

МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА РФ

ФГБОУ ВО «Кубанский государственный
аграрный университет имени И. Т. Трубилина»

Д. А. Замотайлова

Учебная практика: по получению первичных
профессиональных умений и навыков,
в том числе первичных умений и навыков научно-
исследовательской деятельности
(в области информатики)

Учебное-методическое пособие

Краснодар
КубГАУ
2018

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	3
1 ОБЩИЕ СВЕДЕНИЯ ОБ УЧЕБНОЙ ПРАКТИКЕ: ФОРМИРУЕМЫЕ КОМПЕТЕНЦИИ, ПРОХОЖДЕНИЕ, ЗАЩИТА ОТЧЕТА	4
2 РАБОТА С ДОКУМЕНТОМ В MS EXCEL С ПРИМЕНЕНИЕМ VBA	4
3 РАБОТА С ДОКУМЕНТОМ В MS ACCESS	4
ЗАКЛЮЧЕНИЕ	4
Приложение А	4
Приложение Б.....	4
Приложение В	4

ВВЕДЕНИЕ

Учебная практика по получению первичных профессиональных умений и навыков, в том числе первичных умений и навыков научно-исследовательской деятельности (в области информатики) является обязательным элементом реализации ОПОП ВО по направлению подготовки 09.03.03 Прикладная информатика, направленность «Прикладная информатика в экономике» (программа прикладного бакалавриата).

Целью учебной практики является подготовка обучающихся к эффективному использованию современных компьютерных средств для решения прикладных задач как в процессе обучения в вузе, так и в будущей профессиональной деятельности.

На практике обучающиеся овладеют дополнительными практическими навыками работы на компьютере с операционной системой Windows XP/Windows 7 и прикладными программами (Word, Excel, Access), приобретут расширенные практические навыки и получат опыт создания программных продуктов и применения численных методов для решения информационных задач на базе языка программирования VBA.

Задачами учебной практики являются:

- закрепление основы и углубление знаний в области разработки макросов;
- получение дополнительных практических навыков в использовании основных приёмов обработки экспериментальных данных, численного решения уравнений, решения задач оптимизации, выбора необходимых программных и технических средств для конкретных целей пользователя.
- приобретение опыта работы в решении экономических задач: изучение принципов построения информационно-правовых баз данных, применяемых на практике, а также приобретение практического опыта их применения: изучение дополнительного материала публикуемого в периодической печати, с целью актуализации знаний полученных в процессе обучения.

Данное учебно-методическое пособие предназначено для обучающихся в целях обеспечения высокого уровня их подготовки в рамках учебной практики.

1 Общие сведения об учебной практике: формируемые компетенции, прохождение, защита отчета

Вид практики – учебная, тип практики – по получению первичных профессиональных умений и навыков, в том числе первичных умений и навыков научно-исследовательской деятельности.

Учебная практика проходит на кафедре информационных систем факультета прикладной информатики КубГАУ после завершения сессии II учебного семестра в течении двух недель.

Способ проведения практики – стационарная.

Практика проводится дискретно: путем выделения в календарном учебном графике непрерывного периода учебного времени для проведения практики.

В результате прохождения практики формируются следующие компетенции

ОПК-2 – способность анализировать социально-экономические задачи и процессы с применением методов системного анализа и математического моделирования

ОПК-4 – способность решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности

ПК-6 – способность собирать детальную информацию для формализации требований пользователей заказчика.

Формирование содержания практики в соответствии с профессиональными стандартами

Трудовые функции	Результаты (освоенные компетенции)	Виды работ на практике, трудовые действия
проектный вид деятельности		
Планирование проекта в соответствии с полученным заданием	ПК-2	подготовка текста плана управления проектом и частных планов в его составе (управления качеством, персоналом, рисками, стоимостью, содержанием, временем, субподрядчиками, закупками, изменениями, коммуникациями); разработка иерархической струк-

Трудовые функции	Результаты (освоенные компетенции)	Виды работ на практике, трудовые действия
		туры работ (ИСП) проекта в соответствии с полученным заданием; разработка расписания проекта в соответствии с полученным заданием; разработка сметы расходов проекта в соответствии с полученным заданием; разработка плана финансирования проекта в соответствии с полученным заданием; рассылка запросов на предложение поставки; сбор предложений поставщиков.

Для успешного прохождения учебной практики обучающиеся используют знания, умения, сформированные в ходе изучения дисциплин «Информатика», «Алгоритмическое и объектно-ориентированное программирование», «Математические и логические основы информатики». Знания, умения и приобретенные компетенции будут использованы при изучении многих дисциплин и разделов ОП, а также при написании выпускной квалификационной работы.

Общая трудоемкость учебной практики составляет 108 часов, 3 зачетные единицы.

Форма контроля – зачет с оценкой.

№ п/п	Разделы (этапы) практики	Виды производственной работы, на практике включая самостоятельную работу студентов и трудоемкость, в часах					Формы текущего и промежуточного контроля
		инструктаж	выполнение заданий, выполнение производственных функций и т.д.	сбор материала по программе в организации	самостоятельная работа	итога	
1	Проведение инструктажа по технике безопасности	2					–
2	Организация практики		6	6	6	18	Проверка дневника

№ п/п	Разделы (этапы) практики	Виды производственной работы, на практике включая самостоятельную работу студентов и трудоемкость, в часах					Формы текущего и промежуточного контроля
		инструктаж	выполнение заданий, выполнение производственных функций и т.д.	сбор материала по программе в организации	самостоятельная работа	итого	
3	Подготовительный этап		4	6	6	18	Проверка дневника
4	Производственный (экспериментальный) этап		6	6	6	18	Проверка дневника
5	Исследовательский этап		6	6	6	18	Проверка дневника
6	Обработка и анализ полученной информации		6	6	6	18	Проверка дневника
7	Подготовка отчета по практике.		6	6	6	18	Проверка отчета
	Всего, час	2	34	36	36	108	Зачет с оценкой (дифференцированный)

Основным итоговым документом, отражающим степень и качество выполнения обучающимся задания практики, является отчет по практике. Контрольные вопросы и задания применяются в процессе аттестации в случае невозможности оценить результаты практики по материалам отчета.

Заключительный контроль (промежуточная аттестация) подводит итог прохождению учебной практики. Учебным планом по учебной практике предусмотрен зачет с оценкой.

Отчет о прохождении практики подготавливается в текстовом редакторе Microsoft Word и представляет собой отчет о проделанной работе: информацию о документе (документах), автоматизация которого производилась; сведения, полученные обучающимся в рамках осуществления самостоятельной научно-исследовательской работы, касающиеся процедуры работы с выбранными документами,

особенностей учета и т.д.; результаты автоматизации в MS Excel и MS Access. В отчете должны найти отражения все этапы прохождения практики.

Защита отчета предполагает ответы на вопросы по защищаемой теме и демонстрацию на компьютере результатов автоматизации.

Помимо отчета о прохождении практики обучающимся предоставляется также пакет документов: индивидуальное задание, выданное обучающемуся руководителем практики в начале ее прохождения; рабочий график (план), сформированный обучающимся совместно с руководителем практики в начале ее прохождения; дневник прохождения практики, заполненный в соответствии с индивидуальным планом и рабочим графиком (планом). Руководитель практики формирует отзыв на работу обучающегося, заполняет аттестационный лист при проведении защиты отчета.

Примеры документов приведены в Приложении А.

Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы

Номер семестра (этап формирования компетенции соответствует номеру семестра)	Этапы формирования компетенций по дисциплинам, практикам в процессе освоения ОП
ОПК-2 – способность анализировать социально-экономические задачи и процессы с применением методов системного анализа и математического моделирования	
2	Основы экономической кибернетики
2	Учебная практика: по получению первичных профессиональных умений и навыков, в том числе первичных умений и навыков научно-исследовательской деятельности (в области информатики)
3	Прикладные нечеткие системы
3	Компьютерная математика
4	Теория систем и системный анализ
4	Математическая экономика
5	Исследование операций
5	Имитационное моделирование
5	Инструментальные средства математической экономики
6	Инструментальные средства моделирования бизнес процессов
7	Информационные системы управления проектами
7	Методы и системы интеллектуального анализа данных
ОПК-4 – способность решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности	
1	Информатика

Номер семестра (этап формирования компетенции соответствует номеру семестра)	Этапы формирования компетенций по дисциплинам, практикам в процессе освоения ОП
2	Учебная практика: по получению первичных профессиональных умений и навыков, в том числе первичных умений и навыков научно-исследовательской деятельности (в области информатики)
1, 2	Алгоритмическое и объектно-ориентированное программирование
3	Программная инженерия
3	Информационные системы и технологии
3	Теоретические основы баз данных
4	Компьютерные системы
4	Машинное обучение
5	Основы изобретательской деятельности и защита интеллектуальной собственности
6	Проектирование информационных систем
6	Информационная безопасность
6	Интеллектуальные информационные системы
7	Электронная коммуникация
7	Рынки информационно-коммуникационных технологий и информационное обеспечение организации продаж
8	IT- инфраструктура предприятий (организаций)
ПК-2 – способность разрабатывать, внедрять и адаптировать прикладное программное обеспечение	
1, 2	Алгоритмическое и объектно-ориентированное программирование
3	Программная инженерия
3	Алгоритмы и структуры данных
3	Прикладные нечеткие системы
3	Компьютерная математика
3	Веб-дизайн и интернет-программирование
3	Информационные технологии визуализации экономической информации
4	Машинное обучение
4	Разработка приложений в среде MS Office
4	Информационные системы управления производственной компанией
5	Инструментальные средства математической экономики
5	Инструментальные средства баз данных
6	Интеллектуальные информационные системы
6	Математическое и инструментальное обеспечение автоматизированных рабочих мест
6	Разработка приложений под мобильные устройства
6	Системы компьютерной математики
7	Информационные системы управления проектами
7	Методы и системы интеллектуального анализа данных
8	Информационные системы в управлении и финансовой деятельности
8	Информационные системы в бухгалтерском учете

Номер семестра (этап формирования компетенции соответствует номеру семестра)	Этапы формирования компетенций по дисциплинам, практикам в процессе освоения ОП
8	Преддипломная практика

Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы

Компетенция	Содержание в соответствии с ФГОС ВО	Этап (период) прохождения практики, в течение которого формируется компетенция
ОПК-2	способность анализировать социально-экономические задачи и процессы с применением методов системного анализа и математического моделирования	Организация практики Подготовительный этап Производственный (экспериментальный) этап Исследовательский этап Обработка и анализ полученной информации Подготовка отчета по практике.
ОПК-4	способность решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности	Организация практики Подготовительный этап Производственный (экспериментальный) этап Исследовательский этап Обработка и анализ полученной информации Подготовка отчета по практике.
ПК-2	способность разрабатывать, внедрять и адаптировать прикладное программное обеспечение	Организация практики Подготовительный этап Производственный (экспериментальный) этап Исследовательский этап Обработка и анализ полученной информации Подготовка отчета по практике.

Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

Планируемые результаты освоения компетенции	Уровень освоения				Оценочное средство
	неудовлетворительно (минимальный)	удовлетворительно (пороговый)	Хорошо (средний)	Отлично (высокий)	
ОПК-2 - способность анализировать социально-экономические задачи и процессы с применением методов системного анализа и математического моделирования					
<p>Знать: методы системного анализа и математического моделирования; управление рисками проекта; возможности ИС; предметная область.</p> <p>Уметь: давать сравнительную характеристику различных моделей, используемых в сфере ИКТ; эффективно использовать аппарат математического моделирования и системный анализ в сфере проектирования ИС; анализировать входные данные; планировать работы в проектах в области ИТ.</p> <p>Владеть: навыками анализа социальных и экономических задач и процессов методами системного анализа и математического моделирования.</p>	Отсутствуют все необходимые знания, умения, владения для анализа социально-экономических задач и процессов с применением методов системного анализа и математического моделирования.	Обладает требуемыми знаниями для анализа социально-экономических задач и процессов с применением методов системного анализа и математического моделирования.	Обладает требуемыми знаниями и умениями для анализа социально-экономических задач и процессов с применением методов системного анализа и математического моделирования.	Обладает требуемыми знаниями, умениями, навыками для анализа социально-экономических задач и процессов с применением методов системного анализа и математического моделирования.	Контрольные вопросы, отчет по практике
ОПК-4 - способность решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности					
Знать: основные требования по организации защиты	Отсутствуют все необходимые знания, умения,	Обладает требуемыми знаниями для обеспечения	Обладает требуемыми знаниями и умениями для	Обладает требуемыми знаниями, умениями, навыками	Контрольные вопросы, отчет по практике

Планируемые результаты освоения компетенции	Уровень освоения				Оценочное средство
	неудовлетворительно (минимальный)	удовлетворительно (пороговый)	Хорошо (средний)	Отлично (высокий)	
<p>информации; основы юридических взаимоотношений между контрагентами; инструменты и методы выдачи и контроля поручений.</p> <p>Уметь:</p> <p>перечислять и давать общую характеристику видов и источников угроз безопасности; оценивать источники угроз информационной безопасности для различных профессиональных областей; использовать современные средства защиты информации; разрабатывать договоры на основе типовой формы; анализировать входные данные; контролировать выданные поручения.</p> <p>Владеть:</p> <p>технологиями использования современных ИКТ в рамках профессиональной деятельности.</p>	<p>владения для обеспечения возможности решения стандартных задач профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности.</p>	<p>возможности решения стандартных задач профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности.</p>	<p>обеспечения возможности решения стандартных задач профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности.</p>	<p>для обеспечения возможности решения стандартных задач профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности.</p>	
ПК-2 - способность разрабатывать, внедрять и адаптировать прикладное программное обеспечение					
<p>Знать:</p> <p>теоретическое и практическое содержание этапов процесса внедрения, адаптации и настройки прикладного программного обеспечения.</p>	<p>Отсутствуют все необходимые знания, умения, владения для разработки, внедрения и адаптации прикладного программного обеспечения.</p>	<p>Обладает требуемыми знаниями для разработки, внедрения и адаптации прикладного программного обеспечения.</p>	<p>Обладает требуемыми знаниями и умениями для разработки, внедрения и адаптации прикладного программного обеспечения.</p>	<p>Обладает требуемыми знаниями, умениями, навыками для разработки, внедрения и адаптации прикладного программного обеспечения.</p>	<p>Контрольные вопросы, отчет по практике</p>

Планируемые результаты освоения компетенции	Уровень освоения				Оценочное средство
	неудовлетворительно (минимальный)	удовлетворительно (пороговый)	Хорошо (средний)	Отлично (высокий)	
<p>печения; дисциплины управления проектами; возможности ИС; предметная область; технологии межличностной и групповой коммуникации в деловом взаимодействии, основы конфликтологии; инструменты и методы коммуникаций; каналы коммуникаций; модели коммуникаций.</p> <p>Уметь: организовывать и управлять процессом внедрения, адаптации и настройки прикладного программного обеспечения, в т.ч., распределять работы, выделять ресурсы, контролировать исполнение; проводить переговоры; проводить интервью; разрабатывать документы; осуществлять коммуникации.</p> <p>Владеть: навыками внедрения, адаптации и настройки прикладного программного обеспечения, обеспечивая соответствие и контроль разработанного кода и процесса</p>	<p>программного обеспечения.</p>		<p>обеспечения.</p>	<p>обеспечения.</p>	

Планируемые результаты освоения компетенции	Уровень освоения				Оценочное средство
	неудовлетворительно (минимальный)	удовлетворительно (пороговый)	Хорошо (средний)	Отлично (высокий)	
кодирования принятым в организации регламентам и стандартам.					

Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

В случае необходимости дополнительной проверки знаний, умений и навыков обучающихся, полученных в рамках прохождения учебной практики, ему могут быть заданы следующие примерные вопросы, связанные с тематикой практики:

1. Возможности и тенденции развития табличных процессоров
2. Надстройки MS Excel
3. Обзор встроенных функций MS Excel
4. Создание списков и баз данных в среде MS Office
5. Современные системы управления базами данных (СУБД) и их применение
6. Программные средства реализации деловой и коммерческой графики
7. Автоматизация работы пользователя в среде MS Office
8. Современные языки и системы программирования
9. Объектно-ориентированное программирование
10. Возможности средств мультимедиа и перспективы их использования
11. Особенности проведения автоматизации документооборота на предприятии
12. Подходы к работе с научными и специальными источниками и современными информационно-коммуникационными технологиями
13. Создание базы данных в MS Access
14. Работа с отчетами в MS Access

15. Работа с запросами в MS Access
16. Создание форм в MS Access
17. Проектирование систем автоматизации документооборота
18. Основы работы с макросами
19. Процесс создания макроса в MS Excel
20. Основы синтаксиса VBA
21. Основы создания функций в VBA
22. Создание форм с помощью VBA
23. Создание экранных форм с помощью VBA
24. Автоматизация экранных форм в VBA
25. Работа с мастером подстановок в MS Access

Критерии оценивания результатов обучения

Результаты выполнения и защиты отчета по учебной практике, оцениваются «отлично», «хорошо», «удовлетворительно», «неудовлетворительно», или «зачтено», «не зачтено» и заносятся в зачетную книжку студента, протокол защиты отчета, ведомость.

Наименование оценочного средства	Критерии оценивания компетенций (результатов)	Оценка	Критерии оценивания
Письменный отчёт по практике (научно-исследовательская работа), рабочий график (план) и дневник практики Выступление обучающегося во время защиты отчета	– соответствие структуры и содержания разделов отчета по практике заданию, требованиям и методическим рекомендациям; – степень раскрытия сущности вопросов, качество представленных аналитических материалов, характеризующих объект исследования – соблюдение требований к оформлению – грамотность	Высокий уровень «5» (зачтено)	Оценку «отлично» или «зачтено» заслуживает обучающийся, который выполнил весь намеченный объем работы в срок и на высоком уровне в соответствии с программой практики, проявил самостоятельность, творческий подход и соответствующую профессиональную подготовку, показал владение теоретическими знаниями и практическими навыками проведения аналитического исследования, умение работать с аналитической информацией, и системно оценивать представленную в них информацию, а также умение делать выводы и аргументировать собственную позицию; требования к оформлению полностью соблюдены.
		Средний уровень «4» (зачтено)	Оценку «хорошо» или «зачтено» заслуживает обучающийся, который полностью выполнил намеченную на период практики программу, од-

Наименование оценочного средства	Критерии оценивания компетенций (результатов)	Оценка	Критерии оценивания
	речи и правильность использования профессиональной терминологии во время защиты отчета – полнота, точность, аргументированность ответов во время защиты отчета		нако допустил незначительные просчеты методического характера при общем хорошем уровне профессиональной подготовки, недостаточно полно представил аналитические материалы исследования, сформулировал предложения по решению выявленных в процессе практики проблем, составляющих сферу научных интересов обучающегося; имеются упущения в оформлении отчета.
		Пороговый уровень «3» (зачтено)	Оценку «удовлетворительно» или «зачтено» заслуживает обучающийся при частичном выполнении намеченной на период практики программы, если он допустил просчеты или ошибки методического характера, а представленный им информационный материал не позволяет в полной мере сформировать аналитическую базу исследования и требует соответствующей дополнительной обработки и систематизации; имеются существенные отступления от требований к оформлению отчета.
		Минимальный уровень «2» (не зачтено)	Оценки «неудовлетворительно» или «не зачтено» заслуживает обучающийся, не выполнивший программу практики и представивший отчет, выполненный на крайне низком уровне; требования к оформлению отчета не соблюдены.

