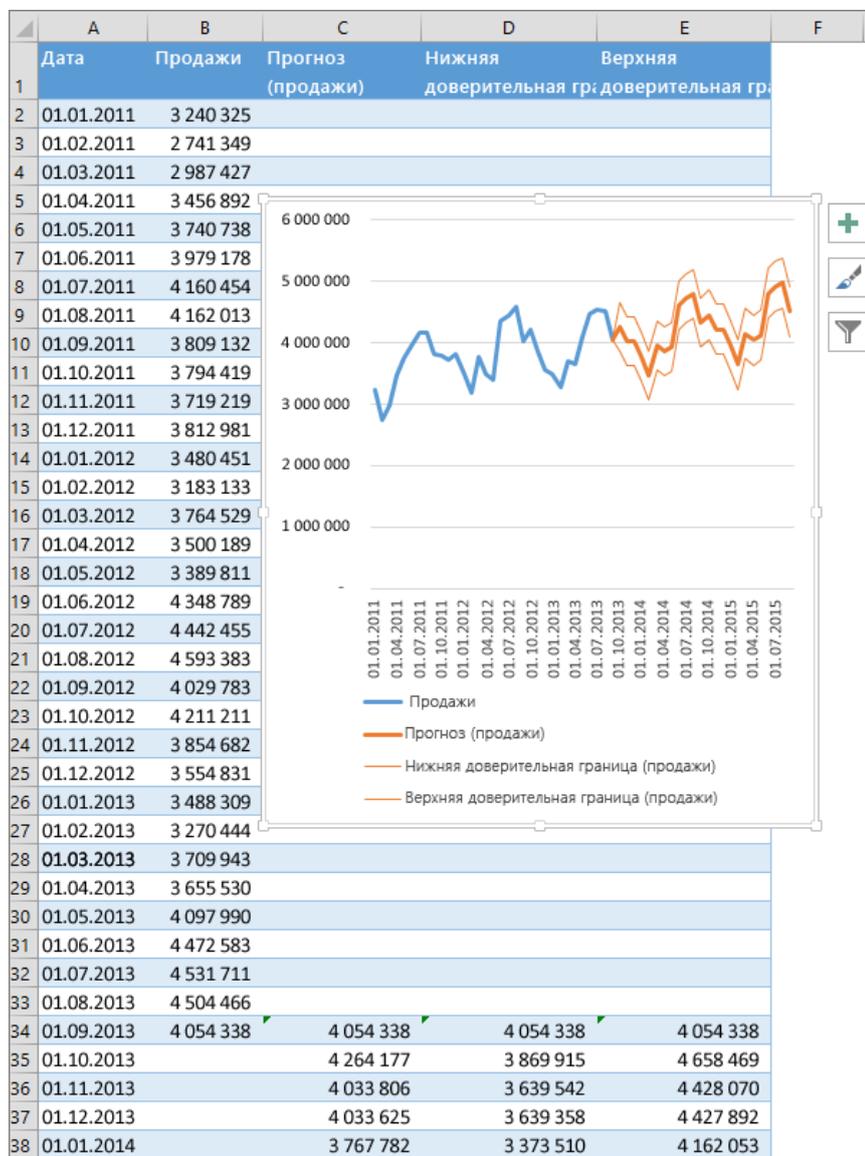


Рис. 23. Прогноз данных

Структурирование данных

Если у вас есть список данных, которые вы хотите сгруппировать и обобщить, можно создать структуру из восьми уровней. Каждый внутренний уровень, представленный более высоким числом в символы структуры, отображает детали для предыдущего внешнего уровня, представленного более низким числом в символах контура. Используйте структуру для быстрого отображения суммарных строк или столбцов или для отображения подробных данных для каждой группы. Вы можете создать структуру строк (как показано в примере ниже), структуру столбцов или структуру строк и столбцов.

Корреляция

Корреляционный анализ – это распространённый метод исследования, применяемый для определения уровня зависимости 1-й величины от 2-й. В табличном процессоре есть особый инструмент, который позволяет реализовать данный тип исследования.

Он необходим для определения зависимости между двумя разными величинами. Иными словами, происходит выявление того, в какую сторону (меньшую/большую) меняется величина в зависимости от изменений второй.

Зависимость устанавливается тогда, когда начинается выявление коэффициента корреляции. Этот метод отличается от анализа регрессии, так как здесь только один показатель, рассчитываемый при помощи корреляции. Интервал изменяется от +1 до -1. Если она плюсовая, то повышение первой величины способствует повышению 2-й. Если минусовая, то повышение 1-й величины способствует понижению 2-й. Чем выше коэффициент, тем сильнее одна величина влияет на 2-ю.

Вычисление корреляции с помощью Пакета анализа:

1. Переходим в раздел «Файл».
2. Открылось новое окошко, в котором нужно кликнуть на раздел «Параметры».
3. Жмём на «Надстройки».
4. Находим в нижней части элемент «Управление». Здесь необходимо выбрать из контекстного меню «Надстройки Excel» и кликнуть «ОК».
5. Открылось специальное окно надстроек. Ставим галочку рядом с элементом «Пакет анализа». Кликаем «ОК».
6. Активация прошла успешно. Теперь переходим в «Данные». Появился блок «Анализ», в котором необходимо кликнуть «Анализ данных».
7. В новом появившемся окошке выбираем элемент «Корреляция» и жмем на «ОК».
8. На экране появилось окошко настроек анализа. В строчку «Входной интервал» необходимо ввести диапазон абсолютно всех колонок, принимающих участие в анализе. В настройках отображения вывода изначально выставлен параметр «Новый рабочий лист», что означает показ результатов на другом листе. По желанию можно поменять локацию вывода результата. После проведения всех настроек нажимаем на «ОК».
9. Вывелись итоговые показатели.

Регрессия

Регрессионный и корреляционный анализ – статистические методы исследования. Это наиболее распространенные способы показать зависимость какого-либо параметра от одной или нескольких независимых переменных.

Регрессионный анализ показывает влияние одних значений (самостоятельных, независимых) на зависимую переменную. К примеру, как зависит количество экономически активного населения от числа предприятий, величины заработной платы и др. параметров. Или: как влияют иностранные инвестиции, цены на энергоресурсы и др. на уровень ВВП.

Регрессия бывает:

- линейной ($y = a + bx$);
- параболической ($y = a + bx + cx^2$);
- экспоненциальной ($y = a * \exp(bx)$);
- степенной ($y = a * x^b$);
- гиперболической ($y = b/x + a$);
- логарифмической ($y = b * \ln(x) + a$);
- показательной ($y = a * b^x$).

Запуск регрессионного анализа:

1. Открываем меню инструмента «Анализ данных». Выбираем «Регрессия».

2. Откроется меню для выбора входных значений и параметров вывода (где отобразить результат). В полях для исходных данных указываем диапазон описываемого параметра (Y) и влияющего на него фактора (X). Остальное можно и не заполнять.

3. После нажатия ОК, программа отобразит расчеты на новом листе (можно выбрать интервал для отображения на текущем листе или назначить вывод в новую книгу).

Как интерпретировать результаты:

- Множественный R соответствует коэффициенту корреляции R, который выражает степень зависимости независимых переменных (X) и зависимой переменной (Y). Множественный R равен квадратному корню из коэффициента детерминации, эта величина принимает значения в интервале от нуля до единицы.

- R-квадрат соответствует коэффициенту детерминации R². Данная величина служит мерой определенности, характеризует качество полученной регрессионной прямой. Это качество выражается степенью соответствия между исходными данными и регрессионной моделью (расчетными данными). Мера определенности всегда находится в пределах интервала [0;1]. В большинстве случаев значение R-квадрат находится между этими значениями, называемыми экстремальными, т.е. между нулем и единицей. Если значение R-квадрата близко к единице, это означает, что построенная модель объясняет почти всю изменчивость соответствующих переменных. И наоборот, значение R-квадрата, близкое к нулю, означает плохое качество построенной модели.

- Нормированный R-квадрат учитывает лишь значимые факторы модели, и показывает, что определенный процент общей вариации результативного признака объясняется вариацией факторных признаков. Это говорит о том, что выбранные факторы влияют на величину приобретаемых товаров и услуг, что подтверждает правильность их включения в построенную модель.

- Стандартная ошибка соответствует остаточному стандартному отклонению. Данная ошибка дает только общую оценку степени точности коэффициентов регрессии. Она позволяет получить некоторое представление о кривой функции плотности вероятности. Однако она не несет информации о том, находится ли полученная оценка в середине распределения и, следовательно, является точной или в «хвосте» распределения и, таким образом, относительно неточна. Чем больше дисперсия случайного члена, тем, очевидно, больше будет выборочная дисперсия остатков и, следовательно, существеннее стандартные ошибки коэффициентов в уравнении регрессии, что позволяет с высокой вероятностью заключить, что полученные коэффициенты неточны. Однако это всего лишь вероятность. Возможно, что в какой-то конкретной выборке воздействия случайного фактора в различных наблюдениях будут взаимно погашены и в конечном итоге коэффициенты регрессии будут точны. Проблема состоит в том, что, вообще говоря, нельзя утверждать, произойдет это или нет.

- «Наблюдения» соответствуют числу наблюдений.

Самостоятельное занятие №5

«Практическое применение инструментов анализа данных»

Цель: освоить на практике работу с инструментами для анализа данных.

Задание:

1. Мгновенное заполнение.

- Сформировать таблицу с исходными данными.

- Таблица должна состоять из 10 (минимум) строк и 5 (минимум) столбцов.

Обязательно должны быть текстовые поля, числовое поле, поле с датой и/или временем.

- Продемонстрируйте использование функции “Мгновенное заполнение” разными способами, используя все поля.

2. Таблица данных.
 - Смоделируйте ситуацию, когда вам может пригодиться таблица данных.
 - Введите исходные данные, как было показано на занятии, и заполните таблицу данными с помощью инструмента «Таблица данных».
3. Прогноз данных (если у вас версия Excel 2019).
 - Сформируйте таблицу с датами через равный интервал и числовыми данными (числовые данные можно создать через функции случайных чисел). Используйте функцию прогноза, чтобы вычислить значения на будущие периоды.
4. Корреляция и регрессия.
 - Сформируйте таблицу данных (из нескольких столбцов).
 - Важно: не используйте функции случайных чисел, необходимо, чтобы ваши данные были логически зависимы. Например, повышение запыленности помещения и показатель заболеваемости сотрудников, или снижение стоимости товара и повышение спроса на него. Для наглядности (чтобы проследить зависимость двух рядом данных) используйте графики в Excel.
 - Необходимо создать минимум три столбца данных, которые будут связаны.
 - Для определения коэффициентов корреляции, а также расчета регрессии воспользуйтесь надстройкой «Пакет анализа».
 - Прокомментируйте полученные коэффициенты корреляции и регрессии и другие статистические показатели. Для этого используйте «Рецензирование» - «Создать примечание».

Практическое занятие №8

«Сводные таблицы и диаграммы в Excel»

Сводные таблицы

Сводная таблица – это эффективный инструмент для вычисления, сведения и анализа данных, который упрощает поиск сравнений, закономерностей и тенденций.

Создание сводной таблицы

1. Выделите ячейки, на основе которых вы хотите создать сводную таблицу.
2. На вкладке Вставка нажмите кнопку Сводная таблица.
3. В разделе Выберите данные для анализа установите переключатель Выбрать таблицу или диапазон.
4. В поле Таблица или диапазон проверьте диапазон ячеек.
5. В разделе Укажите, куда следует поместить отчет сводной таблицы установите переключатель На новый лист, чтобы поместить сводную таблицу на новый лист. Можно также выбрать вариант На существующий лист, а затем указать место для отображения сводной таблицы.
6. Нажмите кнопку ОК.

Настройка сводной таблицы

1. Чтобы добавить поле в сводную таблицу, установите флажок рядом с именем поля в области Поля сводной таблицы.
2. Чтобы переместить поле из одной области в другую, перетащите его в целевую область.

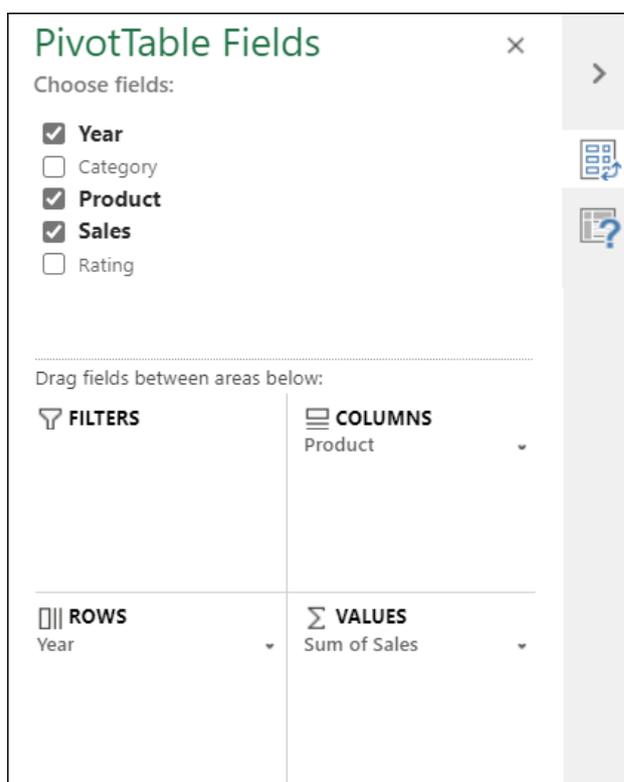


Рис. 24. Настройка сводной таблицы

Изменение исходных данных сводной таблицы

После создания сводной таблицы можно изменить диапазон исходных данных. Например, расширить его и включить дополнительные строки данных. Однако если исходные данные существенно изменены, например содержат больше или меньше столбцов, рекомендуется создать новую сводную таблицу.