Перечень основной и дополнительной учебной литературы

Основная

1. Метелица Н.Т. Основы информатики [Электронный ресурс]: учебное пособие/ Метелица Н.Т., Орлова Е.В. – Электрон. текстовые данные. – Краснодар: Южный институт менеджмента, 2012. – 113 с. – Режим доступа: <http://www.iprbookshop.ru/9751>. – ЭБС «IPRbooks»

2. Прохорова О.В. Информатика [Электронный ресурс]: учебник/ Прохорова О.В. – Электрон. текстовые данные. – Самара: Самарский государственный архитектурно-строительный университет, ЭБС АСВ, 2013. – 106 с. – Режим доступа: <http://www.iprbookshop.ru/20465>. – ЭБС «IPRbooks»

3. Цветкова А.В. Информатика и информационные технологии [Электронный ресурс]: учебное пособие/ Цветкова А.В. – Электрон. текстовые данные. – Саратов: Научная книга, 2012. – 182 с. – Режим доступа: <http://www.iprbookshop.ru/6276>. – ЭБС «IPRbooks»

Дополнительная

1. Информатика [Электронный ресурс]: учебное пособие/ С.В. Тимченко [и др.]. – Электрон. текстовые данные. – Томск: Томский государственный университет систем управления и радиоэлектроники, Эль Контент, 2011. – 160 с. – Режим доступа: <http://www.iprbookshop.ru/13935>. – ЭБС «IPRbooks»

2. Метелица Н.Т. Информатика. Часть 1 [Электронный ресурс]: учебное пособие/ Метелица Н.Т., Орлова Е.В. – Электрон. текстовые данные. – Краснодар: Южный институт менеджмента, 2009. – 114 с. – Режим доступа: <http://www.iprbookshop.ru/9554>. – ЭБС «IPRbooks»

3. Метелица Н.Т. Информатика. Часть 2 [Электронный ресурс]: учебное пособие/ Метелица Н.Т., Орлова Е.В. – Электрон. текстовые данные. – Краснодар: Южный институт менеджмента, 2009. – 99 с. – Режим доступа: <http://www.iprbookshop.ru/9556>. – ЭБС «IPRbooks»

4. Цветкова А.В. Информатика и информационные технологии [Электронный ресурс]: учебное пособие/ Цветкова А.В. – Электрон. текстовые данные. – Саратов: Научная книга, 2012. – 182 с. – Режим доступа: <http://www.iprbookshop.ru/6276>. – ЭБС «IPRbooks»

Перечень ресурсов информационно-телекоммуникационной сети «Интернет»

№	Наименование ресурса	Тематика	Уровень доступа	Начало действия и срок действия договора	Наименование организации и номер договора
1	РГБ	Авторефераты и диссертации	Доступ с компьютеров библиотеки (9 лицензий)	19.09.2017 - 1308.2018 (Со дня первого входа в ЭБС)	ФГБУ «Российская государственная библиотека» дог. Дог. №095/04/0155
2	Znanium.com	Универсальная	Интернет доступ	16.07.2018 16.07.2019	Договор № 3135 ЭБС
3	Издательство «Лань»	Ветеринария Сельск. хоз-во Технология хранения и переработки пищевых продуктов	Интернет доступ	12.01.18- 12.01.19	ООО «Изд-во Лань» Контракт №108
4	IPRbook	Универсальная	Интернет доступ	12.11.2017- 12.05.2018 18.05.18 – 18.12.18	ООО «Ай Пи Эр Медиа» Контракт №3364/17 Контракт №4042/18
5	Scopus	Универсальная	Доступ с ПК университета.	10.05.2018 31.12.2018	Договор SCOPUS/612 от 10.05.2018
6	Web of Science	Универсальная	Доступ с ПК университета.	02.04.2018 31.12.2018	Договор WoS/612 от 02.04.2018
7	Консультант Плюс	Правовая система	Доступ с ПК университета	01.01.2018 31.12.2018	Договор № 8068; от 15.01.2018
8	Научная электронная библиотека eLibrary (РИНЦ)	Универсальная	Интернет доступ		–
9	Образовательный портал КубГАУ	Универсальная	Доступ с ПК университета		
10	Электронный Каталог библиотеки КубГАУ	Универсальная	Доступ с ПК библиотеки		

Далее во второй и третьей части данного учебно-методического пособия приведены теоретические сведения и методические рекомендации по выполнению основных заданий учебной практики.

Пример титульного листа для отчета о прохождении практики приведен в Приложении Б.

Перечень заданий для прохождения практики приведен в Приложении В.

2 РАБОТА С ДОКУМЕНТОВ В MS EXCEL С ПРИМЕНЕНИЕМ VBA

Цель: Изучить основы работы с редактором VBA. Получить навык выполнения простейшей программы, сохранения и открытия файла, содержащего макросы.

Секрет использования VBA заключается в правильном понимании объектной модели в каждом отдельном приложении. Следует отметить, что VBA всего лишь управляет объектами, а у каждого программного продукта своя объектная модель. Приложением можно управлять программным образом только с помощью объектов, которые представлены в этом приложении.

1.Открытие окна редактора VBA.

Для удобной работы со средой целесообразно вывести на ленте управления вкладку «Разработчик». Для вывода вкладки «Разработчик» на ленте управления необходимо выполнить команду: «Кнопка Office» (☰), в появившемся диалоговом окне в правом нижнем выбрать кнопку «Параметры Excel», в окне «Параметры Excel» установить флажок в опции «Показывать вкладку разработчик на ленте» (рисунок 2.1).

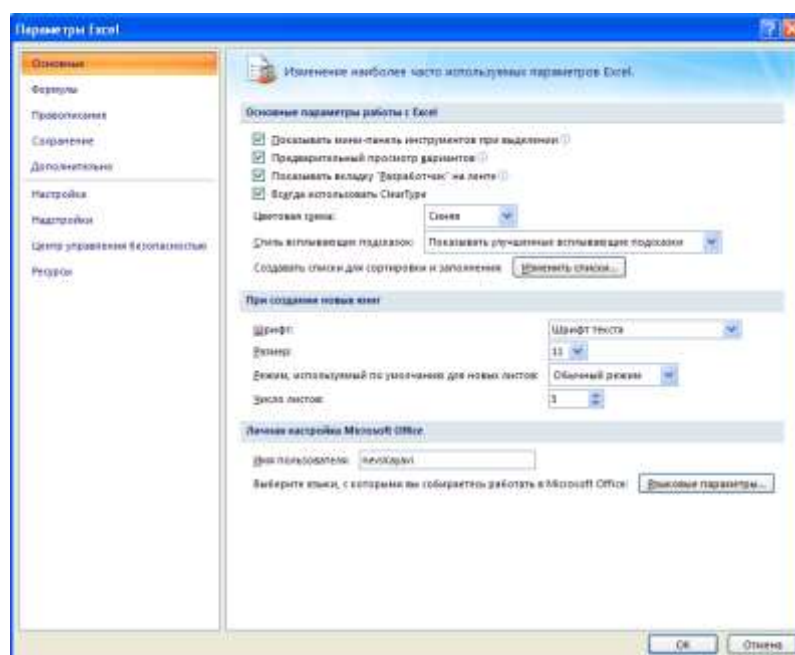


Рисунок 2.1 – Диалоговое окно «Параметры Excel»

В результате выполненных действий появится вкладка «Разработчик» Для запуска редактора Visual Basic необходимо щелкнуть мышью по команде «Visual Basic» (рисунок 2.2).

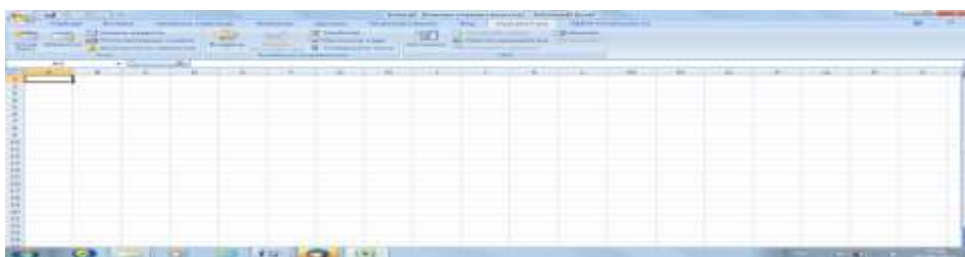


Рисунок 2.2 – Вкладка «Разработчик» на ленте управления

В результате выполненных действий откроется окно редактора VBA (рисунок 2.3).

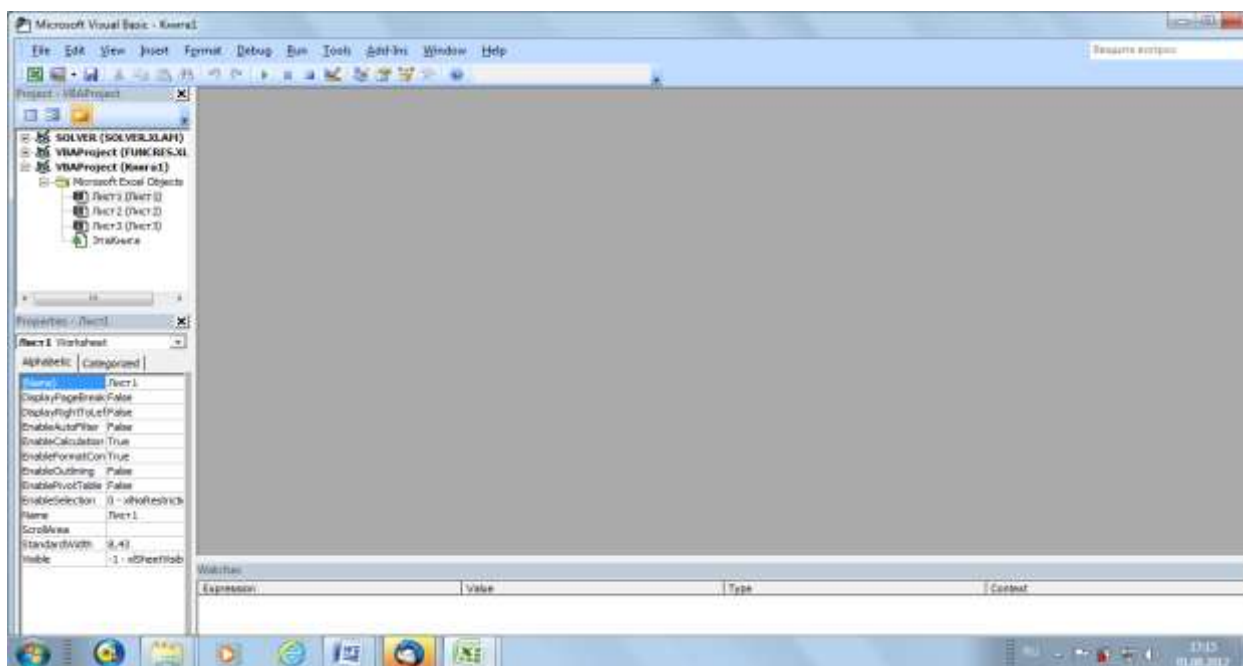


Рисунок 2.3 – Окно редактора VBA

2. Создание макроса.

Макрос, написанный на VBA, представляет собой последовательность кода, находящуюся в модуле VBA. Для того, чтобы приступить к написанию модуля необходимо выполнить команду: Insert/Module. В результате выполнения этой команды откроется окно документа, в котором необходимо вводить текст содержимого модуля.

Создайте модуль и введите текст макроса:

```
Sub приветствие()  
p = InputBox("введите слово")  
MsgBox "вы ввели:" & p  
End Sub
```

Для того, чтобы выполнить созданный макрос, выполните команду меню: Run/Run Sub.

Так же макрос можно выполнить из окна приложения Excel. Для перехода из редактора VBA в окно приложения достаточно выбрать кнопку «View Microsoft Excel» (рисунок 2.4).

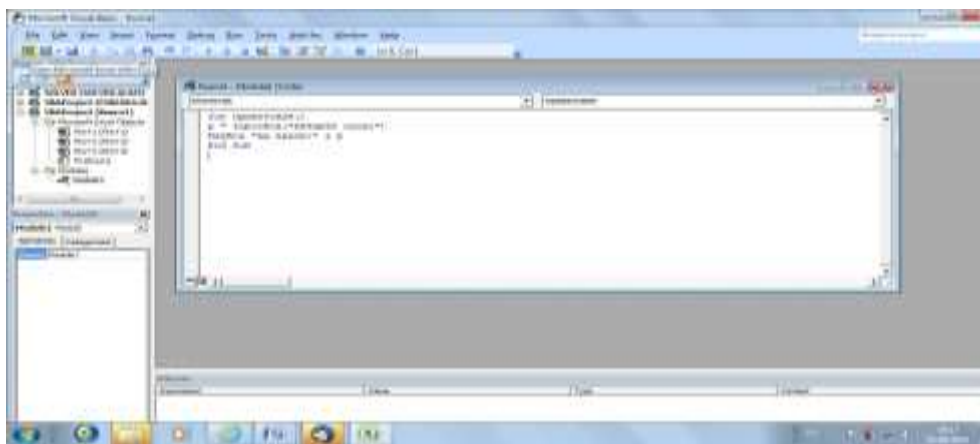


Рисунок 2.4 – Панель элементов редактора VBA

В приложении Excel для запуска макроса необходимо выполнить команду: Вид/Макросы, либо Разработчик/Макросы. В результате выполнения одной из этих команд появиться диалоговое окно «Макросы» (рисунок 2.5).

Командные кнопки, расположенные в правой части окна позволяют: выполнить макрос, перейти к редактору VBA с текстом макроса, удалить макрос. Так же через это диалоговое окно, выбрав команду «Создать» можно перейти в редактор VBA для создания нового макроса. Для создания нового макроса необходимо ввести имя вновь создаваемого макроса и выбрать команду «Создать».

Выполните команду: Вид/Макросы, в поле «Имя макроса» наберите: «пример» и нажмите кнопку «Создать».

В открывшемся окне редактора VBA наберите следующий текст:

Sub пример()

Dim x1 As Integer, x2 As Integer, x As Integer

x1 = InputBox("введите первое число")

x2 = InputBox("введите второе число")

x = x1 + x2

MsgBox "сумма введенных чисел равна:" & x

End Sub

Выполните макрос.

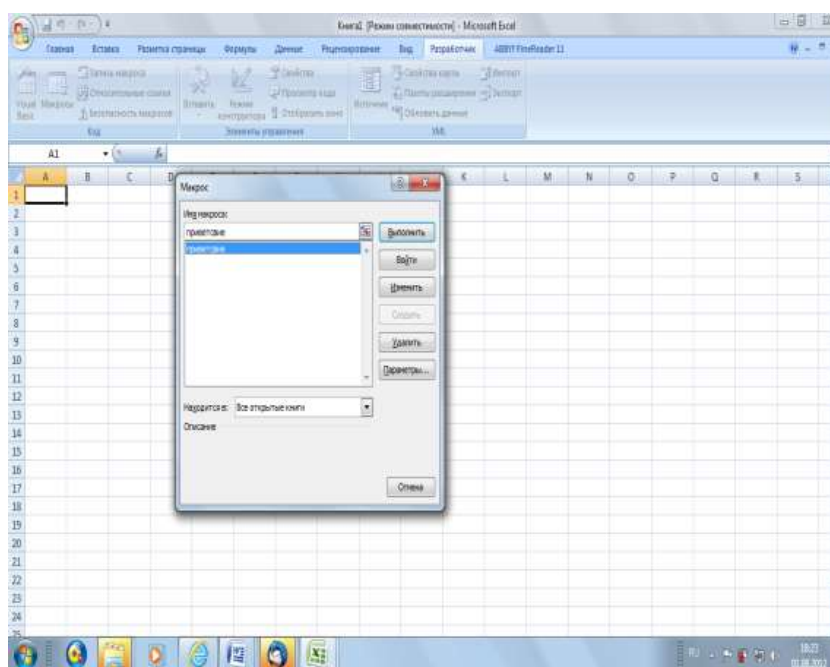


Рисунок 2.5 – Диалоговое окно «Макрос»

3. Сохранение файла, содержащего макрос.

Чтобы иметь доступ к макросам, сохраненным в файле, его необходимо сохранить с поддержкой макросов. Для этого необходимо выполнить команду: Кнопка Office/Сохранить как/Книга Excel с поддержкой макросов (рисунок 2.6).

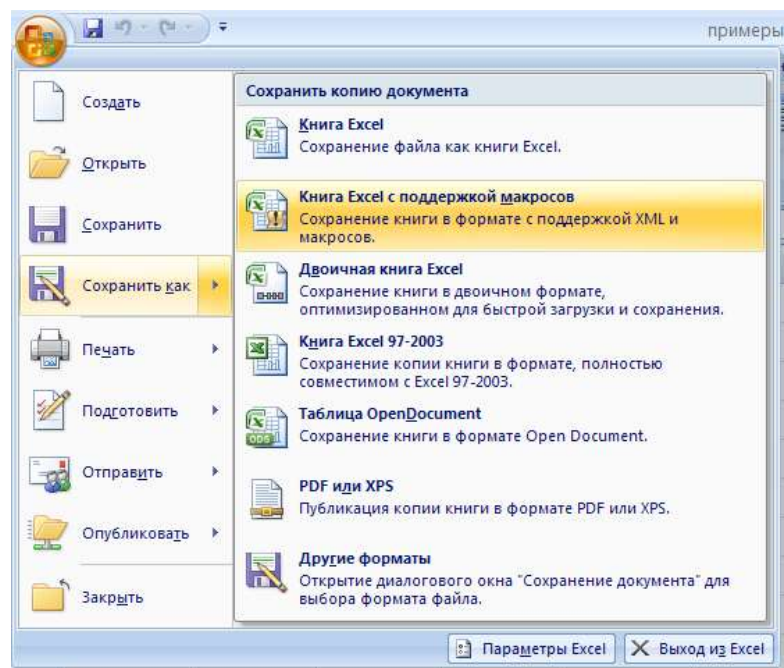


Рисунок 2.6 – Команда «Сохранить как»

Сохраните файл под именем: «Лабораторные работы VBA», закройте файл.

4. Пошаговое выполнение программы.

При отладке программы может возникнуть необходимость пошагового выполнения программы с просмотром промежуточных результатов.

Откройте файл лабораторной работы. Внимание! При открытии файла появится сообщение системы безопасности о том, что макросы отключены и кнопка «Параметры» (рисунок 2.7).



Рисунок 2.7 – Сообщение системы безопасности при открытии файла.

Для того, чтобы иметь доступ к текстам макросов, необходимо нажать кнопку «Параметры» и включить опцию «Включить это содержимое». Данную процедуру следует выполнять каждый раз при открытии файла, содержащего макросы.

Откройте текст макроса пример(). Для пошагового выполнения программы, а так же для вывода переменных, промежуточное зна-

чение которых будет просматриваться в окне просмотра используется пункт меню Debug (рисунок 2.8).

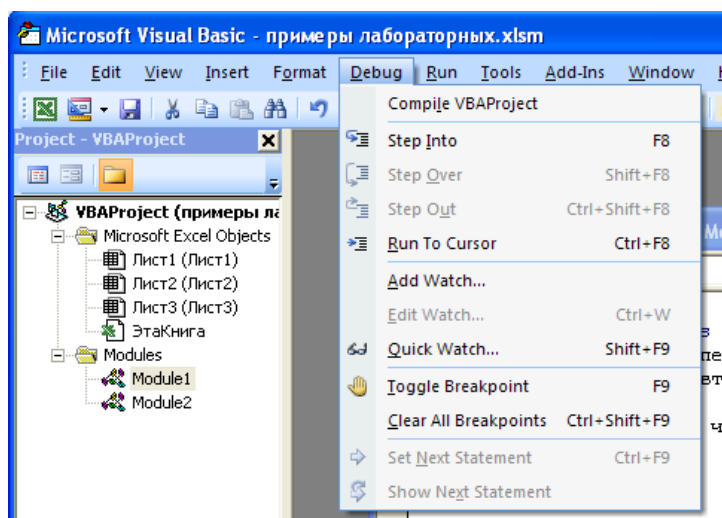


Рисунок 8 –Команда меню пункта Debug.

Таблица 2.1 – Значение пунктов меню работы с макросами

Пункт меню	Значение
Compile VBA project	Компилировать VBA проект
Step into	Шаг выполнения программы
Step over	Шаг через оператор
Step out	Шаг назад
Run to cursor	Выполнить до курсора
Add watch	Добавить переменную в окно просмотра
Edit watch	Редактировать переменную в окне просмотра
quick watch	Показать окно просмотра
Toggle breakpoint	Установить точку прерывания выполнения программы
Clear all breakpoints	Удалить все точки прерывания выполнения программы

Выполните команду debug/add watch. В результате выполнения команды появиться диалоговое окно Add watch (рисунок 2.9).

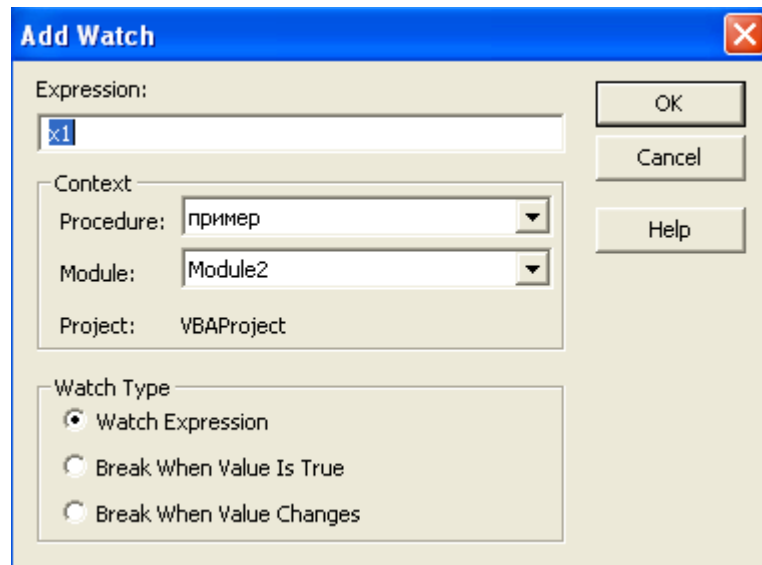


Рисунок 2.9 – Диалоговое окно Add watch

В поле Expression введите x1. Данная переменная отобразится в нижней части экрана в области Watch. Аналогично выведите в область Watch значение переменных x2,x. Начните пошаговое выполнение программы. Обратите внимание на то, как будут меняться значения переменных.

Сохраните файл, закройте приложение.

Основы синтаксиса VBA.

Цель: Изучить основы VBA: типы данных, правила именования переменных, правила объявления переменных, базовые операторы, реализующие линейный алгоритм. Приобретение навыков написания пользовательских функций и процедур.

В объектной модели Excel представлено несколько мощных объектов анализа данных, например, рабочие листы, диаграммы, сводные таблицы, сценарии, а также многочисленные математические, финансовые и пр. функции. С помощью VBA можно работать с этими объектами и разрабатывать автоматизированные процедуры.

Классы объектов организованы в иерархическую структуру. Excel-Application, рабочая книга – Workbook, рабочий лист – Worksheet. Рабочий лист содержит такие объекты как: диапазон – Range, ячейка – cells, диаграмма – chart и другие.

1. Ввод/вывод с помощью ячеек электронной таблицы.

Для обращения к ячейке текущего листа электронной таблицы в Excel используется оператор

Cells(<параметр1>, <параметр2>), где

<параметр1>- номер строки, в которой расположена ячейка.

<параметр2>- номер столбца, в которой расположена ячейка.

Например, для того чтобы обратиться к ячейке A1, необходимо указать `cells(1,1)`.

Если возникает необходимость обратиться к ячейке конкретного листа, то следует использовать следующий оператор:

Worksheets(<параметр3>).**Cells**(<параметр1>, <параметр2>), где параметр3- это имя листа, на котором расположена ячейка.

2. Имена переменных

Чтобы оперировать с переменной в программе, ей нужно присвоить некоторое имя. Стандартный способ сделать это в языке Visual Basic - написать следующий оператор:

Dim <имя переменной> **As** <тип переменной>.

Если требуется описать несколько переменных, то можно использовать следующую инструкцию:

Dim <имя переменной 1> **As** <тип переменной1>, <имя переменной 2> **As** <тип переменной 2>, ..., <имя переменной n> **As** <тип переменной n>

С помощью оператора **Dim** переменные объявляются или описываются. При объявлении переменной мы сообщаем Visual Basic, что нужно отвести участок для временного хранения данных и назвать его некоторым именем. Тип переменной, указанный после ключевого слова **As**, в частности, сообщает компилятору, сколько следует выделить памяти под переменную.

Имя (идентификатор) переменной выбирается произвольно, однако при этом следует соблюдать некоторые ограничения:

- Имя переменной может начинаться только с буквы;
- В оставшейся части имени можно использовать любую комбинацию букв, цифр и символа подчеркивания. В именах переменных нельзя использовать пробелы, точки и другие знаки препинания.
- Имя переменной должно быть уникальным в пределах ее зоны видимости.

- Длина имени переменной не должна превышать 255 символов.
- Имя переменной не должно совпадать ни с одним из ключевых слов языка.

При выборе переменной необходимо учитывать два противоречивых требования. С одной стороны, оно должно быть достаточно информативным, чтобы облегчить понимание программы, а с другой - как можно более коротким, чтобы ускорить ввод текста программы в компьютер.

3. Типы переменных

Род информации, которая может содержаться в переменной, определяется ее типом. Тип задает размер памяти, выделяемый под переменную, и набор допустимых операций, которые можно совершать с данной переменной. Типы переменных представлены в таблице 2.2.

Таблица 2 – Типы переменных

№ п/п	Тип	Размер байт	Содержимое	Область значений
1	Boolean	2	Логическое значение	True или False
2	Byte	1	Целое неотрицательное	0 ... 255
3	Integer	2	Целое	-32768 ... +32767
4	Long	4	Длинное целое	-2147483648 ... +2147483647
5	Single	4	Число с плавающей точкой одинарной точности	-3.402823E+38 ... -1.401298E-45; +1.401298E-45 ... +3.402823E+38
6	Double	8	Число с плавающей точкой двойной точности	-1.79769313486231E+308 ... -4.94065645841247E-324; +4.94065645841247E-324 ... +1.7 97 693134 8 6232E-308

7	Currency	8	Число с фиксированной точкой (денежная величина)	-922337203685477.5808 ... +922 3372 03 68 54 7 7.5 8 07
8	Decimal (подтип Variant)	14	Десятичное целое	- 79228162514264337593543950335 ... + 79228162514264337593543950335
			Десятичное с фиксированной точкой	-7.9228162 5142 64337593543950335 ... -0.00000000000000000000000000000001; + 0.00000000000000000000000000000001... +7.9228162 514264337593543950335
9	Date	8	Дата/время	#01/01/0100 00:00:00# ... #12/31/9999 23:59:59#
10	String		Строка переменной длины	Длина >= 0
	String * длина		Строка фиксированной длины	длина = 1 ... 65526
11	Object		Объект	Любая ссылка на объект
12	Variant		Все	

Тип данных **Variant** – это такой тип данных, который может сохранять в себе данные любых типов из числа допустимых в языке VBA. В переменной типа Variant может быть заключен текст, числовые значения, даты и объекты. В языке VBA тип данных Variant принят по умолчанию, т.е. если переменная была создана без указания типа, то она автоматически создается с типом Variant. Переменные этого типа принимают характеристики того конкретного типа данных, к которому относятся значения, сохраняемые в них в данный момент. Для числовых данных Variant принимает характеристики числового типа Double, для строковых данных – характеристики типа String и т. д. Благодаря такой гибкости переменные типа Variant очень удобны, поскольку избавляют программиста от необходимости использовать конкретные типы данных. Нет необходимости думать о том, что именно будет храниться в той или иной переменной – достаточно просто всем им назначить тип Variant. Однако, несмотря на все свои преимущества, этот тип данных предполагает использование большого объема памяти для хранения информации и требует значительных затрат времени при выполнении даже самых простых математических операций или операций сравнения. Поэтому целесообразно точно определить, какой

из возможных типов данных больше всего подходит для каждой величины, а затем объявлять конкретный тип данных для каждой используемой в программе переменной. Результат будет намного эффективнее, чем при использовании типа Variant для всех без исключения переменных в программе.

4. Объявление переменных

Компилятор Visual Basic не требует явного объявления переменных перед их использованием. Если используемая в программе переменная нигде ранее не описана, считается, что она имеет стандартный тип Variant. В переменной этого типа можно хранить любые данные, однако ее использование, как было отмечено ранее, сопряжено с потреблением дополнительных ресурсов.

Если явно не объявлять переменные, это может привести к ошибкам в программе.

5. Оператор присваивания

Прежде чем использовать переменную в программе, ей необходимо присвоить значение. Самый простой способ присвоения заключается в использовании оператора присваивания "=", который имеет следующий синтаксис:

«переменная» = «выражение»

Аргумент «переменная» задает имя переменной, которой будет присвоено значение выражения, стоящего справа от знака равенства. Например:

X = 10

S = "Иванов"

6. Математические операторы

Математические операторы позволяют выполнять в программе действия над числами. Математические операторы представлены в таблице 2.3.

Таблица 3 – Математические операторы

Оператор	Выполняемая операция
+	Сложение
-	Вычитание
*	Умножение
/	Деление
\	Целочисленное деление
Mod	Остаток от деления нацело
^	Возведение в степень

7. Создание подпрограмм

Текст модуля VBA состоит из подпрограмм. Существуют два вида подпрограмм: процедура и функция.

Структура подпрограммы включает в себя заголовок, описание переменных и операторы.

Для передачи значений между программой и подпрограммой применяется список параметров, расположенный в заголовке подпрограммы. Список параметров указывается после имени подпрограммы в круглых скобках. В списке указываются имена переменных и их тип. Список параметров может быть пустым.

Основное различие между процедурой и функцией в том, что через имя функции возвращается какое-то значение. Поэтому, в конце заголовка для функции требуется указать тип возвращаемого значения, а в области операторов подпрограммы присвоить имени функции требуемое значение.

7.1 Основы создания процедур

Каждой процедуре соответствует макрос. Вывод данных в простейшем случае осуществляется либо в ячейки Excel либо в окно сообщения MsgBox, соответственно ввод – либо из ячеек, либо с использованием диалогового окна InputBox.

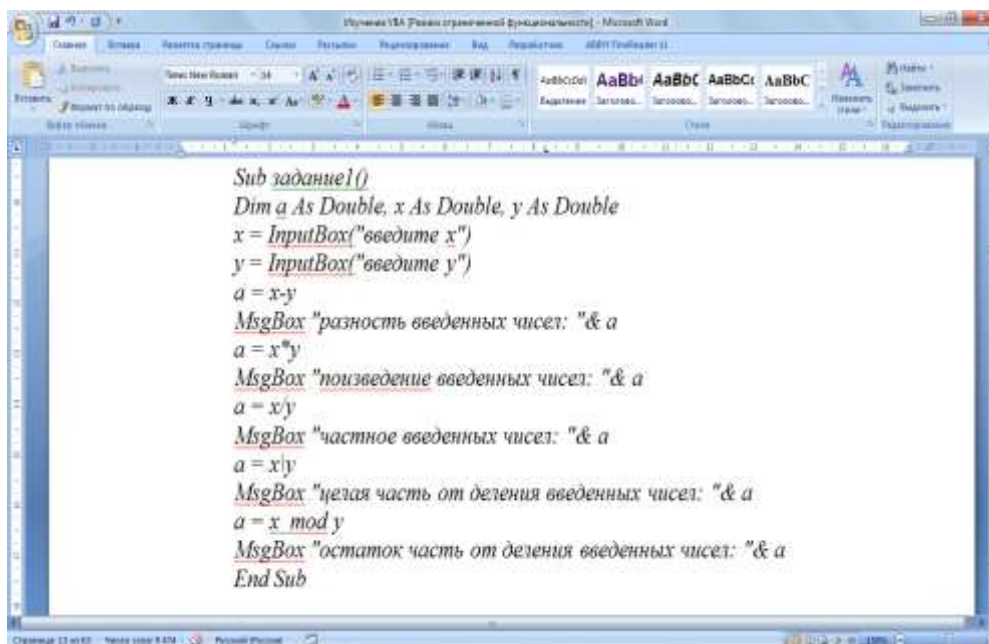
Синтаксис макроса выглядит следующим образом:

Sub «имя макроса» («параметры»)

операторы

End Sub

Откройте файл предыдущей лабораторной работы. Наберите в редакторе VBA текст следующего макроса:



```
Sub задание1()  
    Dim a As Double, x As Double, y As Double  
    x = InputBox("введите x")  
    y = InputBox("введите y")  
    a = x-y  
    MsgBox "разность введенных чисел: "&a  
    a = x*y  
    MsgBox "произведение введенных чисел: "&a  
    a = x/y  
    MsgBox "частное введенных чисел: "&a  
    a = x\y  
    MsgBox "целая часть от деления введенных чисел: "&a  
    a = x mod y  
    MsgBox "остаток часть от деления введенных чисел: "&a  
End Sub
```

7.2 Основы создания функций

Синтаксис функции выглядит следующим образом:

Function «имя функции» («параметры») [**As** «тип»]

операторы

End Function

Функции, как и переменные, имеют тип, задаваемый с помощью ключевого слова **As**. Если тип функции не задан, по умолчанию ей присваивается тип **Variant**. Тип функции определяет в свою очередь тип возвращаемого ею значения. Возвращаемое значение присваивается имени функции «имя функции» и может быть использовано в выражениях программного кода аналогично стандартным функциям Visual Basic. Для обращения к функции из Excel следует выбрать **мастер функций/категории заданные пользователем**. Использовать как обычную функцию Excel. Либо функцию можно по имени вызывать из другой функции или макроса.

Создание форм

Цель работы: получение практических навыков построения пользовательских форм и программирования действий, связанных с выводом информации на формы

Вы можете создавать свои окна с элементами управления. Они обычно нужны, когда требуется запросить информацию у пользователя. Разнообразные диалоги настроек, параметров и т. д. Такие окна называются пользовательскими формами.

Чтобы создать форму, откройте редактор VBA (Alt + F11), выберите в списке слева Microsoft Excel Objects и в его меню выберите Insert→UserForm.

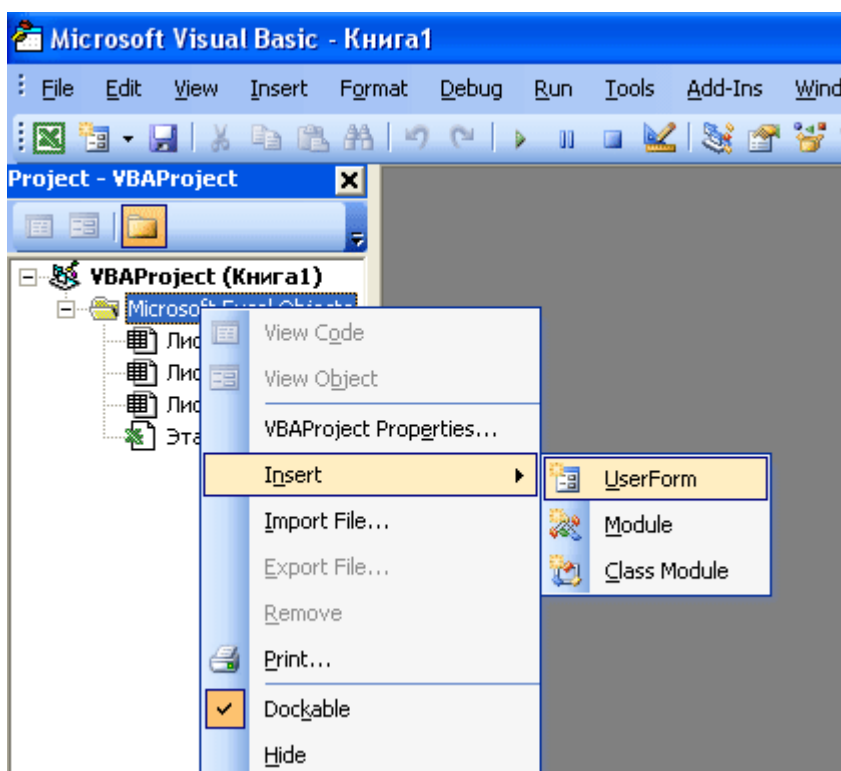


Рисунок 2.10 – Создание формы

На экране появилась пустая форма. В ее контекстном меню выбираем Properties – открывается окно свойств формы.