1. Щелкните Отчет сводной таблицы.

2. На вкладке «Анализ» в группе «Данные» нажмите кнопку «Изменить источник данных» и выберите «Изменить источник данных».

3. Выполните одно из указанных ниже действий.

- Чтобы изменить источник данных для таблицы Excel на другую таблицу или диапазон ячеек, щелкните «Выбрать таблицу или диапазон», а затем введите первую ячейку в текстовом поле «Таблица или диапазон» и нажмите кнопку «ОК».

- Чтобы использовать другое подключение, сделайте следующее:

- Выберите «Использовать внешний источник данных» и нажмите кнопку «Выбрать подключение».

- Отобразилось диалоговое окно «Существующие подключения».

- В списке «Показать» в верхней части диалоговых окнах выберите категорию подключений, для которых нужно выбрать подключение, или выберите «Все существующие подключения» (значение по умолчанию).

- Выберите подключение в списке «Выберите подключение» и нажмите кнопку «Открыть».

- Нажмите кнопку ОК.

Упорядочение полей сводной таблицы с помощью списка полей

После создания сводной таблицы становится доступным список полей. Вы можете изменять структуру сводной таблицы путем добавления и упорядочения ее полей.

Если щелкнуть в любом месте сводной таблицы, должен появиться список полей. Если после щелчка внутри сводной таблицы список полей не отображается, откройте его, щелкнув в любом месте сводной таблицы. Затем на ленте в разделе Работа со сводными таблицами щелкните Анализ > Список полей.

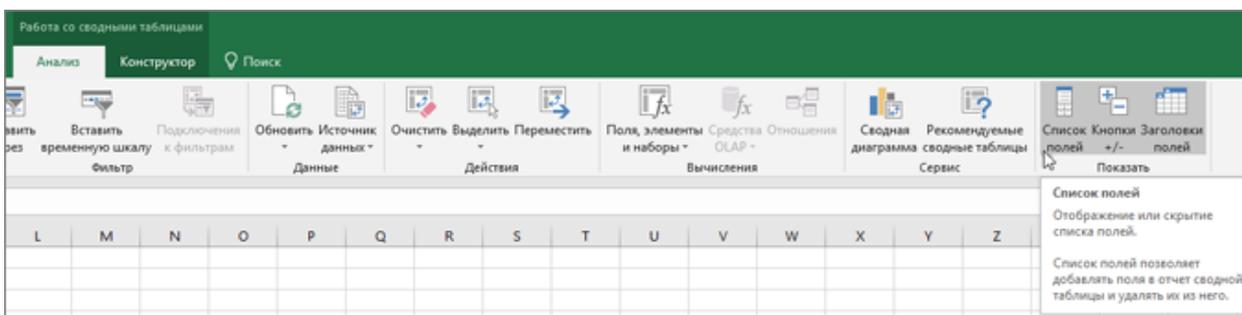


Рис. 25. Сводные таблицы, список полей

В этом списке есть раздел полей, в котором выбираются поля для сводной таблицы, и раздел областей (в нижней части), в котором поля можно упорядочить нужным образом.

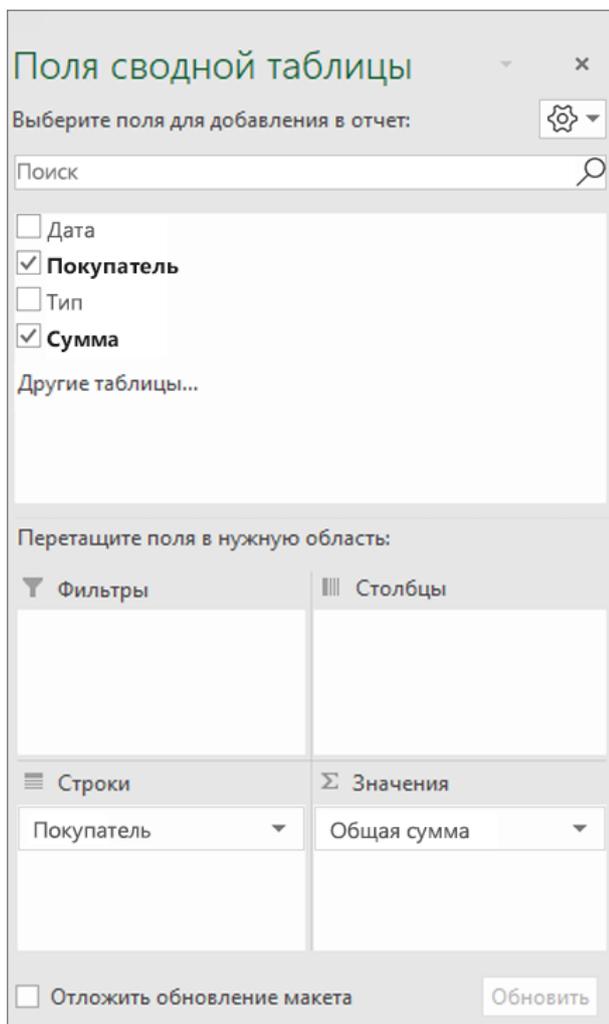


Рис. 26. Поля сводной таблицы

С помощью раздела полей в списке можно добавлять поля в сводную таблицу, устанавливая флажки рядом с именами полей, которые нужно разместить в стандартной области списка полей.

Как правило, нечисловые поля добавляются в область строк, числовые – в область значений, а иерархии даты и времени – в область столбцов.

С помощью раздела областей (в нижней части) в списке полей можно менять порядок полей, перетаскивая их между четырьмя областями.

Поля из области фильтров отображаются как фильтры отчетов верхнего уровня над сводной таблицей, как показано ниже.

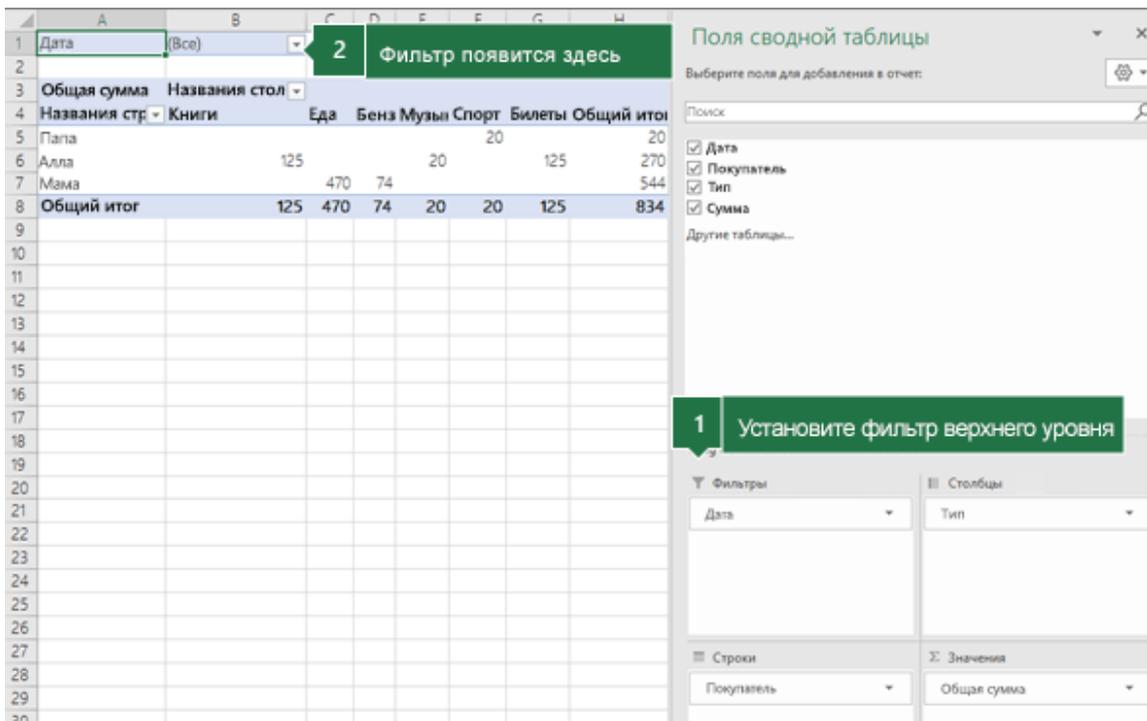


Рис. 27. Фильтры сводной таблицы

Поля из области столбцов отображаются как названия столбцов в верхней части сводной таблицы, как показано ниже.

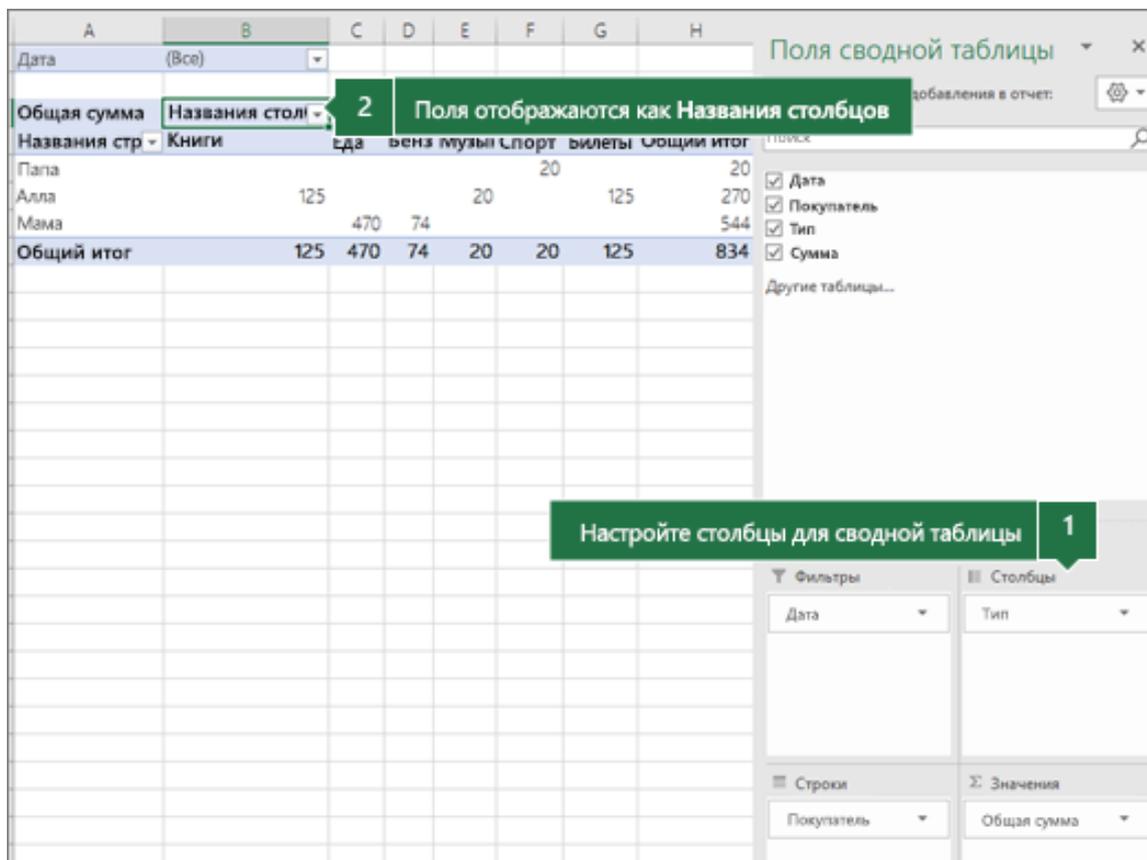


Рис. 28. Столбцы сводной таблицы

В зависимости от иерархических отношений между полями столбцы более низкого уровня могут быть вложены в столбцы более высокого.

Поля из области строк отображаются как названия строк в левой части сводной таблицы, как показано ниже.