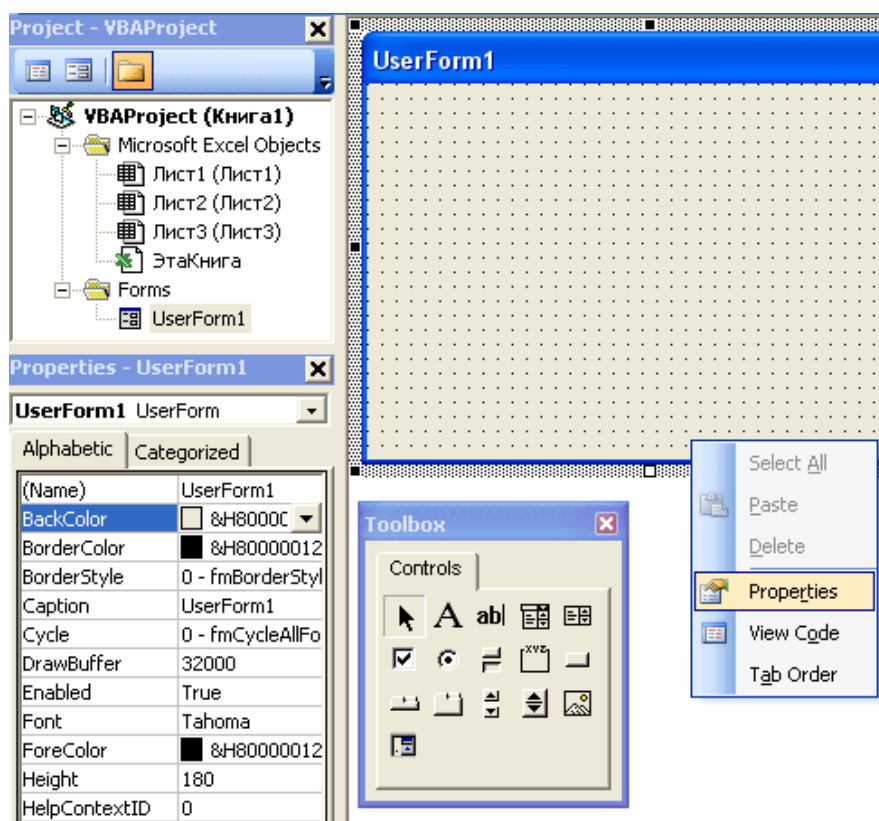


Рисунок 2.11 – Окно создания формы

В этом окне можно изменить заголовок формы, цвет фона, шрифт и многие другие параметры. Кроме окна свойств есть еще инструментальная панель Toolbox. В ней содержатся элементы управления, которые можно разместить на форме. Для начала добавим, например, кнопку.

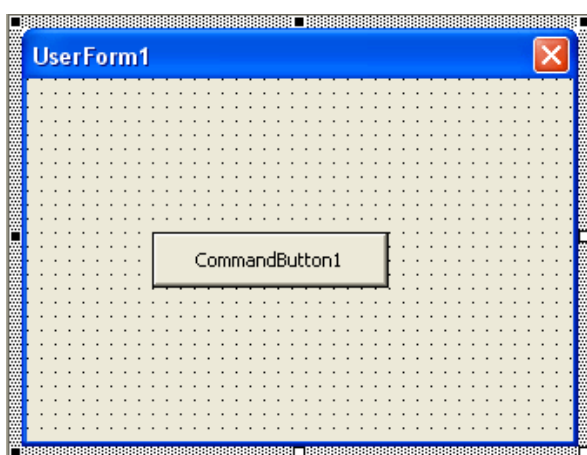


Рисунок 2.12 – Добавление кнопки на форму

Теперь в окне свойств доступны свойства кнопки. Можно изменить надпись на ней, или добавить картинку. Осталось только привязать к этой кнопке свой код. Это очень просто - по двойному щелчку на кнопке вы попадете в окно редактора VBA, где уже создана процедура обработки нажатия на кнопку.

```
Private Sub CommandButton1_Click()  
End Sub
```

Задание

Создать экранную форму «Ведомость лабораторных работ» для заполнения журнала учёта лабораторных работ (рисунок 2.13).

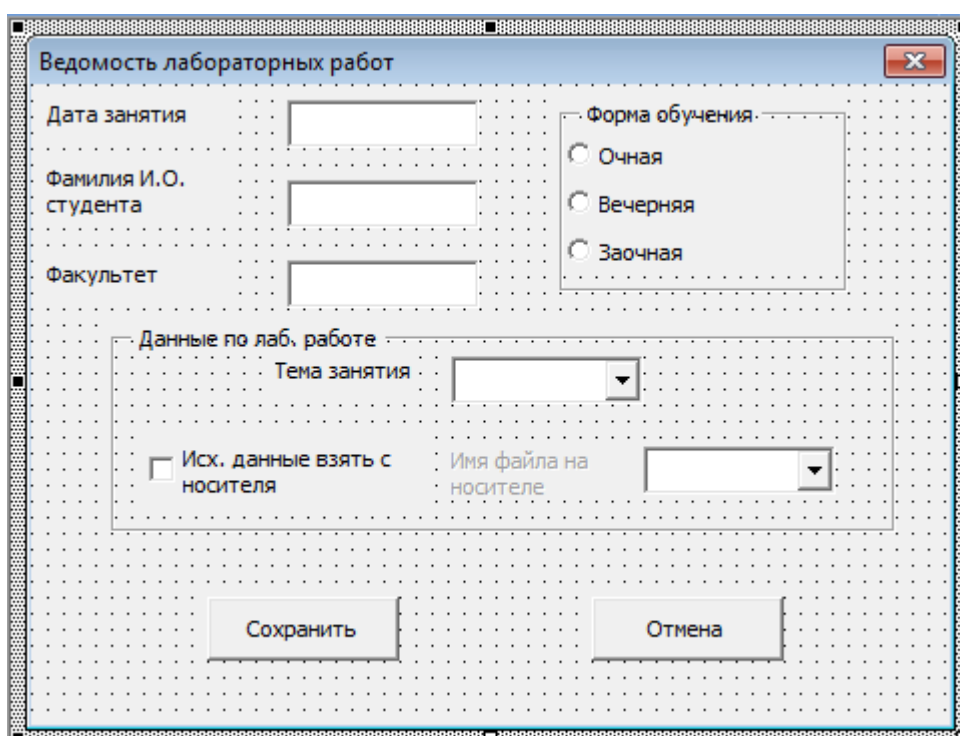


Рисунок 2.13 – Форма «Ведомость лабораторных работ»

Решение

Разобьем задачу на несколько этапов.

1.Создание экранной формы

Для этого:

- 1.1 Открыть новую рабочую книгу в Excel.
- 1.2 Перейти в редактор Visual Basic (нажмите комбинацию клавиш Alt+F11).

1.3 Для вставки формы выполните команды Вставка – Экранная форма (для нерусифицированной версии Insert —UserForm). В рабочую книгу будет вставлена экранная форма, показанная на рисунке 2.14.

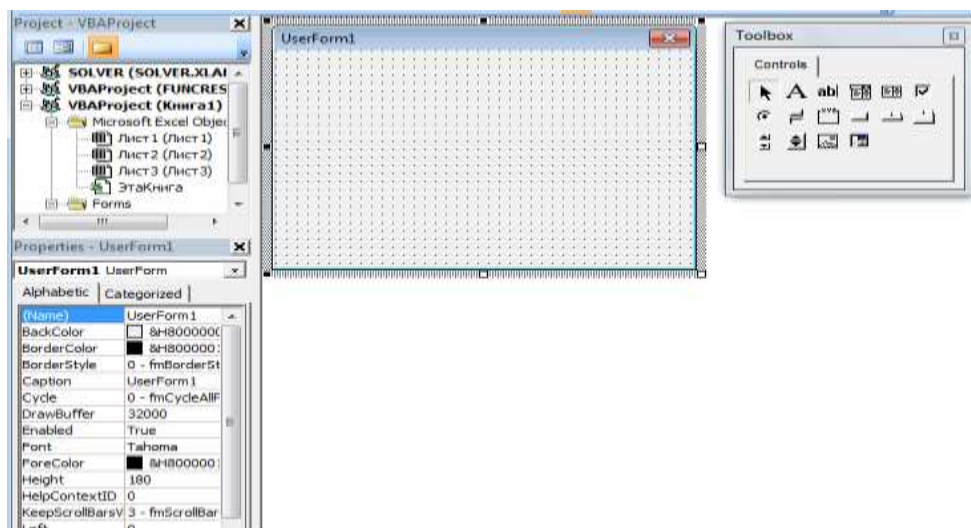


Рисунок 2.14 – Вставка формы

Новая форма представляет собой пустое серое окно со строкой заголовка. Это та основа, на которой пользователь может создать собственные окна любых типов.

1.4 Установим значения свойств формы: т. е. дадим ей имя и введём название в строку заголовка.

Для этого:

- Если окно свойств не отображено на экране, выполнить команды Вид – Окно свойств (или нажать клавишу F4). Появится окно свойств (рисунок 2.15).

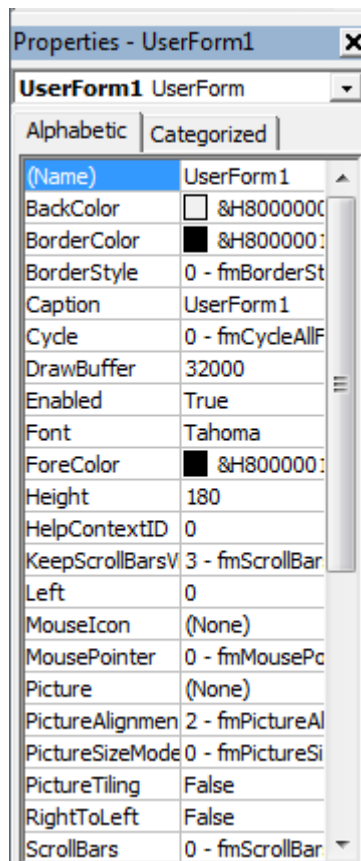


Рисунок 2.15 – Окно свойств

- Выберите свойство Name (Имя), которое находится в верхней части списка, упорядоченного по алфавиту. Вместо UserForm1 введите Ведомость. Это имя, по которому к форме будет обращаться программа

- Для ввода заголовка формы выберите свойство Caption. Вместо UserForm1 (Visual Basic присваивает свойствам Name и Caption одинаковые названия) введите Ведомость лабораторных работ. Введённый текст сразу появится в строке заголовка формы

1.5 Вставим элементы управления в экранную форму. Мы хотим получить форму, показанную на рис. 4. Как видно, в форме использованы различные элементы управления.

- Чтобы вставить их в форму, выполните следующие действия:

Если панель элементов не отображена на экране, выполните команды Вид – Панель элементов (View – Toolbox).

Появится панель элементов рисунке 2.16.

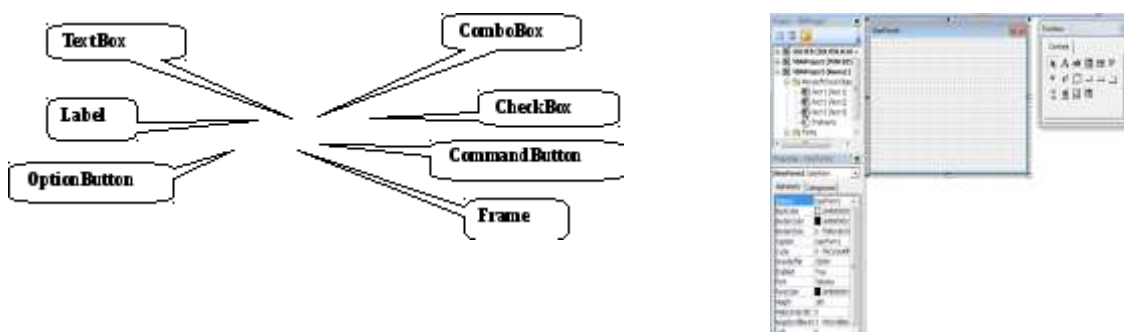


Рисунок 2.16 – Панель элементов



- На панели элементов выберите элемент Надпись (Label): . Поставив указатель мыши на этот элемент, нажмите левую клавишу и переместите его в левый верхний угол формы.
- Если окно свойств элемента Надпись скрыто, нажмите клавишу F4. 'Задайте свойству Caption значение Дата занятия. При этом на форме вместо слова Label1 появится надпись «Дата занятия». Свойство Name оставляем без изменения, т.е. за этой надписью закреплено предложенное Visual Basic имя Label1.
- На панели элементов выберите элемент Поле (TextBox): , поместите его правее надписи Дата занятия. Сохраним за полем предлагаемое Visual Basic имя TextBox1.
- Продолжайте аналогично предыдущим подпунктам создавать элементы управления и описывать значения их свойств согласно таблице 2.4.

Таблица 2.4 – Элементы управления и их свойства

Тип элемента управления	Значение свойства Name	Значение свойства Caption	Другие свойства
Надпись	Label1	Дата занятий	-
Поле	TextBox1	-	-
Надпись	Label2	Фамилия И.О. студента	-
Поле	TextBox2	-	-
Надпись	Label3	Факультет	-
Поле	TextBox3	-	-
Рамка	Frame1	Форма обучения	-
Переключатель	OptionButton1	Очная	Value=True
Переключатель	OptionButton2	Вечерняя	-

Переключатель	OptionButton3	Заочная	-
Рамка	Frame2	Данные по лаб. работе	-
Надпись	Label4	Тема занятия	-
Поле со списком	ComboBox1	-	-
Флажок	CheckBox1	Исх. данные взять с носителя	Value=False
Надпись	Label5	Имя файла на носителе	Enabled=False
Поле со списком	ComboBox2	-	-
Командная кнопка	ComandButton1	Сохранить	Default=True
Командная кнопка	ComandButton2	Отмена	Cancel=True

Замечание 1. В форму дважды нужно вставить элемент Рамка (Frame). Сначала следует *разместить на форме* этот элемент, а затем поместить в нём другие элементы управления, иначе ража закроет их собой.

Замечание 2. При вставке элементов управления лучше придерживаться того порядка, который указан в табл. 4 (в противном случае Visual Basic присвоит им имена с другой нумерацией и при автоматизации формы они будут работать неправильно).

Для каждого элемента мы сохраняем предложенное Visual Basic свойство Name, для элементов Надпись. Переключатель (Опция), Рамка и Флажок задаём свойство Caption. Для ряда элементов управления необходимо также задать дополнительные свойства.

Прокомментируем эти дополнительные свойства.

Свойства командных кнопок Кнопка Сохранить. Чтобы командная кнопка работала как кнопка Сохранить, необходимо задать ей значение свойства *Default* (По умолчанию), равное *True* (Истина). В экранной форме только одна командная кнопка может иметь значение *True* свойства *Default*.

Кнопка Отмена. Щелчок по этой кнопке должен вызывать отмену введённых команд. Чтобы командная кнопка работала как кнопка Отмена, следует для неё задать свойству *Cancel* значение *True*. Так же, как и в случае свойства *Default*, в экранной форме только одна командная кнопка может иметь значение *True* свойства *Cancel*

Свойства переключателей Для переключателей в группе Форма обучения прежде всего нужно указать, какой из них будет вы-

бран по умолчанию (т. е. будет находиться во включённом состоянии до выбора, сделанного пользователем).

Для этого нужно задать значение *True* свойства *Value* (Значение). Установим его для первого переключателя Очная. Отметим, что в группе только один переключатель может иметь значение *True* свойства *Value*.

Блокировка элементов управления В нашей форме элемент управления **Имя файла** необходим только тогда, когда установлен флажок **Исходные данные взять с носителя**. Чтобы поле со списком **Имя файла** было недоступно пользователю, а также отображалось серым цветом (такой цвет сигнализирует о недоступности элемента), его свойство *Enabled* (Разблокировка) следует положить равным *False* (Ложь). Когда флажок **Исходные данные взять с носителя** будет установлен, свойство *Enabled* следует поменять на *True*. Это будет осуществляться программным путём.

Свойства флажка. Чтобы в исходном состоянии флажок не был установлен, следует задать равным *False* значение его свойства *Value*.

- Сохранить созданную форму с именем Ведомость.

2 Автоматизация экранных форм

Итак, нами создана экранная форма для ввода сведений об учёте лабораторных работ. Теперь нужно описать процедуры для вывода этой формы на экран и для сохранения введённых в неё данных.

2.1 Инициализация экранных форм

Инициализация означает создание процедуры, осуществляющей ввод в форму начальных данных (заполнение полей со списками и ввод даты занятий).

В процедуре инициализации экранной формы должны быть предусмотрены блоки задания исходных данных для полей со списками.

Для создания списков выполните следующие действия:

1. Перейдите в текущую рабочую книгу. Переименуйте рабочий Лист2 на «Списки».
2. На рабочем листе «Списки» введите исходные данные согласно таблице 2.5.

Таблица 2.5 – Исходные данные

	А	В	С
1	Тема занятия		Имя файла
2	Операционная система Windows		Список абонентов
3	Текстовый процессор Word		Инвестиционные проекты
4	Табличный процессор Excel		Оптимизационные задачи
5	СУБД Access		
6	Visual Basic		

3. Для присвоения диапазону ячеек имени *Занятия* выполнить следующее:

- выделить диапазон ячеек А2:А6;
- выполнить Вкладка ленты управления Формулы/(группа определенные имена)/ Присвоить имя:

- в открывшемся диалоговом окне ввести имя *Занятия*;

4. Аналогично диапазону ячеек С2:С4 присвойте имя *Файл* .

5. Сохраните рабочую книгу с прежним именем.

6. Перейдите в Visual Basic.

7. Дважды щёлкните по экранной форме (не на элементах управления, а именно по форме!). Откроется окно с заголовком процедуры UserForm_СКск (Щелчок по форме) – рисунок 2.17.

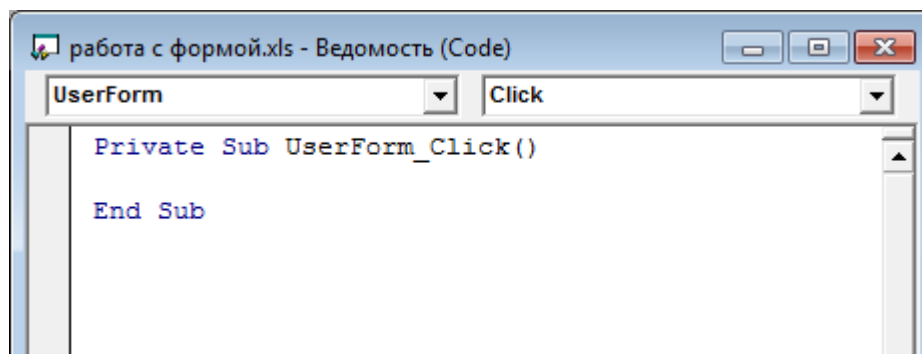


Рисунок 2.17 – Окно процедуры

8. Нам пока не нужна процедура UserForm Click. Которая вызывается после щелчка по форме. Нам нужна процедура, которая

будет выполняться при открытии формы. На языке программистов она называется процедурой, обрабатывающей событие Activate (Активизация). Список процедур, возможных для формы, находится в правой верхней части окна (рис. 17).

Щёлкнем по кнопке со стрелкой рядом со словом Click и в раскрывшемся списке выберем событие Activate. В окне программы появится заголовок процедуры UserForm_Activate.

9. Ввести код процедуры для записи исходных данных с листа *Список* в поля со списками и ввода данных:

```
Private Sub UserForm_Activate()  
With ComboBox1  
.RowSource = "Занятия"  
.ListIndex = 0  
End With  
With ComboBox2  
.RowSource = "файл"  
.ListIndex = 0  
End With  
TextBox1 = Format(Now, "dd/mm/yy")  
End Sub
```

10. Нажать клавишу F5 для выполнения процедуры. Отобразится экранная форма с заполненными списками *Тема занятия* и *Имя файла*, а также заполненным полем *Дата занятия* (рисунок 2.18).

Рисунок 2.18 – Ведомость

2.2 Создание процедуры для вызова формы

Для создания процедуры вызова экранной формы нужно воспользоваться командой Visual Basic «вывести на экран» - Show. Для этого следует выполнить следующие действия:

11. Перейти в редактор Visual Basic (если вы не находитесь в нём) и вставить новый модуль в рабочую книгу (Вставка – Модуль)/(Insert-Module).

12. Создать новую процедуру (Вставка – Процедура)/ (Insert-Procedure) и назвать её *Загрузка_формы*.

13. Ввести код процедуры.

```
Public Sub Загрузка_формы()  
    Введомость.Show  
End Sub
```

14. Перейти в рабочую книгу на Лист1 и вставить командную кнопку:

- Вкладка Разработчик (группа элементы управления) /Вставить/Кнопка (рисунок 2.19)

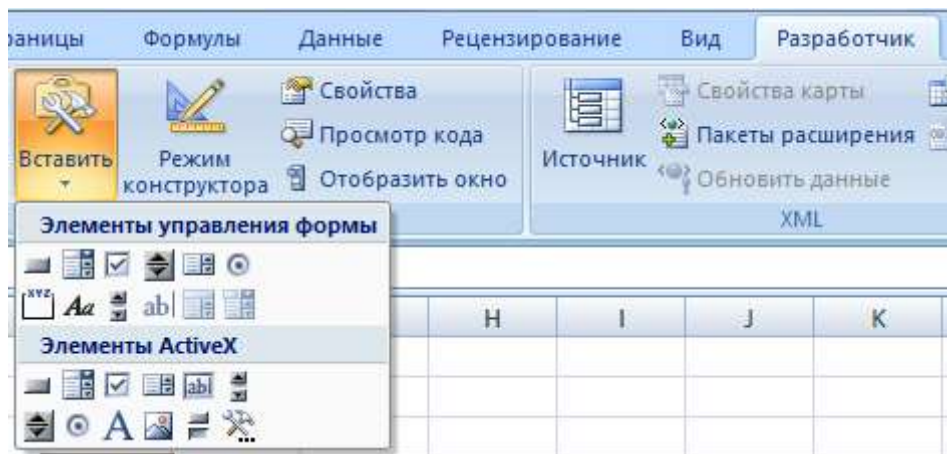


Рисунок 2.19 – Элементы управления

- Разместить кнопку на рабочем Листе1

15. После вставки кнопки откроется диалоговое окно Назначить макрос объекту. Выбрать в открывшемся списке Имя макроса пункт *Загрузка_формы*(рисунок 2.20).

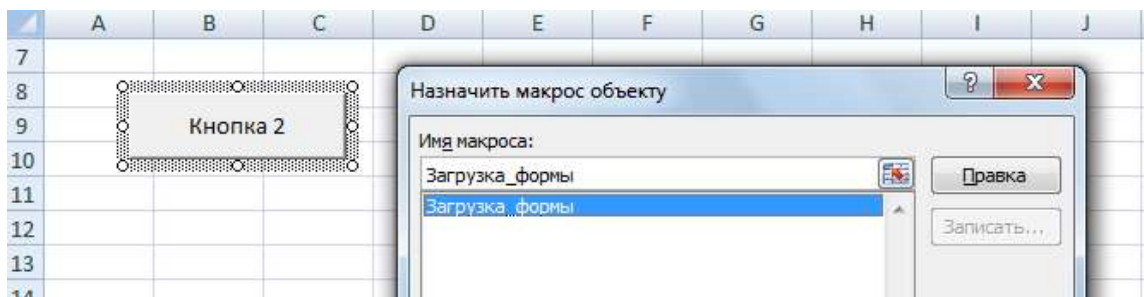


Рисунок 2.20 – Загрузка формы

16. Щёлкнуть по командной кнопке, стереть имя Кнопка 2 и дать кнопке имя *Ведомость работ*. После этого следует щёлкнуть вне кнопки, чтобы снять с неё выделение.

17. Щёлкнуть по этой командной кнопке. Откроется наша форма. Обратите внимание: списки *Темы занятий* и *Имя Файла* заполнены, как и поле *Дата занятий*.

18. Закрывать форму.

2.3 Управление флажком *Исходные данные взять с носителя*

При установке флажка *Исходные данные взять с носителя* надо снять блокировку с поля ввода Имя файла. За подобные действия должна отвечать процедура, вызываемая событием Change (Изменить). Это событие генерируется Visual Basic всякий раз. Когда изменяется значение свойства *Value* (Значение) какого-либо элемента. Выполните следующую последовательность действий:

1. Дважды щёлкните по флажку *Исходные данные взять с носителя*, чтобы открыть окно кода программы.

2. В списке событий в верхней части окна кода выберите событие *Change* и создайте процедуру

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

3. Перейдите в рабочую книгу на Лист1 и щёлкните по командной кнопке Ведомость работ. Отобразится форма (см. рисунок 2.19).

4. Установите флажок *Исходные данные взять с носителя*. Поле ввода *Имя файла на носителе* станет доступным (рис.21).

Ведомость лабораторных работ

Дата занятия: 17.10.12

Фамилия И.О. студента: []

Факультет: []

Форма обучения:

- Очная
- Вечерняя
- Заочная

Данные по лаб. работе

Тема занятия: [Операционная система Windows]

Исх. данные взять с носителя

Имя файла на носителе: [Список абонентов]

Сохранить Отмена

Рисунок 2.21 – Ведомость

5. Сохраните рабочую книгу.

2.4 Сохранение данных, введённых в форму

После того, как пользователь ввёл всю нужную информацию в форму, он щелкает по кнопке Сохранить. Процедура, которая запускается на выполнение после щелчка по этой кнопке, должна проверить правильность введённых данных и в случае их корректности записать данные в рабочий лист.

2.4.1 Создание процедуры проверки правильности ввода данных

Поле ввода может принять любую информацию. Однако необходимо проследить, чтобы пользователь не забыл ввести сведения в

поля ввода *Фамилия И.О. студента*. Для создания такой процедуры необходимо выполнить следующие действия:

1. В редакторе Visual Basic отобразите разрабатываемую форму.

2. Дважды щёлкните по кнопке Сохранить. В окне кода появится заголовок процедуры CommandButton1_Click. Опишем код процедуры:

3.

```
Private Sub CommandButton1_Click()  
If TextBox2 = "" Then  
MsgBox ("Пожалуйста, введите фамилию И.О. студента")  
TextBox2.SetFocus  
Exit Sub  
End If  
If TextBox3 = "" Then  
MsgBox ("Пожалуйста, введите название факультета")  
TextBox3.SetFocus  
Exit Sub  
End If  
Unload Me  
End Sub
```

3. Перейдите в рабочую книгу и щёлкните по командной кнопке, открывающей форму.

4. Щёлкните по кнопке *Сохранить*. Появится окно с сообщением «Пожалуйста, введите Фамилию И.О. студента».

5. Щёлкните по кнопке ОК, чтобы закрыть окно сообщения. На экране наша форма. Курсор находится в поле ввода *Фамилия И.О. студента*. Введите в это поле следующие данные: Иванов И.И. и щёлкните по кнопке *Сохранить*.

6. Появится окно с сообщением Пожалуйста, введите название факультета.

7. Щёлкните по кнопке ОК, чтобы закрыть окно сообщения. На экране наша форма. Курсор находится в поле ввода *Факультет*. Введите в это поле слово *Экономический* щёлкните по кнопке *Сохранить*. Форма исчезнет с экрана.

2.4.2 Процедура сохранения вводимых данных

Данные, введённые в форму, следует сохранить в рабочем листе книги. Для этого:

1. Перейдите на рабочий Лист 3 и Дайте ему имя «Форма».

2. В первой строке листа создайте строку заголовков, повторяющих названия элементов управления. За образец возьмите рабочий лист, показанный на рисунок 2.22.

	A	B	C	D	E	F
	Дата	Фамилия И.О. студента	факультет	Форма	Тема занятия	Имя файла
1	04.09.2012	Иванов ИИ.	Экономический	очная	СУБД Access	Список абонентов
2						

Рисунок 2.22 – Пример рабочего листа

3. Перейдите в редактор Visual Basic и отобразите экранную форму.

4. Чтобы открыть окно процедуры CommandButton1_Click, дважды щелкните по кнопке *Сохранить*.

5. Введите в процедуру CommandButton1_Click сразу после имени процедуры три оператора описания:

```
Dim форма As Object
Dim I As Integer
Set форма = Sheets("форма")
```

6. Теперь в текст той же процедуры перед оператором Unload Me введите следующий код:

7.

```
Worksheets("форма").Activate
For I = 1 To 100
If форма.Cells(I, 1) = "" Then
форма.Cells(I, 1) = TextBox1.Value
форма.Cells(I, 2) = TextBox2.Value
форма.Cells(I, 3) = TextBox3.Value
If OptionButton1 = True Then
форма.Cells(I, 4) = "Очная"
ElseIf OptionButton2 = True Then
форма.Cells(I, 4) = "Вечерняя"
ElseIf OptionButton3 = True Then
форма.Cells(I, 4) = "Заочная"
End If
форма.Cells(I, 5) = ComboBox1.Value
If CheckBox1.Value = True Then форма.Cells(I, 6) = ComboBox2.Value
Exit For
End If
Next
CheckBox1.Value = False
TextBox2 = ""
TextBox3 = ""
```

7. Удалите оператор Unload Me

8. Теперь процедура, выполняющаяся при щелчке по кнопке

Сохранить, выглядит так:

```
Private Sub CommandButton1_Click()  
Dim Форма As Object  
Dim I As Integer  
Set Форма = Sheets("Форма")  
If TextBox2 = "" Then  
MsgBox ("Пожалуйста, введите Фамилию И.О. студента")  
TextBox2.SetFocus  
Exit Sub  
End If  
If TextBox3 = "" Then  
MsgBox ("Пожалуйста, введите название факультета")  
TextBox3.SetFocus  
Exit Sub  
End If  
Worksheets("Форма").Activate  
For I = 1 To 100  
If Форма.Cells(I, 1) = "" Then  
Форма.Cells(I, 1) = TextBox1.Value  
Форма.Cells(I, 2) = TextBox2.Value  
Форма.Cells(I, 3) = TextBox3.Value  
If OptionButton1 = True Then  
Форма.Cells(I, 4) = "Очная"  
ElseIf OptionButton2 = True Then  
Форма.Cells(I, 4) = "Вечерняя"  
ElseIf OptionButton3 = True Then  
Форма.Cells(I, 4) = "Заочная"  
End If  
Форма.Cells(I, 5) = ComboBox1.Value  
If CheckBox1.Value = True Then Форма.Cells(I, 6) = ComboBox2.Value  
Exit For  
End If  
Next  
CheckBox1.Value = False  
Unload Me  
End Sub
```

9. Для того, чтобы при нажатии кнопки Отмена происходил выход из формы без сохранения исходных данных, щелкните по ней на форме и создайте процедуру:

```
Private Sub CommandButton2_Click()  
Unload Me  
End Sub
```

10. Перейдите в рабочую книгу на Лист

11. Сохраните рабочую книгу.

12. Щёлкните по кнопке, вызывающей экранную форму. Введите в пате Фамилия И.О. студента фамилию Иванов П.П. в поле Факультет- *ФГУСЭС*. Установите переключатель Форма обучения в положение *Очная*, в поле со списком Тема занятия выберите «СУБД Access». Установите флажок Исходные данные взять с дискеты в положение *Включено*, а в поле со списком *Имя файла на дискете* – *Список абонентов* Щёлкните по кнопке Сохранить.

13. Форма закроется, а введённые в неё данные будут записаны в ячейки рабочего листа Формы.

14. Повторите ввод данных в форму (для другого студента) и убедитесь, что данные о нём будут записаны в следующей строке таблицы.

Изучив основы работы с VBA, проведите работу с документом в соответствии со своим вариантом.

Форма документа, созданная с использованием VBA, должна давать возможность пользователю добавлять данные в документ, сохранять их, а также содержать кнопку «Отчистить».

Проект также должен выключать форму, позволяющую работать с реестром документов.

По желанию могут быть добавлены проверки на корректность введенных данных.

3 РАБОТА С ДОКУМЕНТОМ В MS ACCESS

Реляционная модель данных была предложена Е. Ф. Коддом (Dr. E.F.Codd), известным исследователем в области баз данных, в 1969 году.

Реляционная база данных представляет собой хранилище данных, содержащее набор двухмерных таблиц.

Система управления базами данных Access (СУБД Access) входит в стандартный набор прикладных программ пакета Microsoft Office, который – так сложилось исторически – используется практически в каждой организации нашей страны.

Создание базы данных, это:

- 1) теоретическое продумывание, проектирование структуры будущих таблиц и их взаимных связей в базе данных,
- 2) создание структуры этих таблиц на компьютере,
- 3) установление связей таблиц базы данных,
- 4) заполнение таблиц реальными данными (с использованием форм).

Использование базы данных – это создание для практических целей запросов и отчетов.

Назначение любой СУБД – создание, ведение и обработка баз данных. СУБД предоставляет значительные возможности по работе с хранящимися данными, их обработке и совместному использованию.

В СУБД Access предусмотрено много дополнительных сервисных возможностей. *Мастера* помогут создать таблицы, формы или отчеты из имеющихся заготовок. *Выражения* используются в Access, например, для проверки допустимости введенного значения. *Макросы* позволяют автоматизировать многие процессы без программирования, тогда как встроенный в Access язык VBA (Visual Basic for Applications) дает возможность опытному пользователю программировать сложные процедуры обработки данных.

Структура таблицы и типы данных

При разработке структуры таблицы, прежде всего, необходимо определить названия полей, из которых она должна состоять, типы полей и их размеры. Каждому полю таблицы присваивается уникальное имя, которое не может содержать более 64 символов. Имя

желательно делать таким, чтобы функция поля узнавалась по его имени. Далее надо решить, данные какого типа будут содержаться в каждом поле. В MS Access можно выбирать любые из основных типов данных. Один из этих типов данных должен быть присвоен каждому полю. Значение типа поля может быть задано только в режиме конструктора.

В Access существует несколько способов создания пустой таблицы:

- 1) **мастер** таблиц;
- 2) ввод данных непосредственно в пустую таблицу в **режиме таблицы**;
- 3) определение всех параметров макета таблицы в **режиме конструктора**.

Объекты базы данных

Объектами базы данных являются:

Таблицы - совокупность записей, где хранится основная информация.

Форма представляет собой специальный формат экрана, используются для ввода данных в таблицу и просмотра одной записи.

Запрос – это инструмент для анализа, выбора и изменения данных. С помощью Access могут создаваться несколько видов запросов.

Отчеты – это средство организации данных при выводе на печать.

Из всех типов объектов только таблицы предназначены для хранения информации. Остальные используются для просмотра, редактирования, обработки и анализа данных – иначе говоря, для обеспечения эффективного доступа к информации. ктами базы данных являются:

Таблицы - совокупность записей, где хранится основная информация.

Форма представляет собой специальный формат экрана, используются для ввода данных в таблицу и просмотра одной записи.

Запрос – это инструмент для анализа, выбора и изменения данных. С помощью Access могут создаваться несколько видов запросов.

Отчеты – это средство организации данных при выводе на печать.

Из всех типов объектов только таблицы предназначены для хранения информации. Остальные используются для просмотра, редактирования, обработки и анализа данных – иначе говоря, для обеспечения эффективного доступа к информации.

Типы данных

Текстовый – наиболее чисто используемый в Access тип данных. Этот тип данных подходит для хранения адресов, для полей с кратким описанием, для числовых данных, не требующих расчетов, таких, как телефонные номера и почтовые индексы. Длина – 255 символов.

Поле Мемо – предназначен для полей, длина которых превосходит 255 символов. Пример: длинное поле описания. Поле Мемо может хранить до 65 535 символов, что приблизительно равно 32 страницам текста.

Числовой. Данные, используемые для математических вычислений, за исключением финансовых расчетов (для них следует использовать тип «Денежный»).

Дата/время. Значения дат и времени. Сохраняет 8 байтов. Можно вводить даты с 1 января 100 года по 31 декабря 9999 года. Access предлагает несколько различных форматов дат.

Денежный. Используется для денежных значений и для предотвращения округления во время вычислений, для выполнения вычислений над полем, которое содержит числа, в левой части которых не более 15 знаков, а справа от запятой не более четырех знаков.