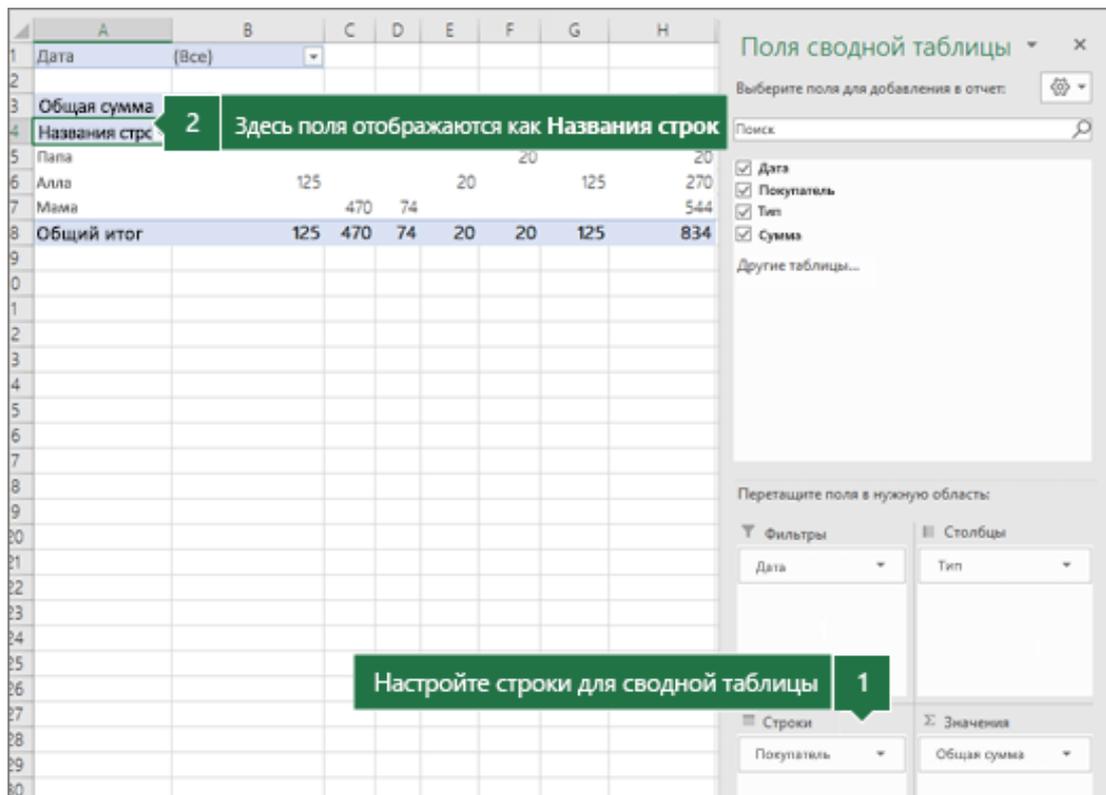


Рис. 29. Строки сводной таблицы

В зависимости от иерархических отношений между полями строки более низкого уровня могут быть вложены в строки более высокого.

Поля из области значений отображаются в сводной таблице как просуммированные числовые поля, как показано ниже.

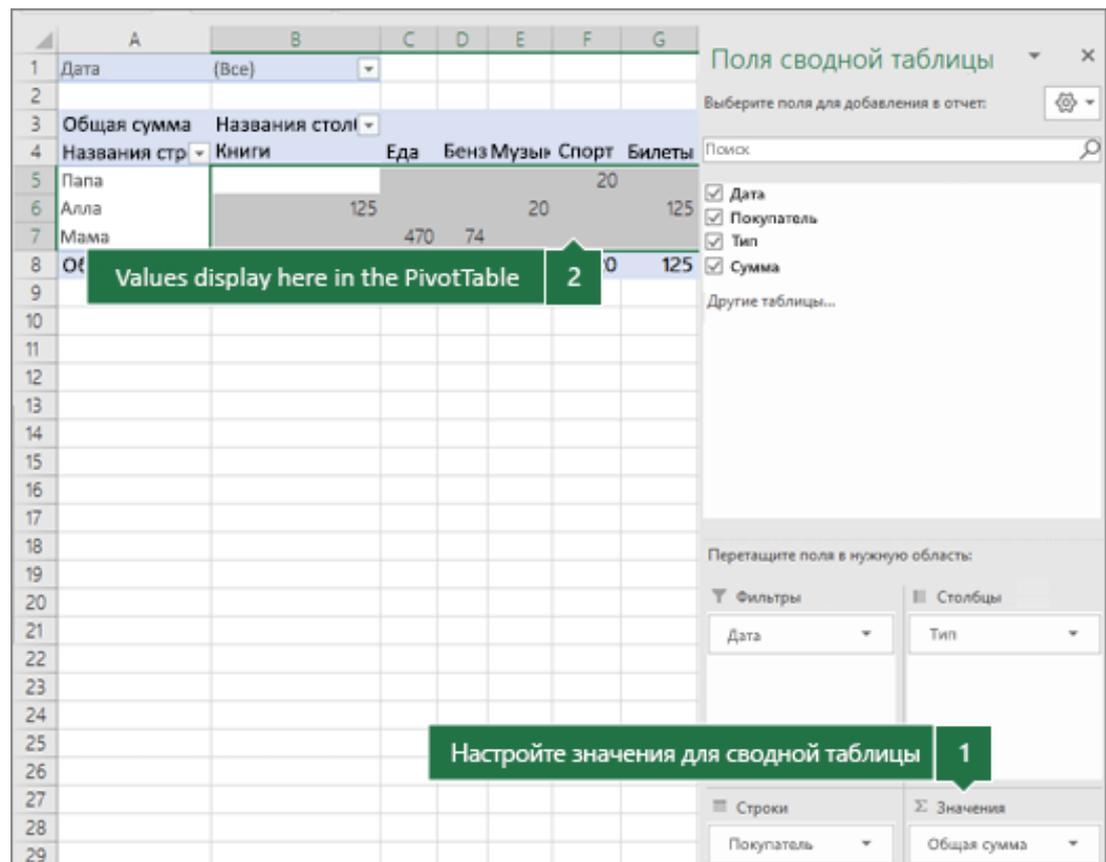


Рис. 30. Значения сводной таблицы

Если в области больше одного поля, их порядок можно изменить перетаскиванием в нужное положение. Чтобы удалить поле из сводной таблицы, перетащите его за пределы раздела областей.

Использование срезов для фильтрации данных

Срезы предоставляют кнопки, которые можно использовать для фильтрации таблиц или срезов. Помимо быстрой фильтрации срезы также указывают текущее состояние фильтрации, что позволяет легко понять, что именно отображается в данный момент.

1. Щелкните в любом месте таблицы или таблицы.
2. На вкладке «Главная» перейдите к срезу.
3. В диалоговом окне «Вставка срезов» выберите флажки для полей, которые вы хотите отобразить, а затем выберите «ОК».
4. Для каждого выбранного поля будет создан срез. Если нажать любую из кнопок среза, фильтр будет автоматически применяться к связанной таблице или таблице.
5. Вы можете настроить параметры среза на вкладке «Срез» (в более новых версиях Excel) или на вкладке «Конструктор» (в Excel 2016 и более старых версиях) на ленте.
6. Если вы хотите подключить срез к одной или нескольким срезов, перейдите в «Подключения к отчетам» проверьте срез, который нужно включить в срез, и выберите «ОК».

Создание временной шкалы сводной таблицы для фильтрации дат

Вместо настройки фильтров для отображения дат вы можете воспользоваться временной шкалой сводной таблицы. Это параметр динамического фильтра, позволяющий легко фильтровать по дате или времени и переходить к нужному периоду с помощью ползунка. Чтобы добавить эту шкалу на лист, на вкладке Анализ нажмите кнопку Вставить временную шкалу.



Рис. 31. Временная шкала

Как и срез для фильтрации данных, временную шкалу можно добавить один раз и затем использовать в любой момент для изменения диапазона времени сводной таблицы.

1. Щелкните в любом месте сводной таблицы, чтобы показать группу Работа со сводными таблицами на ленте, и на вкладке Анализ нажмите кнопку Вставить временную шкалу.

2. В диалоговом окне Вставка временных шкал установите флажки рядом с нужными полями дат и нажмите кнопку ОК.

Использование временной шкалы для фильтрации по периоду времени

Вставив временную шкалу, вы можете фильтровать данные по периоду времени на одном из четырех уровней (годы, кварталы, месяцы или дни).

1. Нажмите на стрелку рядом с отображаемым временным уровнем и выберите нужный вариант.

2. Перетащите полосу прокрутки временной шкалы к периоду времени, который вы хотите проанализировать.

3. В элементе управления отрезком времени нажмите на плитку периода времени и перетащите ее, чтобы добавить дополнительные плитки для выбора нужного диапазона дат. С помощью маркеров отрезка времени отрегулируйте диапазон дат с обеих сторон.

Сводные диаграммы

Иногда сложно представить обобщаемую картину, когда необработанные данные еще не обобщены. Первый из них может быть в том, чтобы создать таблицу, но не все могут взглянуть на числа в таблице и быстро увидеть, что происходит. Сводные диаграммы — отличный способ наглядного предоставления данных.

Создание сводной диаграммы

1. Выделите ячейку в таблице.
2. Выберите элементы Вставка > Сводная диаграмма
3. Нажмите кнопку ОК.

Создание диаграммы на основе сводной таблицы

1. Выделите ячейку в таблице.
2. Выберите элементы Работа со сводными таблицами > Анализ > Сводная диаграмма
3. Выберите диаграмму.
4. Нажмите кнопку ОК.

	A	B	C
1	МЕСЯЦ	КАТЕГОРИЯ	СУММА
2	Январь	Транспорт	740,00 ₽
3	Январь	Бакалея	2 350,00 ₽
4	Январь	Хозтовары	1 750,00 ₽
5	Январь	Развлечения	1 000,00 ₽
6	Февраль	Транспорт	1 150,00 ₽
7	Февраль	Бакалея	2 400,00 ₽
8	Февраль	Хозтовары	2 250,00 ₽
9	Февраль	Развлечения	1 250,00 ₽
10	Март	Транспорт	900,00 ₽
11	Март	Бакалея	2 600,00 ₽
12	Март	Хозтовары	2 000,00 ₽
13	Март	Развлечения	1 200,00 ₽

Рис. 32. Данные для сводной диаграммы

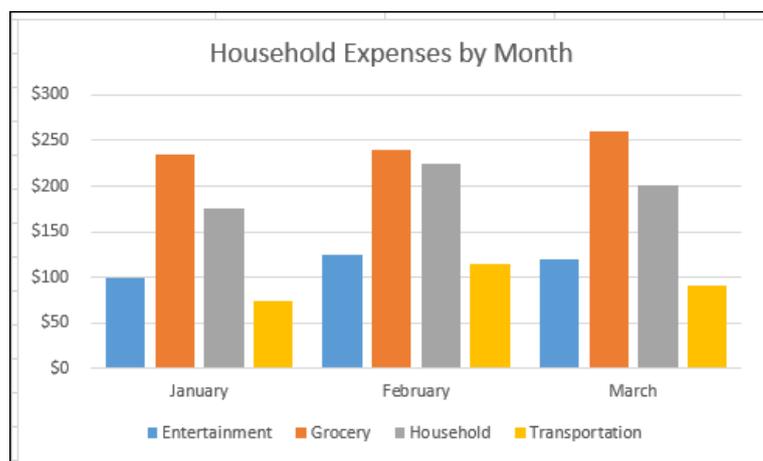


Рис. 33. Сводная диаграмма

Самостоятельное занятие №6

«Практическое применение инструментов сводных таблиц и диаграмм»

Цель: освоить на практике работу с инструментами сводных таблиц и диаграмм.

Задание:

1. Создать сводную таблицу.
- Сформировать таблицу с исходными данными. Требования: таблица должна

содержать не менее 50 строк, столбцы - текстовые (2-3), числовые (1-2), дата (1).

- Чтобы генерировать ФИО, города, страны, профессии, воспользуйтесь ссылкой: «<https://ciox.ru/class-text>»

- Чтобы генерировать числовые значения или даты, используйте формулу СЛУЧМЕЖДУ.

- Создайте сводную таблицу, в правом углу конструктора таблицы установите значения для строк и столбцов, установите значения для заполнения таблицы (сумма, среднее или др.), задайте фильтр.

- Форматируйте таблицу, смените цветовое отображение, установите условное форматирование (гистограмму, цветовую шкалу или значки).

- Используйте срез или временную шкалу.

2. Создать сводную диаграмму.

- По созданной в задании 1 сводной таблице создайте сводную диаграмму. Скорректируйте ее отображение и форматирование на свой вкус.

- Цель: в наглядном отображении ваших данных.

Практическое занятие №9

«Макросы и введение в программирование»

Макросы

Для автоматизации повторяющихся задач в Microsoft Excel можно быстро записать макрос. Предположим, у вас есть даты в различном формате и вы хотите применить ко всем из них один формат. Это можно сделать с помощью макроса. Вы можете записать макрос, который применяет нужный формат, а затем запускать его при необходимости.

При записи макроса все действия в записи макроса записываются Visual Basic для приложений (VBA) коде. Эти действия могут включать ввод текста или чисел, выбор ячеек или команд на ленте или в меню, форматирование ячеек, строк или столбцов или даже импорт данных из внешнего источника, скажем, Microsoft Access. Visual Basic Приложение (VBA) – это подмножество мощного Visual Basic программирования, которое входит в большинство Office приложений. Хотя VBA позволяет автоматизировать процессы как в приложениях, так и между Office, необязательно знать код VBA или программирование на компьютере, если оно делает то, что вам нужно.

Важно знать, что при записи макроса регистрируются почти все ваши действия. Поэтому если вы допустите ошибку, например нажмете не ту кнопку, средство записи макросов зарегистрирует это действие. В таком случае можно снова записать всю последовательность или изменить код VBA. Поэтому перед записью процесса следует хорошо проработать его. Чем точнее вы запишете последовательность, тем более эффективно будет работать макрос.