Счетчик. Автоматическая вставка уникальных последовательных (увеличивающихся на 1) или случайных чисел при добавлении записи с использованием этого типа данных либо, выбрав соответствующий пункт в свойстве **Новое значение** этого поля. Если удалить одну из последовательных записей, этот тип поля не запомнит и не перенумерует удаленное значение. Это значение будет просто отсутствовать.

Логический (Да/нет). Данные, принимающие только одно из двух возможных значений, таких как «Да/Нет», «Истина/Ложь», «Вкл/Выкл». Значения Null не допускаются.

Поле объекта OLE. Объекты OLE (такие как документы Microsoft Word, электронные таблицы Microsoft Excel, рисунки,

звукозапись или другие данные в двоичном формате), созданные в других программах, использующих протокол OLE.

Гиперссылка. Гиперссылка может иметь вид пути UNC либо адреса URL.

Мастер подстановок. Создает поле, позволяющее выбрать значение из другой таблицы или из списка значений, используя поле со списком.

Несовместимые данные

Необходимо соблюдать для одних наименований полей данных одинаковый тип данных.

Выполним последовательно следующие задания для того, чтобы ознакомиться с особенностями работы СУБД MS Access и иметь в будущем возможность самостоятельно выполнить задание по своему документу.

Задание 1. Создание однотабличной базы данных

Цель работы: Приобрести навыки и умения при работе с таблицами: создавать таблицы с помощью конструктора, задавать тип данных, задавать маску ввода для поля, использовать свойства поля, создавать ключи и индексы для полей таблицы, заполнять таблицы данными и создавать формы.

Этапы работы

1.1. Для создания новой базы данных: загрузите MS Access, в появившемся окне выберите пункт **Новая база данных**;

Задайте имя вашей базы. По умолчанию MS Access предлагает вам имя базы *db1*(Access 2007 – *Database1.accdb*), а тип файла – *Базы данных Access*. Имя задайте *Стоматология*.

1.2. Для создания таблицы базы данных: в окне базы данных выберите вкладку *Создание*, а затем щелкните по кнопке <Конструктор таблиц> (рисунок 3.1).

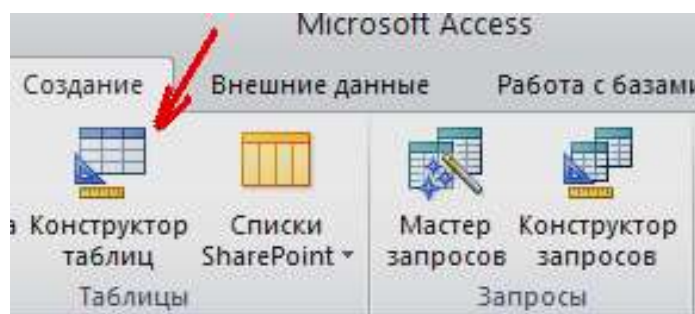


Рисунок 3.1 – Создание таблиц в режиме Конструктора

1.3. Для определения полей таблицы: введите в строку столбца **Имя поля** имя первого поля – *ID врача*.

В строке столбца «Тип данных» щелкните по кнопке списка и выберите тип данных *Числовой*. Поля вкладки *Общие* оставьте такими, как предлагает Access (рисунок 3.2).

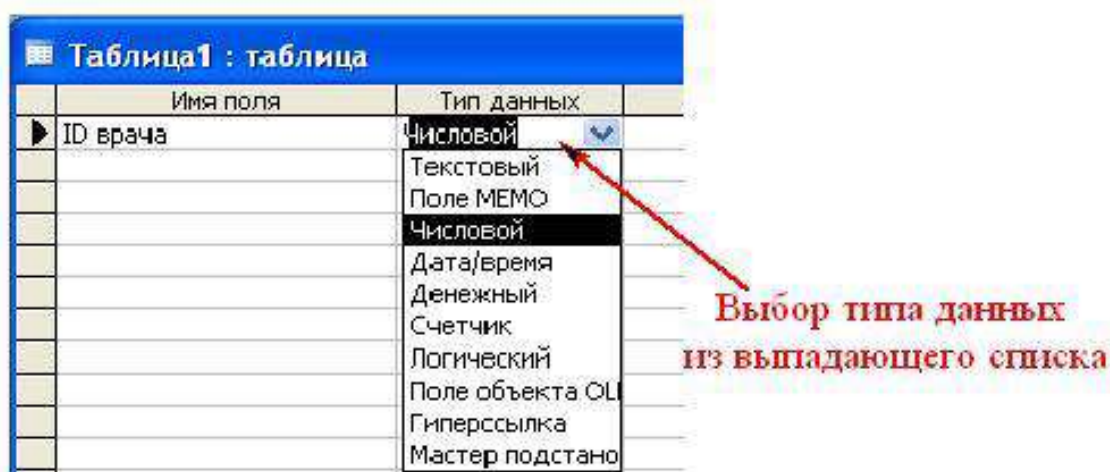


Рисунок 2 – Выбор типа данных

Для определения всех остальных полей таблицы базы данных *Врачи* выполните действия, аналогичные указанным выше в соответствии с таблицей 3.1.

Таблица 3.1 – Описание свойств полей таблицы *Врачи*

Имя поля	Тип данных	Размер поля (вкладка Общие)
ID врача	Числовой	
ФИО	Текстовый	20
Должность	Текстовый	20
№ кабинета	Числовой	Длинное целое (по умолчанию)
Время приема	Текстовый	12
ВидОказываемыхУслуг	Текстовый	50 (Максимальное значение)

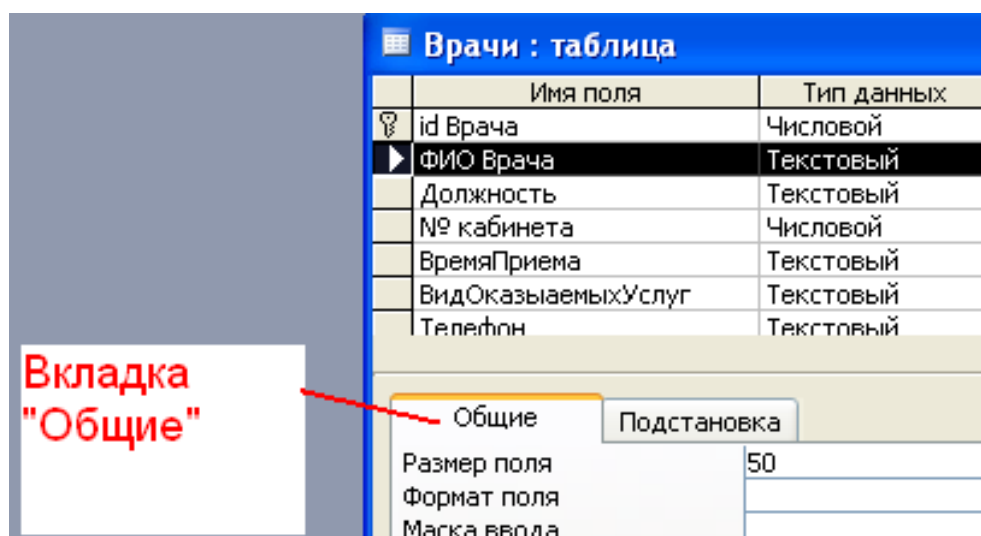
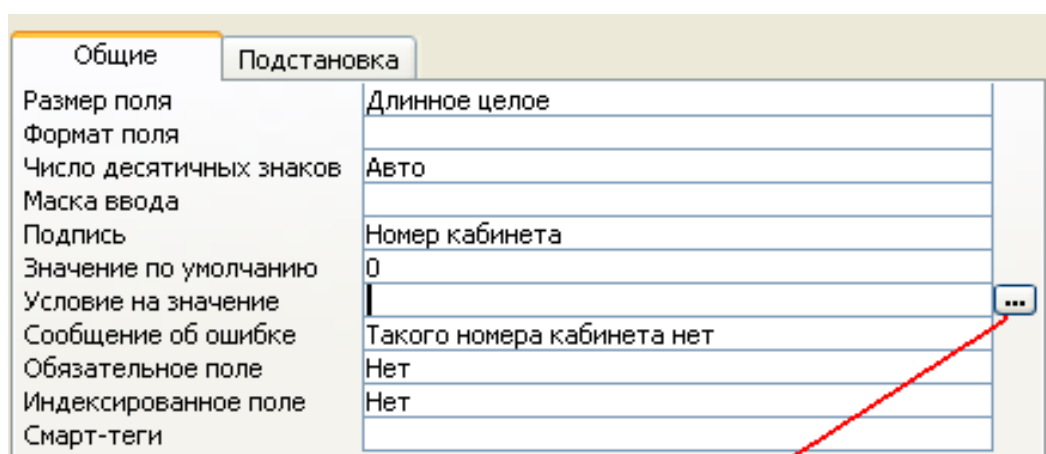


Рисунок 3.3 – Использование для поля дополнительных параметров с помощью вкладки «Общие»

1.4. Задайте условие для поля «№ кабинета».

Пусть по условию в поликлинике 2 этажа и значение номера кабинета находится в пределах от 100 до 130 и от 200 до 230. В строке «Условие на значение» перейдите в режим **Построителя выражений** (рис. 4) и введите условие: $\geq 100 \text{ And } \leq 130 \text{ Or } \geq 200 \text{ And } \leq 230$ (рисунок 3.5).

В строке «Сообщение об ошибке» введите: «Такого номера кабинета нет».



Кнопка перехода в режим "Построителя выражений"

Рисунок 3.4 – Кнопка перехода в режим «Построителя выражений»

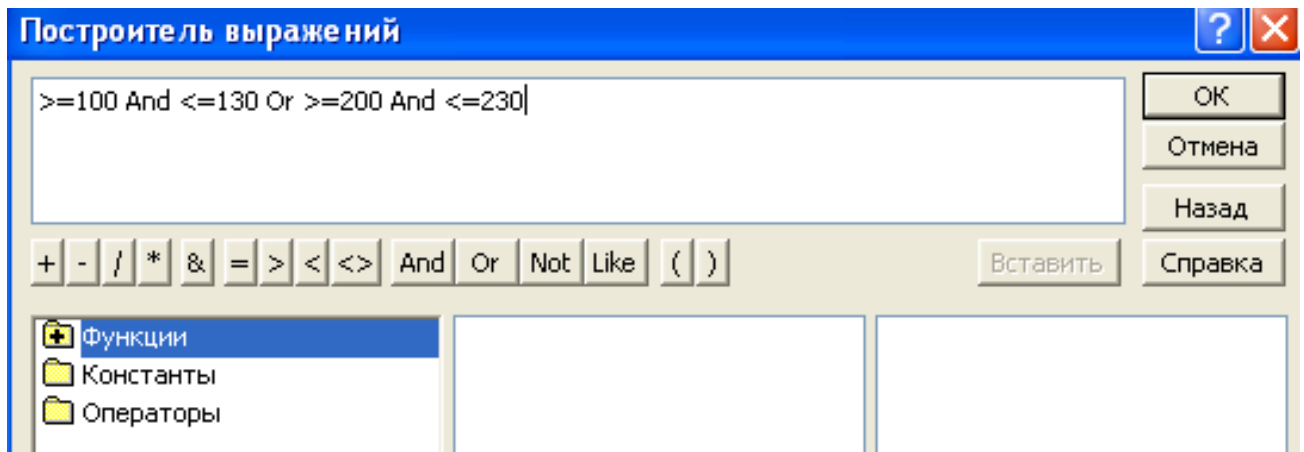


Рисунок 3.5 – Окно «Построителя выражений»

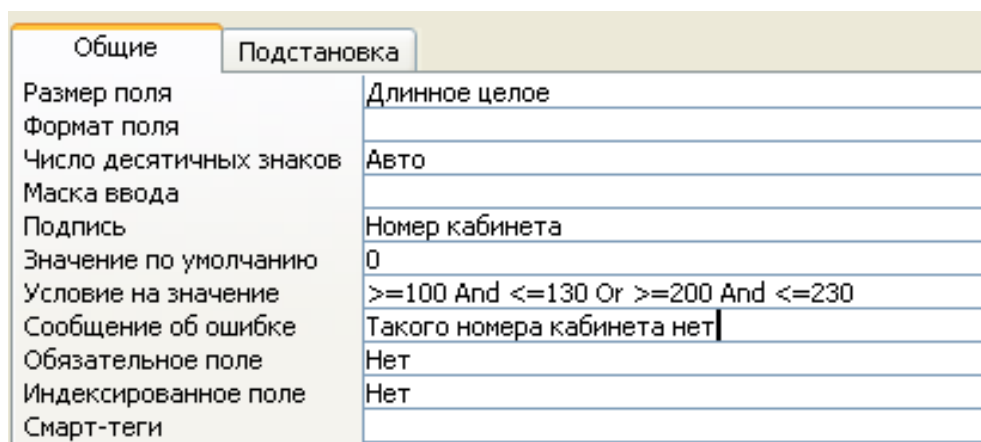


Рисунок 3.6 – Вкладка «Общие» с заданными условиями для поля «№ кабинета»

1.5. Создание ключевых полей.

Первичный ключ – это поле (или ряд полей), которое однозначно идентифицирует каждую запись в таблице. Access не допускает повторных значений в поле первичного ключа.

Сделайте поле **ID врача** ключевым. Для этого откройте таблицу *Врачи* в режиме **Конструктора** и, поместив курсор на имя этого поля, щелкните по кнопке **Ключевое поле** (рисунок 3.7):

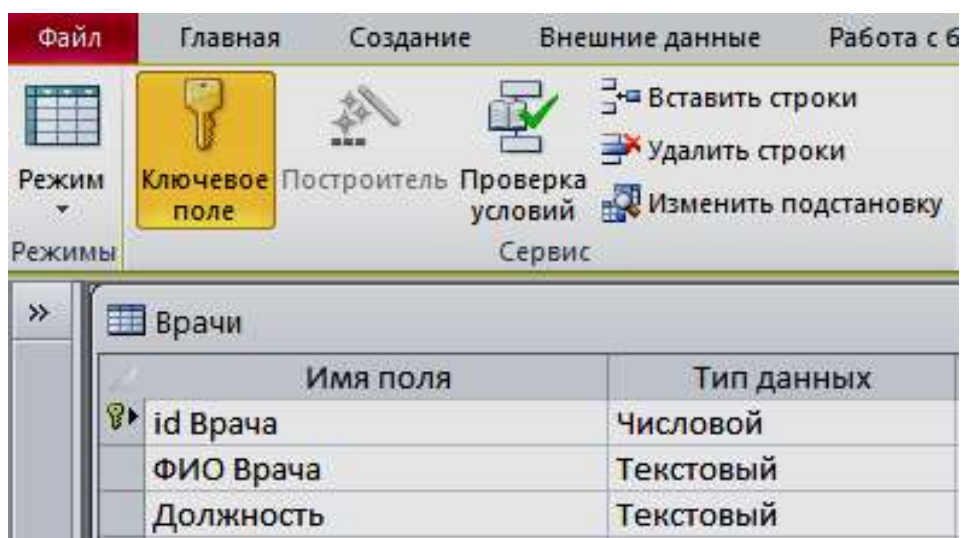


Рисунок 3.7 – Назначение поля ключевым

Закройте таблицу. MS Access предложит сохранить таблицу (по умолчанию «Таблица1»). Задайте имя таблицы «Врачи».

Задание 2. Заполнение базы данных.

Этапы работы

2.1. Создание таблиц-подстановок.

Для упрощения ввода данных в таблицу «Врачи» создайте таблицы-подстановки для полей: «ВремяПриема» и «ВидОказываемыхУслуг». Для поля «Должность» данные будут подставляться из таблицы «Штатное расписание».

Подстановка упрощает режим ввода. При этом нет необходимости набирать длинные названия несколько раз. Достаточно будет выбрать необходимую запись из выпадающего списка.

2.1.1. Таблица «Смена-подстановка» (используется для поля «ВремяПриема»). Поля таблицы (таблица 3.2):

Таблица 3. – Описание свойств полей таблицы «Смена_подстановка»

Смена_подстановка : таблица	
Имя поля	Тип данных
НомерСмены	Текстовый
ВремяПриема	Текстовый

Примечание 1. В результате щелчка по кнопке <ОК> MS Access предложит вам задать ключевое поле. Для создаваемой таблицы щелкните по кнопке <Нет>.

2.1.2. Таблицу «*НаименованиеУслуги_подстановка*» (используется для поля «Вид оказываемых услуг») создайте в **Excel**, чтобы импортировать её в нашу базу данных. Поля таблицы (таблица 3.3):

Таблица 3.3 – Таблица с данными «*НаименованиеУслуги_подстановка*»

Наименование направления услуги
Терапевтическая
Хирургическая
Ортопедическая

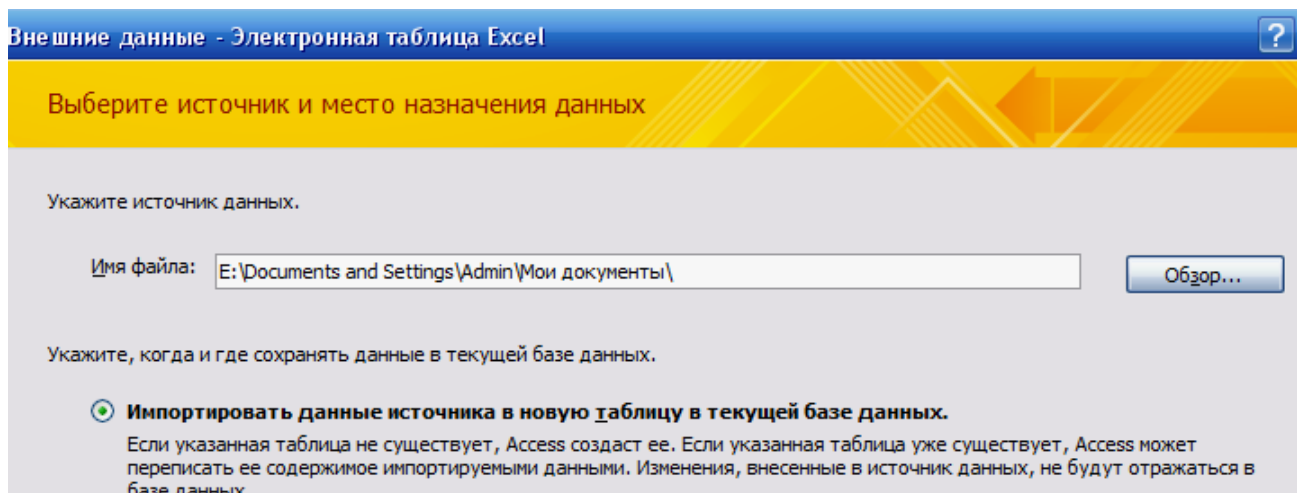
Импорт как самих таблиц, так и полей таблиц можно производить из разных источников и файлов разных форматов.