Макросы и средства VBA находятся на вкладке Разработчик, которая по умолчанию скрыта, поэтому сначала нужно включить ее. Дополнительные сведения см. в статье Отображение вкладки «Разработчик».

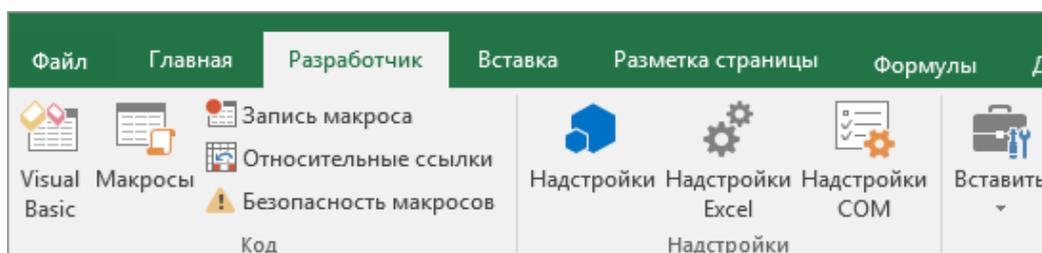


Рис. 34. Панель «Разработчик»

Запись макроса

Перед записью макросов полезно знать следующее:

- Макрос, записанный для работы с диапазоном Excel, будет выполняться только для ячеек этого диапазона. Поэтому если вы добавите в диапазон новую строку, макрос не будет применяться к ней.

- Если вам нужно записать длинную последовательность задач, советуем вместо этого использовать несколько более мелких макросов.

- В макросе могут содержаться и задачи, не относящиеся к Excel. Процесс макроса может охватывать прочие приложения Office и другие программы, которые поддерживают Visual Basic для приложений (VBA). Например, вы можете записать макрос, который сначала обновляет таблицу в Excel, а затем открывает Outlook для ее отправки по электронной почте.

Чтобы записать макрос, следуйте инструкциям ниже.

1. На вкладке Разработчик в группе Код нажмите кнопку Запись макроса.
2. В поле Имя макроса введите название макроса. Сделайте имя понятным, чтобы можно было быстро найти нужный макрос.
3. Чтобы назначить сочетание клавиш для запуска макроса, в поле Сочетание клавиш введите любую строчную или прописную букву. Рекомендуется использовать сочетания клавиш с CTRL+SHIFT, так как они будут заменять собой совпадающие с ними стандартные сочетания клавиш в Excel, пока открыта книга, содержащая макрос. Например, если назначить сочетание клавиш CTRL+Z (Отменить), вы не сможете использовать его для функции "Отменить" в данном экземпляре Excel.
4. В списке Сохранить в выберите книгу, в которой вы хотите сохранить макрос.
5. В поле Описание при необходимости введите краткое описание действий макроса.
6. Чтобы начать запись макроса, нажмите кнопку ОК.
7. Выполните действия, которые нужно записать.
8. На вкладке Разработчик в группе Код нажмите кнопку Остановить запись.

Работа с макросами, записанными в Excel

На вкладке Разработчик щелкните Макросы, чтобы просмотреть макросы, связанные с книгой. Кроме того, можно нажать клавиши ALT+F8. При этом откроется диалоговое окно Макрос.

С помощью редактора Visual Basic (VBE) вы можете добавлять в записанный код собственные переменные, управляющие структуры и другие элементы, которые не поддерживает средство записи макросов. Так как средство записи макросов фиксирует почти каждый шаг, выполняемый во время записи, может также потребоваться удалить ненужный код. Просмотр записанного кода — отличный способ научиться программировать на VBA или отточить свои навыки.

Введение в программирование

Есть несколько основных причин использовать программирование на VBA в Office.

- Автоматизация и повторяемость

Если имеется изменение, которое нужно внести более десяти или двадцати раз, возможно, стоит реализовать его с помощью VBA. Если изменение нужно внести несколько сотен раз, определенно следует рассмотреть VBA. Почти все изменения форматирования или редакторские правки, которые можно внести вручную, можно реализовать на VBA.

- Расширения возможностей взаимодействия с пользователем

Иногда некоторые действия пользователей с документом или приложением Office обязательны или желательны, но не доступны в стандартном приложении. Например, нужно уведомлять пользователей о необходимости что-то сделать при открытии, сохранении или печати документа.

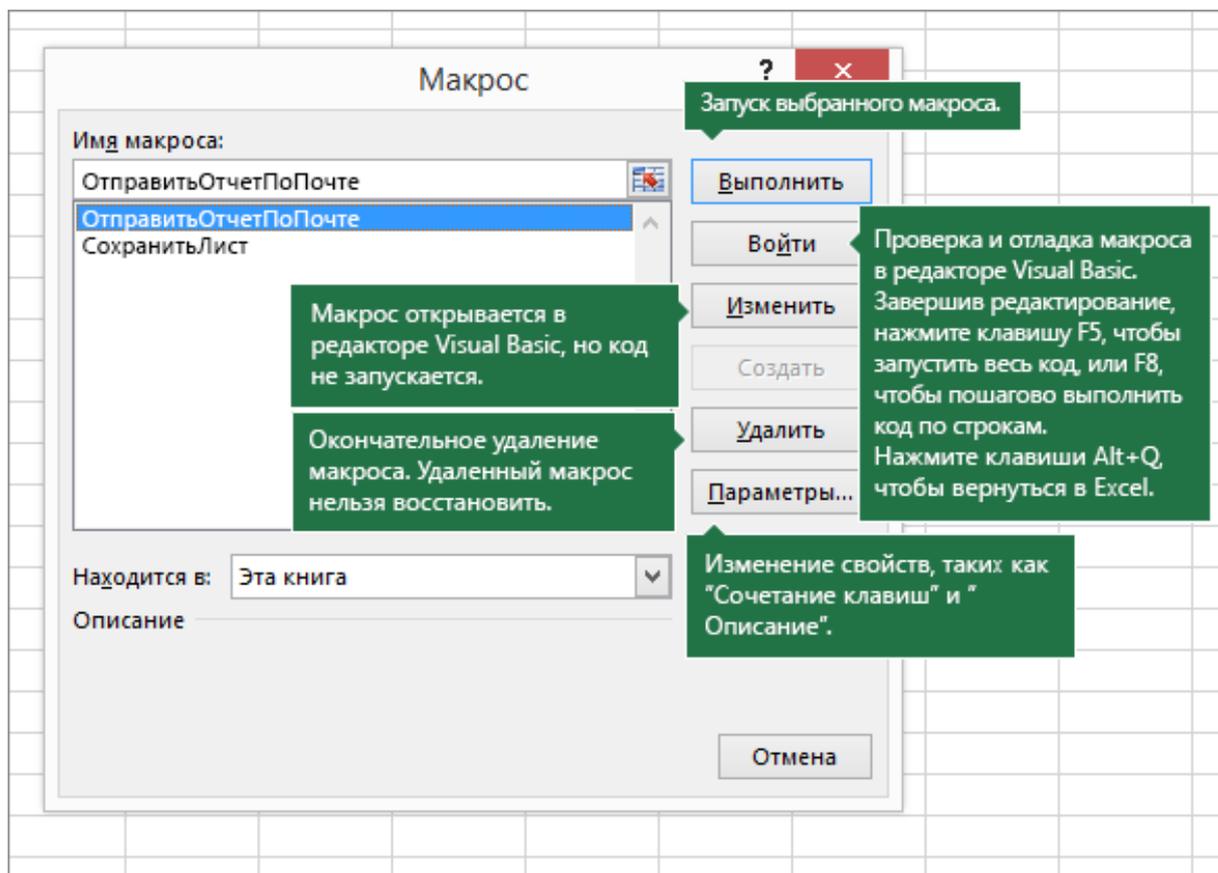


Рис. 35. Работа с макросами

Использование кода для выполнения операций приложениями

Написание кода может казаться очень сложным и непонятным процессом. На самом деле, его базовые принципы основаны на применении повседневной логики и вполне доступны. Объекты в приложениях Microsoft Office, созданные для получения инструкций, можно сравнить с кнопками телефона. Когда вы нажимаете кнопку, телефон распознает команду и включает соответствующую цифру в набираемую последовательность. При программировании вы взаимодействуете с приложением, отправляя инструкции различным объектам. Эти объекты эффективны, но у них есть свои ограничения. Они смогут делать только то, для чего были разработаны, и выполнять только ваши инструкции.

Если нажать кнопку Макрос на вкладке Разработчик, открывается диалоговое окно Макрос, которое предоставляет доступ к подпрограммам или макросам VBA, которые можно использовать из определенного документа или приложения. Кнопка Visual Basic открывает редактор Visual Basic, где можно создавать и редактировать код VBA.

На вкладке Разработчик в Excel также есть кнопка Запись макроса, которая автоматически создает код VBA, позволяющий воспроизводить ваши действия в приложении. Запись макроса — это великолепное средство, которое можно использовать для изучения VBA. Читая такой код, можно понять язык VBA и объединить знания пользователя и программиста Office. Только помните о том, что этот код может быть запутанным, так как редактор макросов делает допущения насчет ваших намерений, которые не всегда точны.

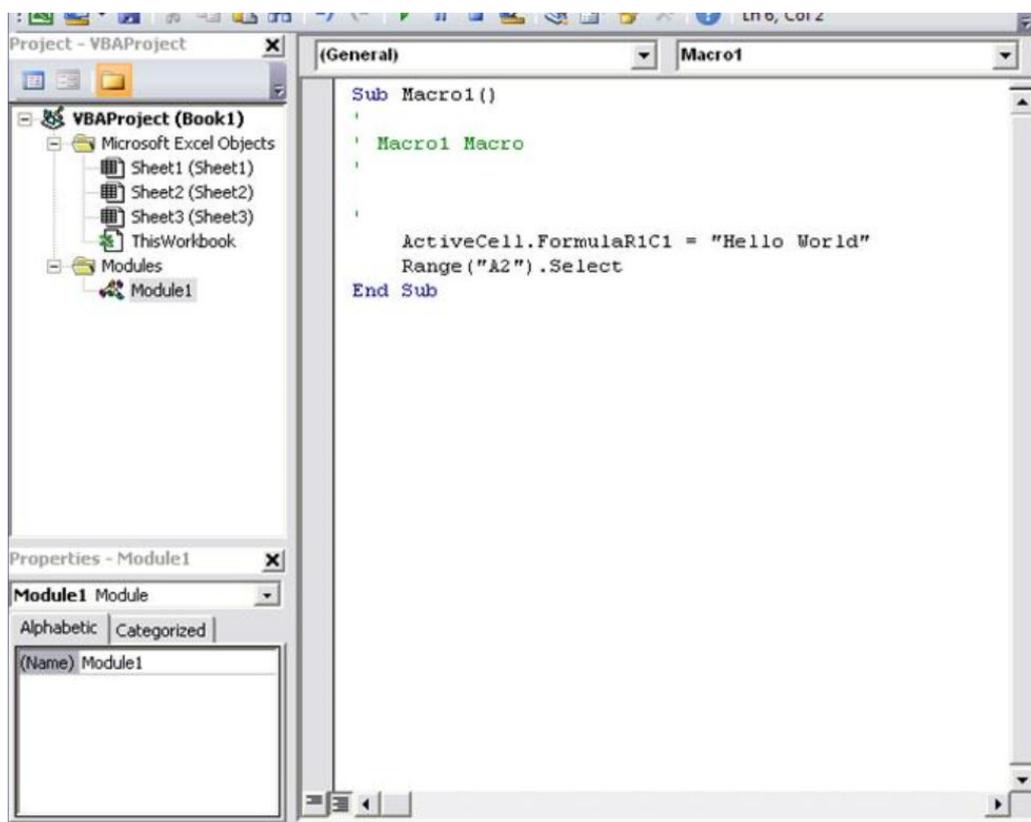


Рис. 36. Окно редактирования кода VBA

Созданный макрос должен выглядеть указанным ниже образом.

```
Sub Macro1 ()
'
' Macro1 Macro
'
'
Range ("B1").Select
ActiveCell.FormulaR1C1 = "Hello World"
Range ("B2").Select
End Sub
```

Учтите схожие моменты с предыдущим примером кода, в котором выбирается ячейка A1, и отличия от него. В этом примере кода выбирается ячейка B1, а затем строка "Hello World" записывается в активную ячейку. Кавычки вокруг текста обозначают строковое значение (в противоположность числовому значению).

В объектной модели есть несколько способов адресации любого объекта (ячейки A1 в этом примере), что может вызывать затруднения.

Иногда самый лучший способ изучить программирование — вносить небольшие изменения в определенный рабочий код и смотреть, что получается. Попробуйте сделать это сейчас. Откройте Макрос1 в редакторе Visual Basic и измените код указанным ниже образом.

```
Sub Macro1 ()
Worksheets (1) .Range ("A1") .Value = "Wow!"
Worksheets (1) .Range ("A1") .Borders .LineStyle = xlDouble
End Sub
```

Обращение к ячейкам

Свойство `Worksheet.Cells` возвращает объект `Range`, который представляет все ячейки на таблице (а не только те ячейки, которые в настоящее время используются).

В этом примере размер шрифта для ячейки C5 на листе1 активной книги составляет 14.

```
Worksheets("Sheet1").Cells(5, 3).Font.Size = 14
```

В этом примере очищается формула в ячейке 1 на листе1 активной книги.

```
Worksheets("Sheet1").Cells(1).ClearContents
```

В этом примере задают размер шрифта и вид шрифта для каждой ячейки на листе1
Arial.

```
With Worksheets("Sheet1").Cells.Font  
    .Name = "Arial"  
    .Size = 8  
End With
```

Далее приведены дополнительные полезные выражения, для установления свойств ячеек.

```
Range("A1").Borders.Color = RGB(0, 0, 0)  
Range("A1").Font.Name = "Times New Roman"  
Range("A1").Font.Color = RGB(255, 0, 0)  
Range("A1").Font.Bold = True  
Range("A1").HorizontalAlignment = xlCenter  
Range("A1").Interior.Color = RGB(230, 230, 230)  
Range("A1").Value = Range("C5").Row  
Range("A1").Value = Range("C5").Column  
Range("A2") = "Привет всем!"  
Range("A3") = Range("A2").Text  
Range("A1").Activate  
Range("A1:D1").Clear  
  
Range("A2").NumberFormat = "# ##0.00"  
Range("A3").NumberFormat = "dd.mm.yyyy"  
Range("A2") = 2362.4568  
Range("A3") = CDate("01.01.2021")  
  
Worksheets(1).Visible = False  
Worksheets("Лист1").Activate  
MsgBox (Worksheets.Count)
```

Переменные

При объявлении переменных обычно используется оператор Dim. Оператор объявления может быть помещен внутрь процедуры для создания переменной на уровне процедуры. Или он может быть помещен в начале модуля в разделе объявлений, чтобы создать переменную на уровне модуля.

В примере ниже создается переменная и указывается тип данных «String».

```
Dim strName As String
```

Если данный оператор отображается в процедуре, переменная strName может использоваться только в этой процедуре. Если же оператор отображается в разделе объявлений модуля, переменная strName доступна для всех процедур данного модуля, но недоступна для процедур других модулей проекта.

Переменные могут быть объявлены одним из следующих типов данных: Boolean, Byte, Integer, Long, Currency, Single, Double, Date, String (для строк переменной длины), String * length (для строк фиксированной длины), Object или Variant. Если тип данных не указан, по умолчанию присваивается тип Variant. Вы также можете создать определяемый пользователем тип с помощью оператора Type.

Вы можете объявить несколько переменных в одном операторе. Чтобы указать тип данных, необходимо задать тип данных для каждой переменной.

В приведенном ниже операторе переменные `intX`, `intY` и `intZ` объявлены типом `Integer`.

```
Dim intX As Integer, intY As Integer, intZ As Integer
```

В приведенном ниже операторе `intX` и `intY` объявлены как `Variant` и только `intZ` объявлен как тип `Integer`.

```
Dim intX, intY, intZ As Integer
```

Нет необходимости указывать тип данных переменной в операторе объявления. Если вы не укажете тип данных, переменной будет присвоен тип `Variant`.

Сокращение для объявления переменных `x` и `y` типом `Integer` в приведенном выше операторе.

```
Dim intX%, intY%, intZ as Integer
```

Оператор `Public` используется для объявления общих переменных на уровне модуля.

```
Public strName As String
```

Общие переменные могут использоваться в любой процедуре проекта. Если общая переменная объявлена в стандартном модуле или модуле класса, она может использоваться в любых проектах, содержащих ссылку на проект, в котором объявлена данная общая переменная.

Оператор `Private` используется для объявления частных переменных на уровне модуля.

```
Private MyName As String
```

Частные переменные могут использоваться только процедурами одного модуля.

Если для объявления переменной в процедуре используется оператор `Static` вместо оператора `Dim`, объявленная переменная сохраняет свое значение между вызовами в этой процедуре.

Уровни доступа

Уровень доступа объявленного элемента – это степень возможности доступа к нему, то есть код, имеющий разрешение на чтение или запись в него. Это определяется не только способом объявления самого элемента, но и уровнем доступа контейнера элемента. Код, который не может получить доступ к содержащему элементу, не может получить доступ к каким-либо содержащимся в нем элементам, даже объявленным `Public` как `Public`. Например, к `Public` переменной в `Private` структуре можно обращаться из класса, который содержит структуру, но не за пределами этого класса.

Ключевое слово `Public` в операторе объявления указывает, что доступ к элементу можно получить из кода в любом месте того же проекта, из других проектов, которые ссылаются на проект, и из любой сборки, построенной из проекта.

Ключевое слово `protected` в операторе объявления указывает, что доступ к элементу может осуществляться только в том же классе или из класса, производного от этого класса.

Ключевое слово `Friend` в операторе объявления указывает, что к элементу можно получить доступ из той же сборки, но не за пределы сборки.

Можно использовать `Friend` только на уровне модуля, интерфейса или пространства имен. Это означает, что можно объявить дружественный элемент на уровне исходного файла или пространства имен либо внутри интерфейса, модуля, класса или структуры, но не в процедуре.

Сочетание ключевого слова Protected Friend в операторе объявления указывает, что доступ к элементу можно получить либо из производных классов, либо из той же сборки или в обоих случаях.

Ключевое слово Private в операторе объявления указывает, что доступ к элементу может осуществляться только в том же модуле, классе или структуре.

Private можно использовать только на уровне модуля. Это означает, что можно объявить закрытый элемент внутри модуля, класса или структуры, но не на уровне исходного файла или пространства имен, внутри интерфейса или в процедуре.

На уровне модуля Dim без каких-либо ключевых слов уровня доступа эквивалентна Private объявлению. Однако может потребоваться использовать Private ключевое слово, чтобы облегчить чтение и интерпретацию кода.

Сочетание ключевого слова Private protected в операторе объявления указывает, что доступ к элементу возможен только в пределах того же класса, а также из производных классов, найденных в той же сборке, что и содержащий его класс.

Массивы

Массивы объявляются так же, как и другие переменные, при помощи операторов Dim, Static, Private или Public. Отличие скалярных переменных (которые не являются массивами) от переменных массивов заключается в том, что для массива, как правило, необходимо указывать размер. Массив с указанным размером является массивом фиксированного размера. Массив, размер которого можно изменить во время выполнения программы, является динамическим массивом.

Индексация массива от 0 или 1 зависит от оператора Option Base. Если не указано Option Base 1, все индексы массива будут начинаться с нуля.

В приведенном ниже примере кода массив фиксированного размера объявлен массивом целых переменных (Integer) с 11 строками и 11 столбцами:

```
Dim MyArray(10, 10) As Integer
```

Первый аргумент определяет количество строк, второй — столбцов.

Объявив динамический массив, вы сможете менять его размер во время выполнения кода. Используйте операторы Static, Dim, Private или Public, чтобы объявить массив, не указывая значение в скобках, как показано в следующем примере:

```
Dim sngArray() As Single
```

В процедуре внутри области массива используйте оператор ReDim, чтобы изменить количество измерений, задать количество элементов и определить нижнюю и верхнюю границы каждого измерения. Вы можете менять динамический массив при помощи оператора ReDim в любое время. Однако значения внутри массива при этом не сохраняются.

```
ReDim sngArray(10)
```

Ключевые слова

Ключевые слова являются зарезервированными, то есть их нельзя использовать в качестве имен программных элементов, таких как переменные или процедуры. Это ограничение можно обойти, заключив имя в квадратные скобки ([]).

Построение диаграмм

Объект Chart (Excel) представляет диаграмму в книге. Диаграмма может представлять собой внедренную диаграмму (содержащуюся в объекте ChartObject) или отдельный лист диаграммы.

Коллекция Charts содержит объект Chart для каждого листа диаграммы в книге. Чтобы вернуть один объект Chart, используйте синтаксис Charts (индекс), где индекс — это номер индекса или имя листа диаграммы.

Номер индекса диаграммы представляет положение листа диаграммы на панели вкладок книги. Charts(1) — это первая (крайняя левая) диаграмма в книге; Charts(Charts.Count) — последняя (самая правая).

Все листы диаграмм включаются в число индексов, даже если они скрыты. Имя листа диаграммы отображается на вкладке книги для диаграммы. Можно использовать свойство Name объекта ChartObject, чтобы установить или вернуть имя диаграммы.

Действия по добавлению диаграммы и ее форматированию показаны в коде далее.

```
Dim My_Chart As Chart
Set My_Chart = ThisWorkbook.Charts.Add
With My_Chart
    .SetSourceData (Sheets("Лист1").Range("A1:A5"))
    .ChartType = xlColumnClustered
    .HasTitle = True
    .ChartTitle.Caption = "Hello"
    .SeriesCollection(1).Interior.Color = RGB(255, 0, 0)
    .SeriesCollection(1).Points(3).Interior.Color = RGB(255, 0, 0)
    .HasLegend = False
    .Location xlLocationAsObject, "Лист1"
End With
```

Самостоятельное занятие №7

«Применение инструмента записи макросов для решения задач»

Цель: освоить на практике работу по написанию макросов.

Задание:

1. Напишите макрос, который:

- создает таблицу из 10 строк и 4 столбцов разных форматов значений,
- наполняет ее (вручную и с использованием функций),
- выполняет расчеты итоговых значений (СУММ, СРЗНАЧ, МИН, МАКС или др.),
- форматирует таблицу (шрифт, цвет, заливка, границы и т.д.),
- создает диаграмму и форматирует ее.

Практическое занятие №10

«Программирование линейных вычислительных процессов в VBA»

Линейный вычислительный процесс (ЛВП) представляет собой набор операторов, выполняемых последовательно, один за другим. Основу программы ЛВП составляют операторы присваивания, ввода и вывода данных.

Оператор присваивания используется для сохранения результата вычисления арифметического выражения в переменной. Он имеет следующий общий вид:

```
переменная = выражение
```

Оператор присваивания работает следующим образом: сначала вычисляется выражение, стоящее справа от знака :=, затем результат записывается в переменную, стоящую слева от знака. Например, после выполнения оператора

```
k = k + 2
```

текущее значение переменной k увеличится на 2.

В процессе программирования линейных вычислительных процессов можно воспользоваться рядом встроенных функций, либо доступных при обращении к определенному модулю.

Язык VBA предоставляет обширный набор встроенных (стандартных) функций, используемых для упрощения создания программ. Существуют математические функции, функции для обработки строк, для работы с датами и временем, для финансовых расчетов.

- `abs` (число) - абсолютное значение числа;
- `atn` (число) - арктангенс значения, заданного в радианах;
- `cos` (число) - косинус угла, заданного в радианах;
- `exp` (число) - это e в степени x , где x - заданное число;
- `fix` (число) - целая часть числа;
- `int` (число) - округленное до целой части число. В случае отрицательного параметра возвращает ближайшее меньшее отрицательное число;
- `log` (число) - натуральный логарифм числа;
- `rnd` - случайное число в интервале от 0 до 1. Перед вызовом функции `Rnd` используйте инструкцию `Randomize` без аргумента для инициализации генератора случайных чисел значением, возвращаемым системным таймером;
- `round` (Числовое_Выражение, Число) - округленное Числовое Выражение с точностью до Число десятичных знаков;
- `sgn` (число) - знак числа; +1 - если параметр положителен, -1 - если параметр отрицателен, 0 - если параметр равен 0;
- `sin` (число) - синус угла, заданного в радианах;
- `sqrt` (число) - квадратный корень;
- `tan` (число) - тангенс угла, заданного в радианах.

Ряд функций доступен из модуля `Math`:

- `Math.Abs`(число) - абсолютное значение числа;
- `Math.Atn`(число) - арктангенс числа;
- `Math.Cos`(число) - косинус числа;
- `Math.Exp`(число) - e в степени x , где x - число;
- `Math.Log`(число) - натуральный логарифм числа;
- `Math.Randomize`(число) - `Randomize` использует число для инициализации генератора случайных чисел функции `Rnd`, задавая ему новое начальное значение. Если опустить число, то в качестве нового начального значения будет использоваться значение, возвращаемое системным таймером.
- `Math.Rnd`(число) - возвращает значение, содержащее псевдослучайное число;
- `Math.Round`(число, число 2) - округляет число до количества знаков после запятой, равного числу 2;
- `Math.Sgn`(число) - знак числа; если +1 - положительное, если 0 - ноль, если -1 - отрицательное;
- `Math.Sin`(число) - синус числа;
- `Math.Sqrt`(число) - корень числа;
- `Math.Tan`(число) - тангенс числа.