Определите формат ячеек: для поля *НаименованиеУслуги* – текстовый.

Сохраните таблицу в той же папке, где находится ваша БД *Стоматология*.

В базе данных *Стоматология* на панели инструментов выберите **Внешние данные Excel**.

В открывшемся диалоговом окне выберите созданную таблицу:



В следующем окне «Импорт электронной таблицы» выберите Лист 1, нажмите <Далее>.

установите флажок «*Первая строка содержит заголовки столбцов*», далее: «*Данные необходимо сохранить в новой таблице*», далее подтвердите описание и название полей, далее выберите «**Не создавать ключ**», Готово.

Таблица «НаименованиеУслуги_подстановка» будет импортирована в вашу БД. Посмотрите ее в режиме **Конструктора**. Все ли поля соответствуют **типу данных**?

2.1.3. Таблица «Штатное расписание» (данные поля «НаименованиеДолжности» будут использоваться для подстановки в таблицу «Врачи» (таблица 3.4).

Таблица 3.4 – Описание свойств полей таблицы «Штатное расписание»

Штатное Расписание : таблица	
Имя поля	Тип данных
Должность	Текстовый
Оклад	Денежный
Надбавка	Денежный

Для поля «Оклад» во вкладке **Общие** в строке «Формат поля» выбрать «Денежный».

Для поля «Должность» задать значение **Ключевое поле**.

Заполнение таблиц данными.

Для заполнения таблиц данными необходимо перейти из режима **Конструктора** в режим **Таблицы** или открыть таблицу двойным щелчком.

Внимание! (данные в таблицу можно ввести только в режиме **Таблицы**).

Заполните таблицу «Смена_подстановка» данными (рисунок 3.8).

Смена_подстановка	
НомерСмет	ВремяПрие
1	08.00-12.00
2	13.00-19.00

Рисунок 3.8 – Таблица «Смена_подстановка» с данными

Заполните таблицу «ШтатноеРасписание» данными (рисунок 3.9).

Штатное Расписание : таблица			
	Должность	Оклад	Надбавка
+	врач-стоматолог	7 000,00р.	700,00р.
+	ортопед	14 000,00р.	1 200,00р.
+	хирург-терапевт	10 000,00р.	1 000,00р.

Рисунок 3.9 – Таблица «ШтатноеРасписание» с данными

Создание подстановочного поля.

Для создания подстановочного поля используется три способа: режим **Конструктора**. Этот способ используется в том случае, если поле для подстановки стоит первым в подстановочной таблице, использование **Мастера подстановок**, или использование языка запросов SQL (с помощью построителя запросов).

1) Откройте таблицу «Врачи» в режиме **Конструктора**. Для поля «Должность» во вкладке **Подстановка** выберите в строке **Тип элемента управления**: «Поле со списком»; в строке **Тип источника строк**: «Таблица или запрос» (задается по умолчанию); в строке **Источник строк** из выпадающего списка выбрать таблицу «Штатное расписание» - то есть таблицу, из которой данные будут подставляться в таблицу «Врачи» (рисунок 3.10):

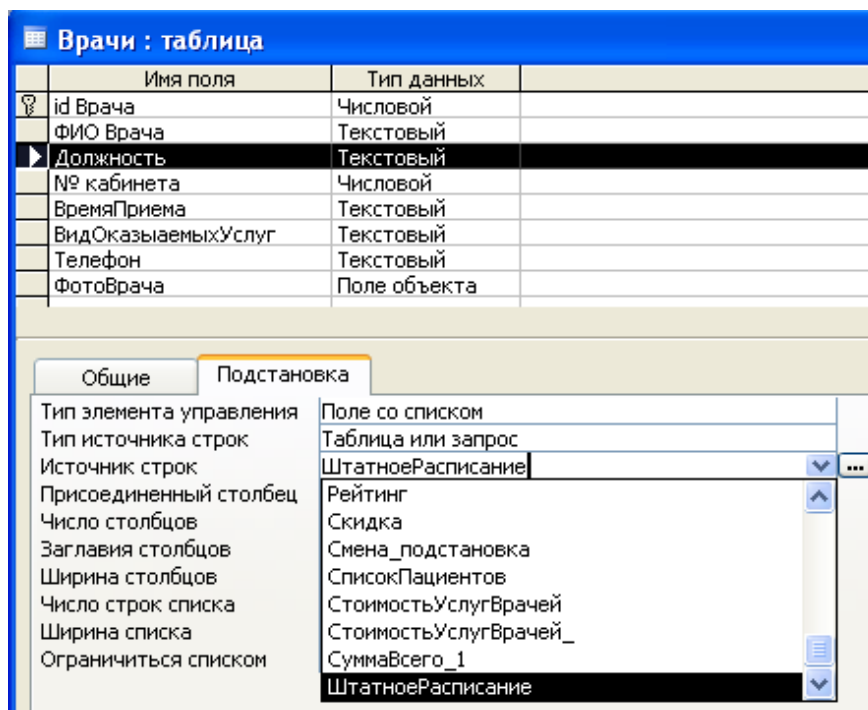


Рисунок 3.10 – Подстановка в режиме **Конструктора**

2) Для создания подстановки для поля «*ВремяПриема*» в столбце **Тип данных** выберите из выпадающего списка «**Мастер подстановки**». В следующем окне нажмите кнопку <Далее>, в открывшемся диалоговом окне выберите «*Таблица: Смена_подстановка*». Нажмите <Далее>.

В следующем окне «Создание подстановки» из списка «Доступные поля» выберите при помощи кнопки поле «*ВремяПриема*» (данные из этого поля будут подставляться в нашу таблицу), нажмите кнопку <Далее>, затем <Готово> (рисунок 3.11).

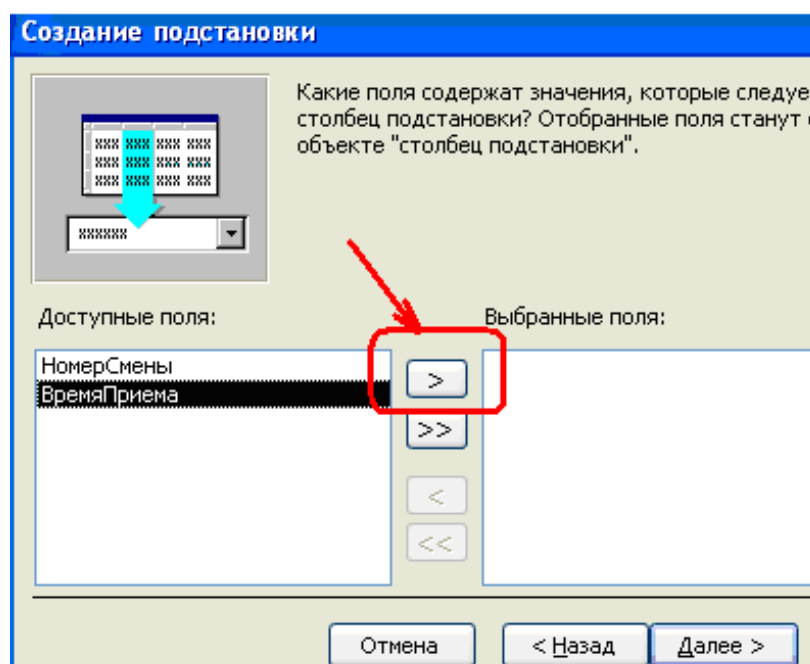


Рисунок 3.11 – Использование Мастера подстановки

3) Использование языка запросов SQL. Для использования построителя запросов для поля «*ВремяПриема*» выберите вкладку **Подстановка**: в строке **Тип элемента управления** - «*Поле со списком*», в строке **Тип источника строк** – «*Таблица или запрос*», в строке **Источник строк** нажмите на кнопку «...» **Построителя выражений** и в открывшемся диалоговом окне **Инструкция SQL: построитель запросов** на панели инструментов выберите: **Вид Режим SQL**.

Запрос будет иметь вид:

```
SELECT      Смена_подстановка.ВремяПриема      FROM
Смена_подстановка;
```

Самостоятельно. Для поля «ВидОказываемыхУслуг» создайте подстановочное поле.

2.4. Ввод ограничения на данные в поле «ID врача».

Для поля «ID врача» в таблице «Врачи» выполним условия: коды врачей не должны повторяться, должна быть обеспечена возможность их изменения (из-за последнего условия в этом поле нельзя использовать тип данных **Счетчик**, в котором данные не повторяются).

Для выполнения этих условий сделайте следующее: установите курсор на строке «ID врача»; во вкладке **Общие** щелкните по строке параметра **Индексированное поле**; выберите в списке пункт «Да (совпадения не допускаются)»; перейдите в режим **Таблица**.

Примечание 2. Индекс - это средство MS Access, ускоряющее поиск и сортировку данных в таблице.

Заполните таблицу «Врачи» данными, используя подстановку (пример на рисунке 3.12):

Использование режима
"Подстановка"

Табельный ном	ФИО Врача	Должность	Номер кабинё	ВремяПриема	ВидОказываемыхУслуг
100	Валиуллин Д.К.	ортопед	101	08.00-12.00	Ортопедическая
101	Галимянова О.А.	врач-стоматолог	102	08.00-12.00	Хирургическая
102	Дюкина Я.Б.	ортопед	203	08.00-12.00	Терапевтическая
103	Елизарова Е.П.	хирург-терапевт	115	08.00-12.00	Терапевтическая
104	Жуков А.Е.	врач-стоматолог	120	08.00-12.00	Терапевтическая
105	Карбышев Г.О.	ортопед	214	08.00-12.00	Ортопедическая
106	Леонидова М.Р.	врач-стоматолог	224	13.00-19.00	Терапевтическая
107	Накипов Н.Д.	хирург-терапевт	315	08.00-12.00	Хирургическая
108	Никоненко П.С.	ортопед	124	13.00-19.00	Ортопедическая

Рисунок 3.12 – Пример заполнения таблицы «Врачи» с использованием режима Подстановки

Задание 3. Размещение новых объектов в таблице.

3.1. Размещение объекта OLE.

Рассмотрим размещение объекта OLE в таблице «Врачи» на примере поля «ФотоВрача». В нашем примере фотографии хранятся в формате графического редактора **Paint** в файлах с расширением *bmp*. (Создайте несколько таких объектов и сохраните их в той же папке, где хранится база данных).

Откройте таблицу «Врачи» в режиме **Конструктора** и добавьте новое поле «*ФотоВрача*». Тип данных – **Поле объекта OLE**.

Перейдите в режим **Таблицы**. В графе «**Фото**» выполните команду правой кнопкой мыши: **Добавить объект**.

В окне **Вставка объекта** (рисунок 3.13) отметьте переключатель «*Создать из файла*». Окно **Вставка объекта** позволяет ввести имя файла с фотографией. Для поиска файла воспользуйтесь кнопкой **Обзор**.

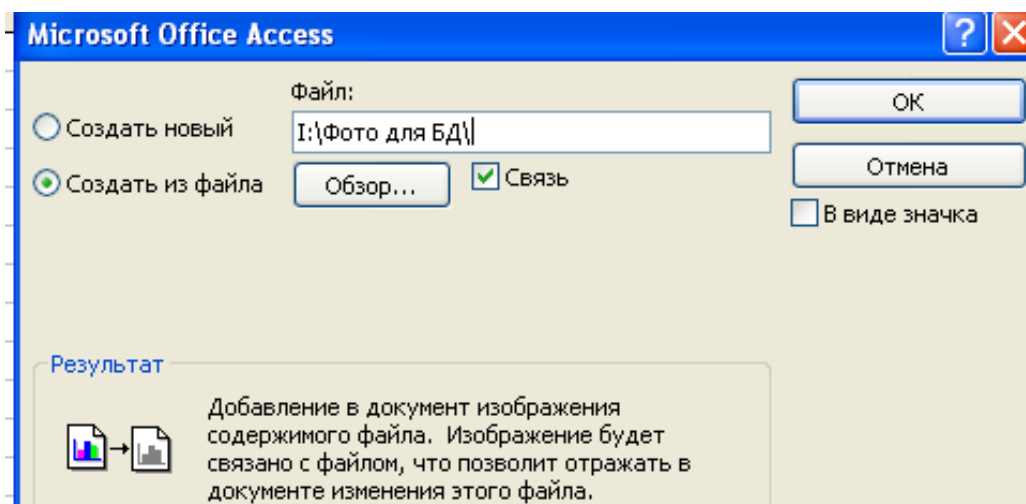


Рисунок 3.13 – Добавление объекта.

Для введения в поле связанного объекта надо установить флажок «*Связь*», что позволит в дальнейшем отражать в документе изменения этого файла

Добавление нового поля в таблицу. Создание маски ввода.

1. Откройте таблицу «*Врачи*» в режиме Конструктора.
2. Добавьте новое поле «*Телефон*». Тип данных – **Текстовый**.
3. Задайте маску ввода для поля **Телефон** (в виде 22-22-222), для этого:

Щелкните на вкладке **Общие** в строке *Маска ввода* по кнопке «...».

Подтвердите сохранение таблицы.

В открывшемся диалоговом окне «*Создание масок ввода*» щелкните по кнопке <Список>. В появившемся диалоговом окне «*Настройка масок ввода*» выполните действия по образцу (рисунок 3.14):

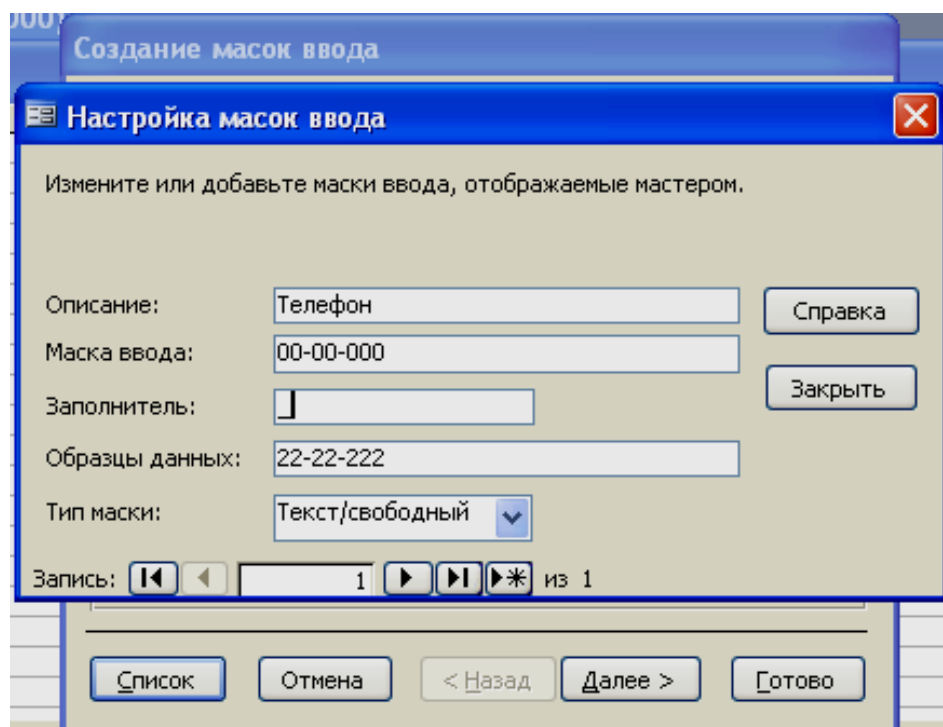


Рисунок 3.14 – Создание маски ввода

Нажмите кнопку <Закреть> и из появившегося списка выберите созданную маску ввода для поля *Телефон*.

Примечание 3. Маска ввода создается только для поля, имеющего тип данных – **текстовый!**

Таблица 3.5 – Некоторые допустимые знаки масок ввода

Знак	Описание
0	Цифра (от 0 до 9, ввод обязателен; знаки плюс [+] и минус [-] не допускаются).
9	Цифра или пробел (ввод не обязателен; знаки плюс и минус не допускаются).
#	Цифра или пробел (ввод не обязателен; пустые знаки преобразуются в пробелы, допускаются знаки плюс и минус).
L	Буква (от А до Z или от А до Я, ввод обязателен).
?	Буква (от А до Z или от А до Я, ввод не обязателен).
A	Буква или цифра (ввод обязателен).
&	Любой знак или пробел (ввод обязателен).

Фильтрация данных по полям

В режиме **Таблица** в таблице «*Врачи*» щелкните по записи *Ортопед* поля «Должность»;

1. щелкните по кнопке или выполните команду **Записи, Фильтр, Фильтр по выделенному**. В таблице останутся только записи о врачах – ортопедах;

Табельный ном	ФИО Врача	Должность	Номер кабинета	ВремяПриема
100	Валиуллин Д.К.	ортопед	101	08.00-12.00
101	Галимянова О.А.	хирург-терапевт	102	08.00-12.00

Рисунок 3.15 – Фильтр данных

Для отмены фильтрации щелкните по кнопке на панели инструментов или выполните команду **Записи, Удалить фильтр**. В таблице появятся все данные.

Задание 4. Создание новых таблиц.

Создайте таблицу **«СписокПациентов»**, которая содержит следующие поля:

Имя поля	Тип данных
НомерКарточки	Числовой
ФИО Пациента	Текстовый
ДомашнийАдрес	Текстовый
МедПолис	Текстовый
Телефон	Текстовый
Пол	Текстовый

1) Для поля **«МедПолис»** создайте маску ввода по образцу: **«АБ-1234»**.

2) Для поля **«Телефон»** создайте маску ввода как в задании 3.

3) Для поля **«Пол»** во вкладке **Общие** в строке **«Значение по умолчанию»** введите: *муж*. Это значение будет указано по умолчанию. В случае необходимости его можно заменить.

4) Заполните таблицу данными (не менее 15 записей).