Ряд функций, доступных при работе на листе, также доступны при обращении к модулю `WorksheetFunction`. Среди них:

- `asin`(число) - арксинус числа;
- `acos`(число) - арккосинус числа;
- `atan`(число) - арктангенс числа;
- `degrees`(число) - перевод числа в градусы;
- `radians`(число) - перевод числа в радианы;
- `pi`() - число пи;
- `Ln`(число) - натуральный логарифм числа;
- `log`(число1б число2) - логарифм числа1 по основанию числа2;

- log10(число) - десятичный логарифм;
- average(число1, число2, число3...) - среднее среди чисел;
- max(число1, число2, число3...) - максимальное среди чисел;
- min(число1, число2, число3...) - минимальное среди чисел;
- count(число1, число2, число3...) - подсчет количества чисел;
- randBetween(число1, число2) - случайное число между двумя заданными;
- sum(число1, число2, число3...) - сумма чисел;
- even(число) - ближайшее большее четное число;
- odd(число) - ближайшее большее нечетное число;
- round(число1, число2) - округляет число до количества знаков после запятой, равного числу 2.

Практическое занятие №11

«Программирование разветвляющегося вычислительного процесса в VBA»

Условный оператор используется для представления разветвляющегося вычислительного процесса в коде программы на языке Visual Basic, т.е. он применяется в тех случаях, когда из двух возможных действий нужно выбрать одно. Выбор определяется в зависимости от выполнения некоторых условий.

В VB имеются две формы условных операторов – однострочная и блочная. Однострочная форма используется в простейших случаях, когда в каждой ветви имеется не более одного оператора, и имеет вид:

```
If <условие> Then <оператор> [Else <оператор>]
```

Действие этого оператора состоит в следующем. Вначале проверяется условие. Если оно удовлетворяется, то выполняется оператор, стоящий после слова Then, в противном случае – оператор, после слова Else. Условие задается выражением булевского типа, т.е. результат его вычисления принимает значение True или False.

Простую однострочную форму рекомендуется использовать для коротких, простых проверок. Однако блочная форма обычно проще для чтения, обработки и отладки.

Допускается также использование блоковой формы синтаксиса:

```
If условие Then
[инструкции]
[Else
[инструкции_else]]
End If
```

Расширенная модификация имеет вид:

```
If <Условие1> Then
<Операторы1>
ElseIf <Условие2> Then
<Операторы2>
...
[Else
<ОператорыN>]
End If
```

Многоточие после конструкции ElseIf обозначает, что эта конструкция может повторяться несколько раз.

В расширенной модификации операторы, стоящие после слова Else, выполняются только в том случае, если не удовлетворяется ни одно условие.

Практическое занятие №12

«Программирование циклов в VBA»

В Excel мы тратим много времени на повторяющиеся простые задачи. Например, как: форматирование нескольких диапазонов, отображение нескольких листов, копирование и вставка в несколько рабочих книг, применение фильтров к нескольким таблицам или сводным таблицам, замена значений, обновление формул и т.д.

Мы можем использовать циклы в наших макросах VBA, чтобы очень быстро повторять действия. Задачи, выполнение которых вручную может занять несколько часов, могут быть выполнены за несколько секунд с помощью цикла.

Различают циклы с предусловием и постусловием. В цикле с предусловием в начале цикла ставится условие, следовательно, код в теле цикла не выполнится, если не будет удовлетворено условие (значит, минимальное количество выполнений этого цикла - 0). В цикле с постусловием условие проверяется в конце, уже после выполнения кода в теле цикла (значит, минимальное количество выполнений этого цикла - 1).

Цикл While

Пока условие будет действительным, инструкции в цикле будут продолжать выполняться (но будьте осторожны, чтобы не сделать бесконечный цикл).

```
Sub while_loop()  
    While [условие]  
        'Инструкции  
    Wend  
End Sub
```

Цикл Do Loop

Другой способ написать цикл, который работает так же, как While Wend (выполняется так долго, пока действительно определенное условие, содержащееся в While команде):

```
Sub do_while_loop()  
    Do While [условие]  
        'Инструкции  
    Loop  
End Sub
```

В следующем варианте показано, что условие может быть прописано в конце цикла Do Loop, что означает, что инструкции будут точно выполнены по меньшей мере один раз:

```
Sub do_while_loop()  
    Do  
        'Инструкции  
    Loop While [условие]  
End Sub
```

Вместо того, чтобы повторять цикл пока выполняется условие, можно выйти из цикла по условию, если мы заменим While на Until:

```
Sub do_while_loop()  
    Do Until [условие]  
        'Инструкции  
    Loop  
End Sub
```

Цикл For

Следующий вид цикла - это цикл со счетчиком. При каждом повторении этого цикла, переменная *i* автоматически увеличивается на 1:

```

Sub for_loop()
    For i = 1 To 5
        'Инструкции
    Next
End Sub

```

Ранний выход из цикла

Возможно выйти из цикла For заранее, воспользовавшись следующей инструкцией:

```
Exit For 'Выход из цикла For
```

Давайте рассмотрим пример, который наглядно покажет:

```

Sub for_loop ()
    Dim max_loops As Integer
    max_loops = Range ("A1") 'В A1: мы указали лимит
количества повторений

    For i = 1 To 7 'Ожидаемое количество циклов: 7
        If i > max_loops Then 'Если A1 является пустой или
содержит число < 7, уменьшить количество повторений
            Exit For 'Если условие ПРАВДА, выход из цикла
For loop
        End If

        MsgBox i
    Next

End Sub

```

Другие операторы Exit :

```
Exit Do 'Выход из цикла Do Loop
```

```
Exit Sub 'Выход из процедуры
```

```
Exit Function 'Выход из функции
```

Самостоятельное занятие №8

«Алгоритмы линейных вычислений, ветвления и циклов»

Цель: освоить на практике работу алгоритмов линейных вычислений, ветвления и циклов в VBA.

Задание:

1. Линейные вычислительные процессы.
 - Напишите две процедуры, которые будут решать задачи на математические формулы из практики №2.
 - Процедуры должны быть помещены в отдельный модуль.
 - Исходные данные, по которым будут выполняться расчеты, должны читаться с листа. Результат должен быть выведен на лист.
 - Добавьте форматирование ячейки с результатом расчета.
 - Усовершенствуйте процедуры так, чтобы ввод данных осуществлялся через диалоговые окна. Результат также должен выдаваться в окне с сообщением.
 - В итоге у вас должны получиться 4 процедуры (2 из первого задания и 2 из второго).

2. Разветвляющиеся вычислительные процессы.

• Напишите две процедуры, которые будут решать задачи на логические функции из практики №3.

• Процедуры должны быть помещены в отдельный модуль.
• Исходные данные, по которым будут выполняться расчеты, должны читаться с листа (одно значение X, или по одному значению - X и A). Результат должен быть выведен на лист.

• Добавьте форматирование ячейки с результатом расчета.
• Усовершенствуйте процедуры так, чтобы ввод данных осуществлялся через диалоговые окна. Результат также должен выдаваться в окне с сообщением.

• В итоге у вас должны получиться 4 процедуры (2 из первого задания и 2 из второго).

3. Циклические вычислительные процессы.

• Решить задание согласно своему варианту по журналу. Используйте циклы.

• Придумать собственные примеры использования циклов:

– с предусловием;

– с постусловием;

– со счетчиком;

и реализовать их.

Практическое занятие №13

«Разработка форм (рабочих окон) в VBA»

В Excel VBA есть очень полезный инструмент, который называется UserForm. Это практичный способ получения информации от пользователя. UserForm похож на окно мастера или диалоговое окно, в которое мы просим пользователей вводить данные, и в нем есть некоторые инструкции.

Чтобы использовать пользовательскую форму VBA, сначала нужно активировать вкладку разработчика на вкладке «Параметры». Как только вкладка разработчика активирована, мы можем перейти в Excel VBA и создать пользовательские формы. Пользовательские формы используются для хранения данных в Excel.

Чтобы добавить UserForm, нужно сделать то же самое, когда мы добавляем новый модуль:

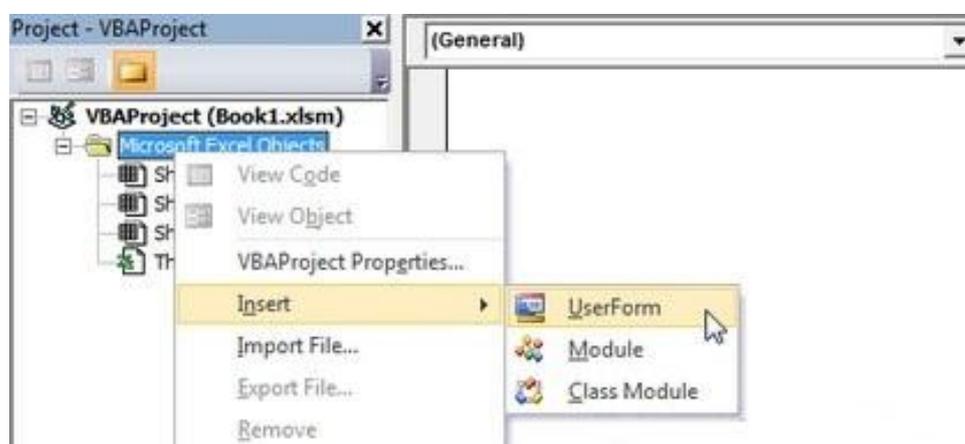


Рис. 37 Добавление UserForm

Перед вами откроется окно следующего вида:

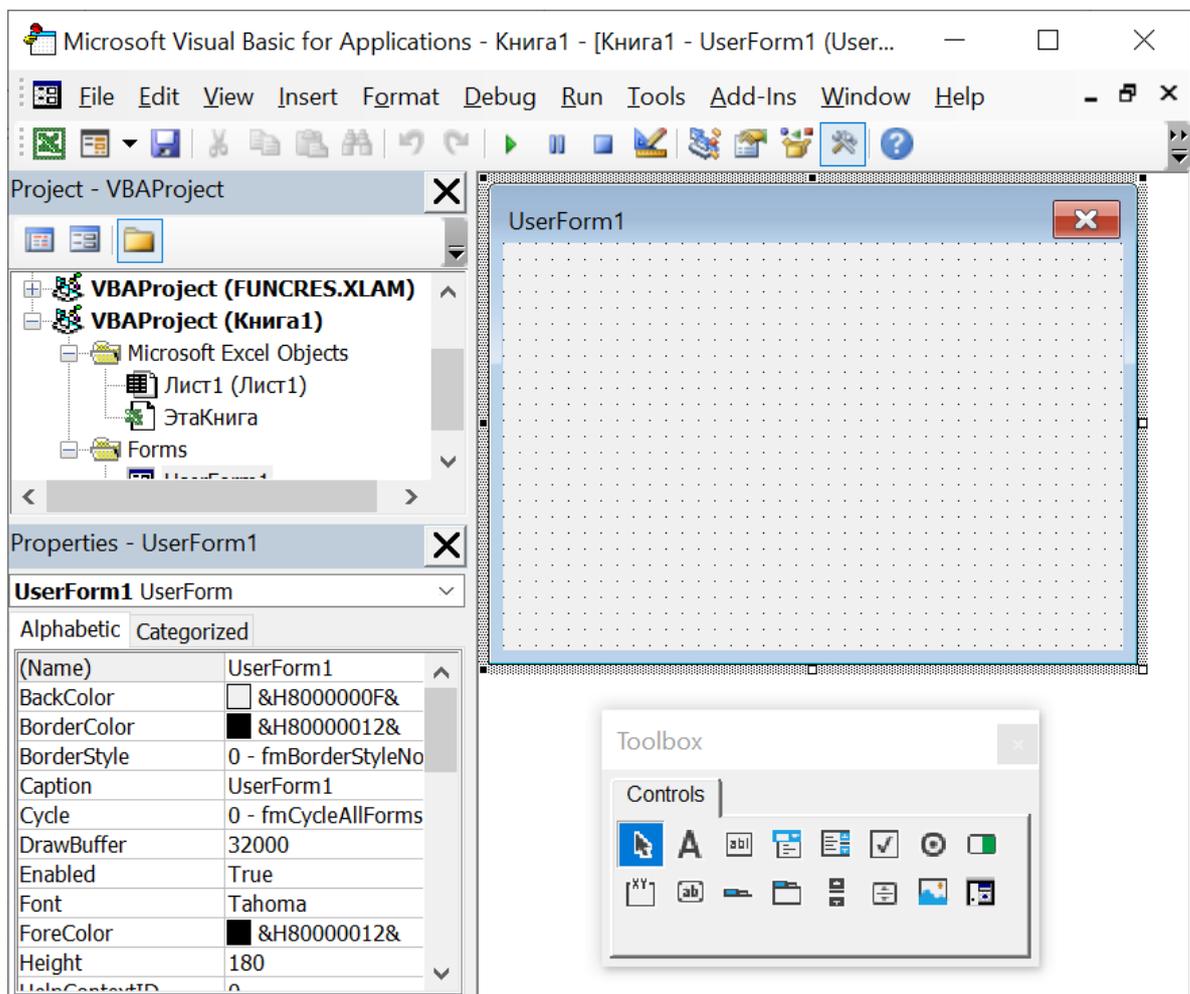


Рис. 38. Область работы с объектом UserForm

Вверху расположено меню с инструментами, слева вверху - окно с иерархией расположения файлов проекта, слева внизу - панель свойств объектов, а рядом с созданной формой - панель инструментов Toolbox.

Нажав на форму один раз (и тем самым активировав ее), вам становятся доступны свойства формы. Среди них:

- Caption - Заголовок формы;
- Height - Высота формы;
- Width - Ширина формы;
- Top - Отступ в пикселях от верхней границы экрана;
- Left - Отступ в пикселях от левой границы экрана;
- BackColor - Фон формы;
- BorderColor - Цвет границы формы;
- Font - Шрифт элементов на форме;
- Enabled - Состояние формы (активна или нет);
- Picture - Картинка на фоне формы;
- ScrollBars - Наличие полос прокрутки.

Нажав на форму 2 раза, вы можете перейти к области кода. Прямо над области кода есть два выпадающих списка: в одном содержатся форма и все элементы, добавленные на нее, а во втором - список событий. Выбрав из списка №1 элемент "UserForm", а из списка №2 - событие Initialize, можно указать в коде, что будет происходить при инициализации (открытии) формы.

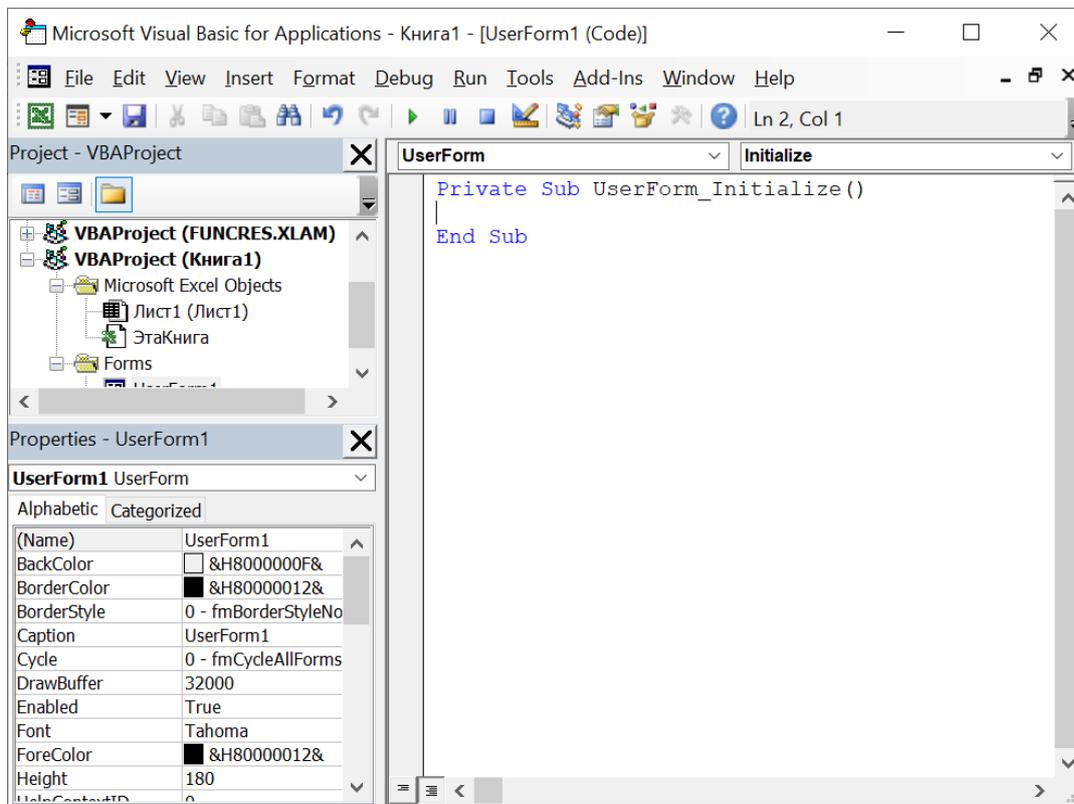


Рис. 39. Окно редактирования кода VBA

Существует и ряд других событий. Вот часть из них:

- Activate - Действия при активации формы;
- Deactivate - Действия при деактивации формы;
- Click - Действия при нажатии на форму;
- DblClick - Действия при двойном нажатии на форму;
- KeyPress - Действия при нажатии клавиш на клавиатуре;
- MouseDown - Действия при нажатии (но не отпускании) кнопки мыши;
- MouseUp - Действия при отпускании (после нажатия) кнопки мыши;
- MouseMove - Действия при движении курсора мыши;
- Scroll - Действия при использовании колеса прокрутки мыши;
- Resize - Действия при изменении размера окна, и т.д.

Важно знать, что обратиться к свойствам формы (и любого элемента управления на форме) можно как из окна Свойства, так и из кода.

Например, можно изменить цвет формы, указав конкретный цвет в палетке, нажав на свойство BackColor панели Свойств. А можно указать цвет с помощью кода, который будет выполняться при открытии (инициализации) формы:

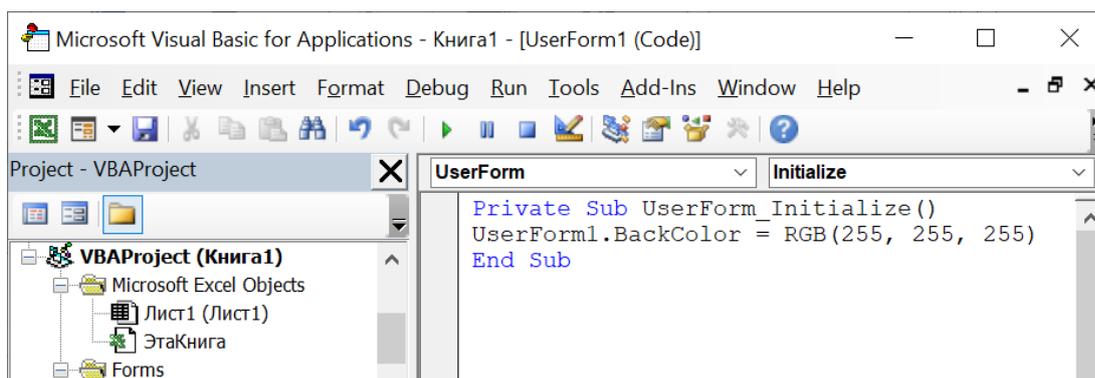


Рис. 40. Код события инициализации формы

На форме можно разместить ряд элементов. Среди них:

- Label - Метка с текстом, надпись;
- TextBox - Поле для ввода информации;
- ComboBox - Выпадающий список с информацией;
- ListBox - Список с перечнем позиций;
- CheckBox - Поле для отметки “галочкой” (или “флажком”);
- OptionButton - Переключатель;
- ToggleButton - Кнопка с фиксированным состоянием (нажата или не нажата);
- Frame - Граница для расположения элементов и группировки;
- CommandButton - Обычная кнопка;
- TabStrip - Диспетчер, содержащий коллекцию вкладок;
- MultiPage - Диспетчер, содержащий коллекцию страниц;
- ScrollBar - Полоса прокрутки;
- SpinButton - Двухнаправленный счетчик;
- Image - Объект изображения;
- RefEdit - Редактор ссылок.

Элемент управления CheckBox предоставляет пользователю возможность выбрать один из двух вариантов: включить или выключить. Если галочка установлена, то CheckBox имеет значение True, в противном случае – False.

ComboBox подобен объекту ListBox. Отличие заключается в том, что ComboBox представляет раскрывающийся список, в котором в определенный момент времени отображается только одно значение. Кроме того, пользователю в поле списка разрешено вводить значение, которое необязательно представляет одну из опций объекта ComboBox.

Каждое создаваемое диалоговое окно будет иметь как минимум один элемент управления CommandButton. Обычно используются кнопки ОК и Отмена.

Элемент управления Frame применяется в качестве оболочки для других элементов управления. Он добавляется в диалоговое окно либо в целях эстетики, либо из соображений логического группирования однотипных элементов управления. Элемент управления Frame требуется в случае, если в диалоговом окне содержится более одного набора элементов управления OptionButton.

Элемент управления Image используется для представления графического изображения, которое сохранено в отдельном файле или вставляется из буфера обмена. Графическое изображение сохраняется вместе с рабочей книгой. Таким образом, вместе с рабочей книгой передавать другому пользователю копию графического файла необязательно. Некоторые графические файлы занимают много места, поэтому их включение в рабочую книгу приведет к радикальному увеличению ее размера.

Элемент управления Label отображает текст в диалоговом окне.

Элемент управления ListBox предоставляет список опций, из которого пользователь может выбрать один вариант (или несколько). Вы вправе указать диапазон на листе, который содержит элементы списка. Этот диапазон может состоять из нескольких столбцов. Кроме того, элемент управления ListBox может заполняться с помощью кода VBA.

Элемент управления MultiPage позволяет создавать диалоговые окна с несколькими вкладками, которые подобны появляющимся после выбора команды Формат ячеек. По умолчанию элемент управления MultiPage состоит из двух вкладок.

Элемент управления OptionButton применяется при выборе пользователем одного варианта из нескольких. Эти элементы управления всегда группируются в диалоговом окне в наборы, содержащие не менее двух опций. Когда один элемент управления OptionButton выбран, все остальные элементы управления OptionButton текущей группы автоматически становятся неактивными. Если в пользовательском диалоговом окне содержится более одного набора элементов управления OptionButton, то каждый из таких

наборов должен иметь собственное значение свойства `GroupName`. В противном случае все элементы управления `OptionButton` в диалоговом окне рассматриваются как члены одной группы. Также можно вставить элементы управления `OptionButton` в объект `Frame`, что приведет к их автоматическому группированию.

Элемент управления `RefEdit` используется тогда, когда пользователь должен выделить диапазон ячеек на листе.

Элемент управления `ScrollBar` в некотором смысле подобен элементу управления `SpinButton`. Разница заключается в том, что пользователь может перетаскивать ползунок объекта `ScrollBar` для изменения значения с большим приращением. Элемент управления `ScrollBar` рекомендуется использовать при выборе значения из большого диапазона.

Элемент управления `SpinButton` позволяет выбрать значение после щелчка на одной из двух кнопок со стрелками. Одна из них применяется для увеличения значения, а вторая – для уменьшения. Элемент управления `SpinButton` часто используется совместно с элементами управления `TextBox` и `Label`, которые содержат текущее значение элемента управления `SpinButton`.

Элемент управления `TabStrip` подобен элементу управления `MultiPage`, однако использовать его сложнее. Элемент управления `TabStrip`, в отличие от `MultiPage`, не выступает контейнером для других объектов. Как правило, элемент управления `MultiPage` обладает более широкими возможностями.

Элемент управления `TextBox` позволяет пользователям вводить текст в диалоговом окне.

Элемент управления `ToggleButton` имеет два состояния: включен и выключен. Щелчок на кнопке приводит к изменению состояния на противоположное и к изменению внешнего вида кнопки. Этот элемент управления может иметь значение `True` (активен) или `False` (неактивен). Он не относится к «стандартным» элементам управления, поэтому использование двух элементов управления `OptionButton` или одного `CheckBox` может оказаться более удачным вариантом.

Окно Свойств

В VBA окно `Properties` позволяет изменять свойства выделенного элемента управления (это может быть обычный элемент управления или сама форма `UserForm`).

В окне `Properties` есть две вкладки. На вкладке `Alphabetic` свойства выбранного объекта отображаются в алфавитном порядке. На вкладке `Categorized` эти свойства сгруппированы по категориям. Обе вкладки отображают одни и те же свойства.

Для того чтобы изменить свойство, необходимо щелкнуть на нем и ввести новое значение. Некоторые свойства могут принимать только ограниченный набор допустимых значений, выбираемых из соответствующего списка. После щелчка на таком свойстве в окне `Properties` будет отображена кнопка со стрелкой, указывающей вниз. Щелкните на этой кнопке, чтобы выбрать значение из предложенного списка. Например, свойство `TextAlign` может принимать одно из следующих значений: 1 - `fmTextAlignLeft`, 2 - `fmTextAlignCenter` и 3 - `fmTextAlignRight`.

После выделения отдельных свойств (например, `Font` и `Picture`) рядом с ними отображается небольшая кнопка с троеточием. Щелчок на этой кнопке приводит к вызову диалогового окна настройки свойства.

Для свойства `Picture` элемента управления `Image` необходимо указать графический файл или вставить изображение из буфера обмена. В последнем случае следует сначала скопировать его в буфер обмена, а затем выбрать свойство `Picture` элемента управления `Image` и нажать комбинацию клавиш `<Ctrl+V>` для вставки содержимого буфера обмена. Если выделить два или более элементов управления одновременно, в окне `Properties` отобразятся только те свойства, которые являются общими для этих объектов.

Объекту `UserForm` присущ ряд свойств, значения которых можно изменять. Эти свойства применяются в качестве значений, заданных по умолчанию, для элементов управления, которые добавляются в пользовательские диалоговые окна. Например, если

изменить свойство Font пользовательского диалогового окна, все добавленные в окно элементы управления будут применять этот шрифт.

Общие свойства

Каждый элемент управления имеет как собственный набор уникальных свойств, так и ряд общих свойств, присущих другим элементам управления. Например, все элементы управления имеют свойство Name и свойства, определяющие его размер и расположение на форме (Height, Width, Left и Right). Если вы собираетесь работать с элементом управления с помощью кода VBA, присвойте ему значащее имя. Например, первый элемент управления OptionButton, который добавлен в пользовательское диалоговое окно, по умолчанию получит имя OptionButton1. В коде ссылка на этот объект будет выглядеть следующим образом: OptionButton1.Value = True. Но если элементу управления OptionButton присвоить описательное имя (например, obLandscape), то можно использовать такой оператор: obLandscape.Value = True.

Многие пользователи предпочитают имена, которые указывают на тип объекта. В предыдущем примере был использован префикс ob, который указывает на то, что объект является элементом управления OptionButton.

Можно изменять значения свойств нескольких элементов управления одновременно. Например, вы вправе создать на форме несколько элементов управления OptionButton и выровнять их по левому краю. Для этого достаточно выделить все элементы управления OptionButton и изменить значение свойства Left в окне Properties. Все выделенные элементы управления примут новое значение свойства Left.

Чтобы получить доступ к справочной информации о свойствах различных элементов управления, щелкните на свойстве в окне Properties и нажмите клавишу <F1>.

Отображение окна

Для того чтобы отобразить пользовательское диалоговое окно с помощью VBA, необходимо создать процедуру, которая вызывает метод Show объекта UserForm. Форму UserForm невозможно отобразить, не выполнив как минимум одну строку кода VBA:

```
UserForm1.Show
```

Данная процедура должна располагаться в стандартном модуле VBA, а не в модуле формы UserForm. При отображении пользовательская форма остается на экране до тех пор, пока ее не скроют. Обычно в нее добавляют элемент управления CommandButton, который запускает процедуру закрытия формы. Эта процедура либо выгружает пользовательскую форму с помощью метода Unload, либо удаляет ее с экрана с помощью метода Hide объекта UserForm.

Пример разработки собственного приложения

Предположим, нам необходимо составить окно для добавления записей о студентах, сдавших экзамен по информатике.

Была создана форма, на которую были помещены элементы Label, TextBox, ComboBox, CheckBox и CommandButton.

В окно будет вводиться информация о ФИО студента, форма обучения (очная, заочная), отмечаться факт сдачи экзамена, и если сдача состоялась, то нужно будет ввести оценку.

После этого нужно будет нажать кнопку “Добавить”, чтобы информация была помещена на лист.

В программе необходимо предусмотреть несколько аспектов:

- в поля “Фамилия”, “Имя”, “Отчество” нужно вводить ограниченное количество символов;
- в выпадающем списке “Форма об.” изначально должны быть два значения: “очная” и “заочная”;
- в поле “Оценка” нельзя вводить ничего, кроме цифр;

- поле “Оценка” и метка с текстом “Оценка” не должны быть видны, если не отмечен факт сдачи экзамена в CheckBox.

Кроме того, при добавлении первой записи на листе должна появляться шапка таблицы. Запуск всей программы должен осуществляться через кнопку на листе.

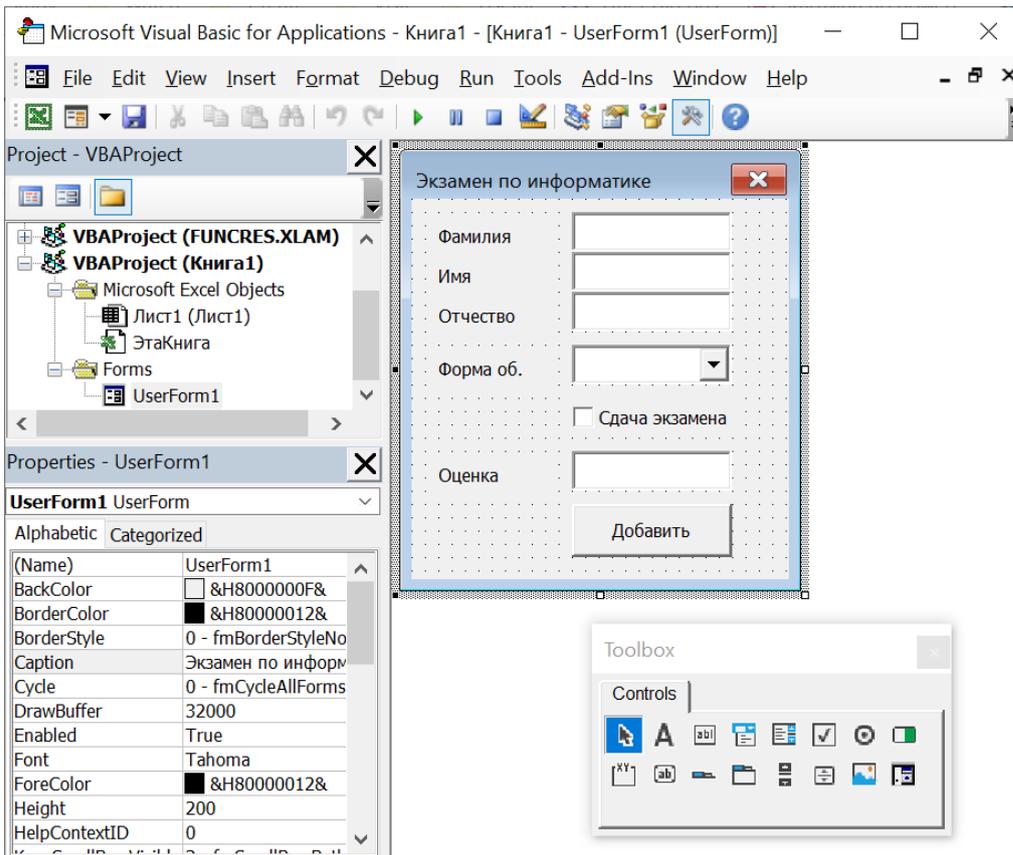


Рис. 41. Пример разработки проекта

Решение

Для начала ограничим количество символов в полях «Фамилия», «Имя», «Отчество», изменив свойство MaxLength.

Чтобы добавить элементы в выпадающий список, напишем следующий код, который будет выполняться при инициализации формы:

```
Private Sub UserForm_Initialize()
    ComboBox1.AddItem ("Очная")
    ComboBox1.AddItem ("Заочная")
End Sub
```

Чтобы в поле «Оценка» нельзя было ввести ничего, кроме цифр, необходимо прописать это ограничение на событии нажатия клавиш в элементе TextBox - KeyPress:

```
Private Sub TextBox4_KeyPress
    (ByVal KeyAscii As MSForms.ReturnInteger)
    Select Case KeyAscii
        Case 48 To 57
        Case Else
            KeyAscii = 0
    End Select
End Sub
```

В данной процедуре Ascii-коды цифр - от 48 до 57.

Теперь добавим событие, при котором поле с оценкой и подпись “Оценка” будут появляться только тогда, когда отмечено поле CheckBox.

По умолчанию устанавливаем для этих элементов свойство Visible, равное False. Затем добавляем на событие изменения CheckBox следующий код:

```
Private Sub CheckBox1_Change()  
If CheckBox1.Value = True Then  
    TextBox4.Visible = True  
    Label5.Visible = True  
Else  
    TextBox4.Visible = False  
    Label5.Visible = False  
End If  
End Sub
```

Теперь добавим событие на нажатие кнопки “Добавить”. Чтобы вести контроль количества добавленных записей, нужно объявить глобальную переменную n в коде UserForm.

```
Public n As Integer
```

В процедуре инициализации формы нужно указать, что n равно 0 (т.е. обнулить значение при запуске программы).

Затем добавить следующий код на кнопку CommandButton для добавления записей на лист.

```
Private Sub CommandButton1_Click()  
If TextBox1.Text <> "" And TextBox2.Text <> "" And  
TextBox3.Text <> "" And TextBox4.Text <> "" And ComboBox1.Text  
<> "" Then  
    If n = 0 Then  
        n = n + 1  
        Cells(1, 1).Value = "Фамилия"  
        Cells(1, 2).Value = "Имя"  
        Cells(1, 3).Value = "Отчество"  
        Cells(1, 4).Value = "Форма обучения"  
        Cells(1, 5).Value = "Оценка"  
  
        Cells(n + 1, 1).Value = TextBox1.Value  
        Cells(n + 1, 2).Value = TextBox2.Value  
        Cells(n + 1, 3).Value = TextBox3.Value  
        Cells(n + 1, 4).Value = ComboBox1.Value  
        Cells(n + 1, 5).Value = TextBox4.Value  
    Else  
        n = n + 1  
        Cells(n + 1, 1).Value = TextBox1.Value  
        Cells(n + 1, 2).Value = TextBox2.Value  
        Cells(n + 1, 3).Value = TextBox3.Value  
        Cells(n + 1, 4).Value = ComboBox1.Value  
        Cells(n + 1, 5).Value = TextBox4.Value  
    End If  
End If  
End Sub
```

В завершение добавим кнопку на рабочем листе (элемент управления ActiveX), чтобы запускать программу.

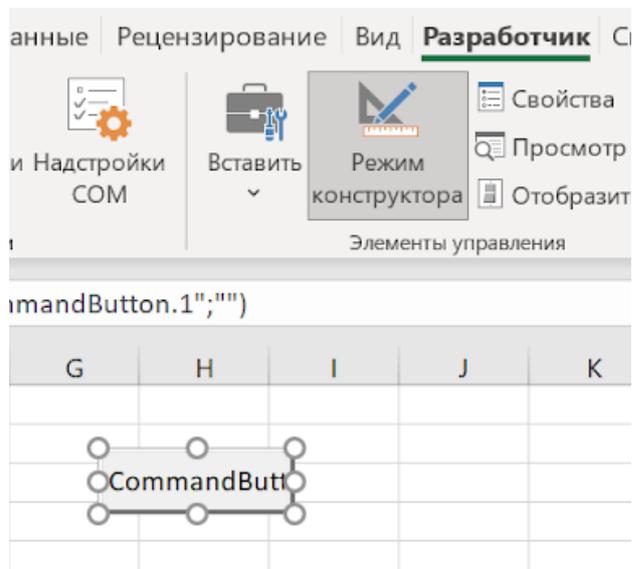


Рис. 42. Режим конструктора

По нажатию на этой кнопке должен быть записан следующий код:

```
Private Sub CommandButton1_Click()
    UserForm1.Show
End Sub
```

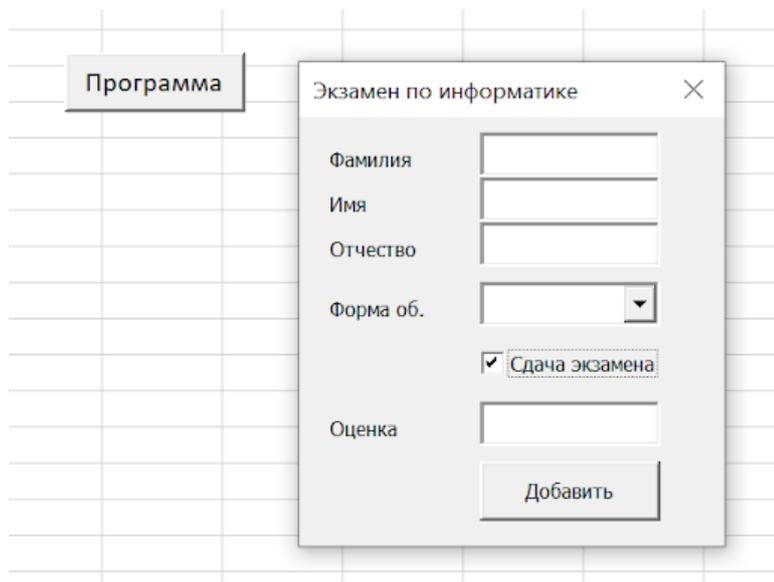


Рис. 43. Работа приложения

Самостоятельное занятие №9

«Дизайн интерфейса собственного приложения и программирование его функций»

Цель: освоить на практике разработку интерфейса и функционала собственного приложения в VBA.

Задание:

Вам необходимо разработать проект, в котором нужно реализовать свои знания по четырем разделам: линейные вычислительные процессы (1), разветвляющиеся вычислительные процессы (2), циклические вычисления (3), формы (4).

Требования:

1. У вас должно быть рабочее окно (по желанию можно добавить дополнительные рабочие окна или вкладки MultiPage). В них вы будете вводить данные. Затем данные

будут обрабатываться, и вы будете выводить результаты в окне MsgBox и на лист.

2. Установите тему своего проекта. Например, анкетирование, ввод информации о людях/жителях/студентах/клиентах, о фильмах/книгах/искусстве, о покупках или продажах, о странах/поездках, информации по тематике вашего любимого предмета/хобби/увлечения. Темой может быть что угодно.

3. Чтобы вводить данные используйте TextBox, ComboBox, ListBox, CheckBox или другие элементы. Для обозначений используйте Label. Для программирования действий используйте CommandButton. Для вывода информации используйте Label, TextBox или другие элементы управления, MsgBox или ячейки на листе - что посчитаете правильным. Разработайте дизайн окна на свой вкус.

4. Продемонстрируйте умение применять вышеуказанные разделы VBA:

- использование формул VBA (линейные вычислительные процессы);
- проверка на определенное условие и расчет в зависимости от него (ветвление);
- многократная обработка или вывод информации (циклы).

5. Во время защиты проекта преподаватель может попросить изменить строку кода или прокомментировать ваш код или форму.