Создайте таблицу **«НаименованиеУслуг»**, которая содержит следующие поля:

НаименованиеУслуг	
Имя поля	Тип данных
№ Услуги	Числовой
НаименованиеУслуги	Текстовый
ХарактеристикаУслуги	Текстовый
Стоимость	Денежный
Скидка	Числовой

Заполните таблицу данными по примеру:

НаименованиеУслуг				
№ Услуги	НаименованиеУслуги	ХарактеристикаУслуг	Стоимость	Скидка
1	Лечение однокорневого пульпита	Терапевтическая	1 250,00р.	3
2	Лечение двухкорневого пульпита	Терапевтическая	1 650,00р.	1
3	Лечение трехкорневого пульпита	Терапевтическая	2 000,00р.	0
4	Лечение кариеса	Терапевтическая	500,00р.	5
5	Светополимерная композитная пломба (Рос	Терапевтическая	350,00р.	0
6	Лечение периодонтита	Терапевтическая	1 500,00р.	5
7	Удаление зубного камня – (один зуб)	Хирургическая	60,00р.	0
8	Чистка всех зубов одной челюсти (ультразву	Хирургическая	800,00р.	0
9	Профессиональное отбеливание - (один зуб)	Хирургическая	300,00р.	3
10	Удаление зуба	Хирургическая	150,00р.	0

Создайте таблицу «*Журнал*», которая содержит следующие поля:

Журнал : таблица		
Имя поля	Тип данных	
ДатаПриема	Дата/время	
НомерКарточки	Числовой	
ФИО Пациента	Текстовый	
ФИО Врача	Текстовый	
ДолжностьВрача	Текстовый	
Услуги	Текстовый	
Количество	Числовой	

Общие Подстановка
 Формат поля Краткий формат даты

1) Для поля «*НомерКарточки*» используйте подстановку:
 SELECT СписокПациентов.НомерКарточки, СписокПациентов.[ФИО Пациента] FROM СписокПациентов;

2) Поля «*ФИО пациента*» и «*Должность врача*» не заполняйте. Для их заполнения будет использован **запрос на обновление**. (Задание № 11)

3) Для поля «ФИО врача» используйте подстановочное поле из таблицы «Врачи».

4) Для поля «Услуги» используйте подстановку из таблиц услуг (таблица «НаименованиеУслуг»).

Задание 5. Ввод и просмотр данных посредством формы.

1) Создать формы для ввода данных с последующей их модификацией.

2) Создать главную форму «Заставка».

Этапы работы

Формы являются основным средством организации интерфейса пользователя в приложениях Access.

Для создания формы:

выберите таблицу, для которой будет создана форма, зайдите во вкладку **Создание**, выберите **Форма**.

Автоматически созданная форма откроется в главном окне. Access предложит сохранить форму по названию таблицы.

Создавать и редактировать формы любой степени сложности позволяет только **Конструктор** форм.

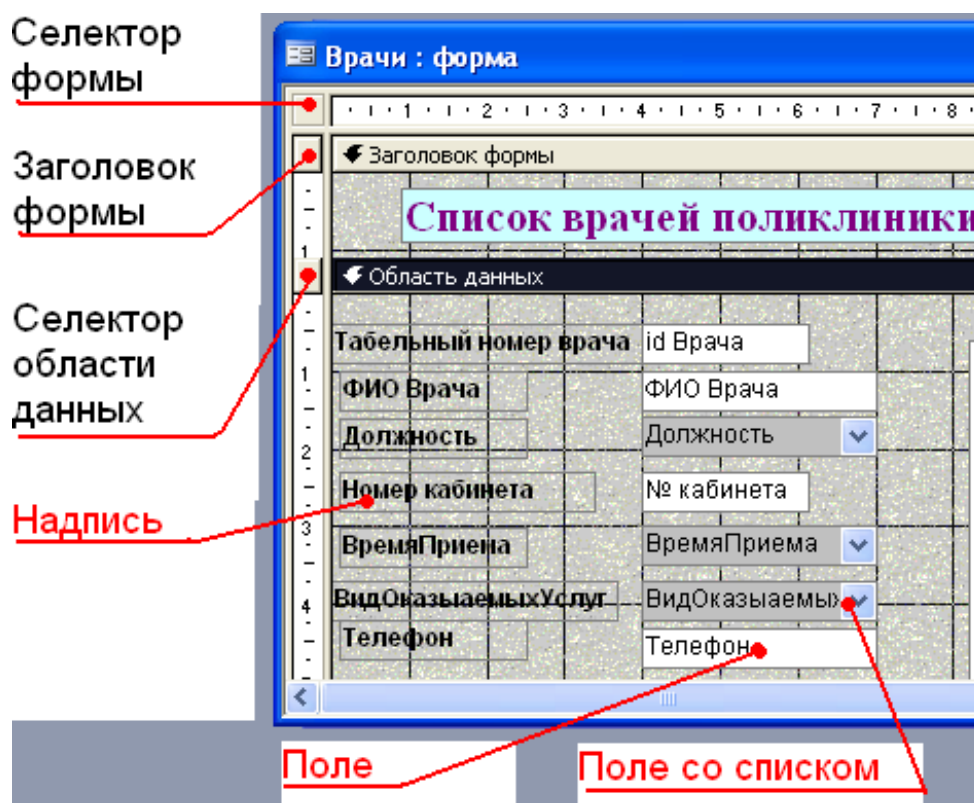





Рисунок 3.16 – Форма в режиме **Конструктора**

Панель элементов и Список полей

Панель элементов появляется в режиме **Конструктора** форм и используется для размещения объектов в форме.. Ниже приведены наиболее используемые элементы формы и соответствующие им кнопки на Панели элементов.


 – **Надпись**. Размещение в форме произвольного текста.


 – **Поле**. Размещение в форме данных из соответствующего поля базовой таблицы/запроса, вывод результатов вычислений, а также прием данных, вводимых пользователем.


 – **Выключатель**. Создание выключателя, кнопки с фиксацией.


 – **Переключатель**. Создание селекторного переключателя.

 – **Флажок**. Создание контрольного переключателя.


 – **Поле со списком**. Размещение элемента управления, объединяющего поле и раскрывающийся список.

 – **Список**. Создание списка, допускающего прокрутку. В режиме формы выбранное из списка значение можно ввести в новую запись или использовать для замены уже существующего значения.

 – **Кнопка**. Создание командной кнопки, позволяющей осуществлять разнообразные действия в форме (поиск записей, печать отчета, установка фильтров и т. п.).

 – **Рисунок**. Размещение в форме рисунка, не являющегося объектом OLE.

 – **Подчиненная форма/отчет**.

 – **Линия (Прямоугольник)** – элементы оформления. Размещение в форме линии для отделения логически связанных данных.



– **Свободная рамка объекта** - любой объект Windows-приложений, редактирование которого будет возможно вызовом соответствующего приложения.

Для создания главной кнопочной формы создайте управляющие кнопки

Кнопки используются в форме для выполнения определенного действия или ряда действий. Например, можно создать кнопку, которая будет открывать таблицу, запрос или другую форму. Можно создать набор кнопок для перемещения по записям таблицы.

На панели инструментов выберите вкладку **Создание Конструктор**

форм. Появится пустая форма. Задайте мышкой размеры формы.

Откроется вкладка **Конструктор** - панель элементов, которая позволяет создавать элементы управления на форме и осуществлять необходимые действия при конструировании (рисунок 3.17):

Выберите на панели инструментов и активируйте **Кнопку**.

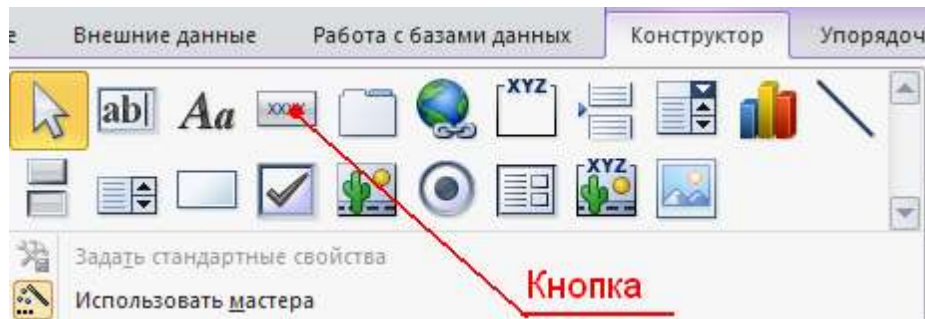


Рисунок 3.17 – Элементы управления

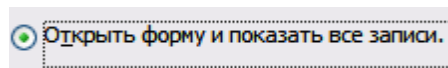
Создайте заголовок формы. Для этого выберите кнопку – **Надпись**, щелкнув по ней, расширьте область заголовка формы и введите в поле надпись **База данных «Врачи»**. Измените размер и цвет шрифта.

Выберите на панели инструментов **Кнопку**. Щелкните мышкой по тому месту в области данных, где должна быть кнопка. Появится диалоговое окно **Создание кнопок**.

Выберите категорию **Работа с формой**, а действие – **Открыть формы**.

Нажмите кнопку <Далее>. Выберите форму *Врачи*, которая будет открываться этой кнопкой, нажмите кнопку <Далее>.

Далее, оставьте переключатель в положении:



В следующем окне поставьте переключатель в положение **Текст**, наберите *Список врачей*. <Далее>.

Задайте имя кнопки *Список* и нажмите <Готово>.

САМОСТОЯТЕЛЬНО: Добавьте кнопку выхода или закрытия главной формы.

Изменить макет элементов управления можно, щелкнув правой кнопкой мыши по элементу и выбрав **Свойства** в режиме **Конструктор** формы:

При этом открывается диалоговое окно свойств элементов управления (рисунок 3.18):

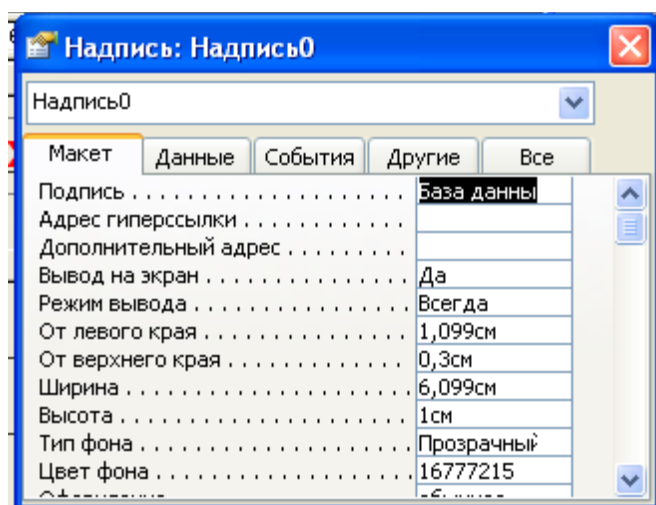


Рисунок 3.18 – Окно свойств полей данных

Для того, чтобы на форме «Заставка» убрать полосы прокрутки, выполните следующие действия: открыть форму в режиме **Конструктор**. Щелкните правой кнопкой мыши на форме и выберите «*Свойства формы*». В диалоговом окне «*Окно свойств*», во вкладке **Макет**:

Полосы прокрутки – выбрать **Отсутствуют**,

Область выделения – выбрать **Нет**,

Кнопки перехода (навигации) – **Нет**,

Разделительные линии – Нет.

Добавьте рисунок (логотип) на главную форму. Для этого щелкните на элементе **Рисунок**, щелкните в левой части заголовка и протащите указатель по форме, чтобы начертить прямоугольник. Отпустите кнопку мыши. Появится диалоговое окно **Выбор рисунка**, позволяющее выбрать графический файл, который будет вставлен в элемент управления.

В режиме **Конструктора**, щелкнув правой кнопкой мыши по созданной кнопке *Список*, выберите **Свойства**. Открывается диалоговое окно **Кнопка: Кнопка0**. Во вкладке **Макет** добавьте фон или рисунок.

Сохраните изменения и переключитесь в режим формы, чтобы посмотреть на окончательный результат.

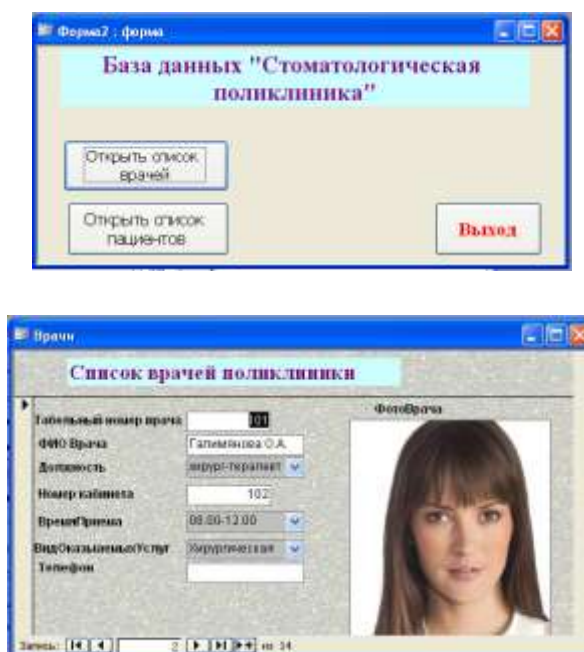


Рисунок 3.19 – Главная форма и форма «Врачи»

Изучив основы работы с СУБД MS Access, проведите работу с документом в соответствии со своим вариантом.

База данных для выбранного документа должна включать в себя несколько связанных таблиц, а также форму (формы) для работы с документом и реестром документов.

ЗАКЛЮЧЕНИЕ

Данное учебно-методические пособие предназначено для обучающихся ОПОП ВО по направлению 09.03.03 Прикладная информатика, направленность «Прикладная информатика в экономике» для обеспечения высокого уровня их подготовки при прохождении учебной практики по получению первичных профессиональных умений и навыков, в том числе первичных умений и навыков научно-исследовательской деятельности (в области информатики).

В пособии представлены общие сведения о практике, а также теоретический и практический материал о работе в MS Excel и MS Access, необходимый для выполнения обучающимися самостоятельных заданий.

Автор надеется, что данное учебно-методическое пособие позволит обучающимся максимально продуктивно выполнить задания по учебной практике, приобрести первичные профессиональные умения и навыки, в том числе первичные умения и навыки научно-исследовательской деятельности.

Приложение А

МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ
ИМЕНИ И.Т.ТРУБИЛИНА»

Факультет прикладной информатики

Кафедра информационных систем

ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ

Студента (ки) _____
1 курса очной (заочной) формы обучения группы _____
Направление подготовки 09.03.03 Прикладная информатика
Направленность Прикладная информатика в экономике
Вид практики: учебная
Тип практики: по получению первичных профессиональных умений и навыков, в том числе первичных умений и навыков научно-исследовательской деятельности (в области информатики)

№ п/п	Содержание задания	Ожидаемый результат
	Анализ предметной области (изучение предложенного документа)	Описание документа и требований, предъявляемым при работе с ним
	Работа с предложенным документом в MS Access	База данных для работы с документом; описание базы данных
	Работа с предложенным документом в MS Excel с применением VBA	Отчет о проделанной работе; описание разработки
	Составления отчета о прохождении практики	Отчет по практике

Студент _____ ФИО

Руководитель от КубГАУ
доцент _____ Замотайлова Д.А.

« ____ » _____ 201__ г.

Ожидаемые результаты прохождения практики соответствуют программе и заявленным компетенциям

МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ
ИМЕНИ И. Т. ТРУБИЛИНА»

Факультет прикладной информатики

Кафедра информационных систем

Рабочий график (план)

Студента (ки) _____

1 курса очной (заочной) формы обучения группы _____

Направление подготовки 09.03.03 Прикладная информатика

Направленность Прикладная информатика в экономике

Вид практики: учебная

Тип практики: по получению первичных профессиональных умений и навыков, в том числе первичных умений и навыков научно-исследовательской деятельности (в области информатики)

Дата	Краткое содержание работы	Ожидаемый результат
	Анализ предметной области (изучение предложенного документа)	Описание документа и требований, предъявляемым при работе с ним
	Работа с предложенным документом в MS Access	База данных для работы с документом; описание базы данных
	Работа с предложенным документом в MS Excel с применением VBA	Отчет о проделанной работе; описание разработки
	Составления отчета о прохождении практики	Отчет по практике

Подпись руководителя практики:

от КубГАУ

_____ Замотайлова Д.А.

« _____ » _____ 201__ г.

МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ
ИМЕНИ И.Т.ТРУБИЛИНА»

Факультет прикладной информатики

**ДНЕВНИК
ПРОХОЖДЕНИЯ ПРАКТИКИ**

Студента (ки) _____

1 курса очной (заочной) формы обучения группы _____

Направление подготовки 09.03.03 Прикладная информатика

Направленность Прикладная информатика в экономике

Вид практики: учебная

Тип практики: по получению первичных профессиональных умений, в том числе первичных умений и навыков научно-исследовательской деятельности (в области информатики)

Период практики с _____ по _____ 20__ г.

Преподаватель, руководитель практики от КубГАУ:

доцент. канд. экон., наук Замотайлова Д.А.

Кафедра информационных систем

Дата	Содержание работы	Полученные результаты	Отметка руководителя практики о выполнении работы
	Анализ предметной области (изучение предложенного документа)	Описание документа и требований, предъявляемым при работе с ним	
	Работа с предложенным документом в MS Access	База данных для работы с документом; описание базы данных	
	Работа с предложенным документом в MS Excel с применением VBA	Отчет о проделанной работе; описание разработки	
	Составления отчета о прохождении практики	Отчет по практике	

Студент

_____ Ф.И.О.

Подпись руководителя практики:

от КубГАУ _____ Замотайлова Д.А.

Аттестационный лист по практике

Ф.И.О

Обучающийся 1 курса направления подготовки 09.03.03 «Прикладная информатика», направленность «Прикладная информатика в экономике», успешно прошел учебную практику (по получению первичных профессиональных умений и навыков, в том числе первичных умений и навыков научно-исследовательской деятельности (в области информатики))

в объеме 108/3 часов/з. ед. с «___» _____ 201__ года по «___» _____ 201__ года в ФГБОУ ВО «Кубанский ГАУ имени И.Т. Трубилина».

В ходе практики обучающийся согласно программы практики освоил следующие компетенции

Наименование компетенций	пороговый	средний	высокий
ОПК-2 - способность анализировать социально-экономические задачи и процессы с применением методов системного анализа и математического моделирования			
ОПК-4 – способность решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности			
ПК-6 - способность собирать детальную информацию для формализации требований пользователей заказчика			

Руководитель практики от университета
доцент каф. информационных систем,
канд. экон. наук

_____ Замотайлова Д.А.

КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ АГРАРНЫЙ УНИВЕРСИТЕТ
ФАКУЛЬТЕТ ПРИКЛАДНОЙ ИНФОРМАТИКИ
Кафедра информационных систем

ОТЧЕТ

по учебной практике (по получению первичных профессиональных умений и навыков, в том числе первичных умений и навыков научно-исследовательской деятельности (в области информатики))

Выполнил студент группы: _____

Руководитель учебной практики: доцент, кандидат экономических наук
Замотайлова Д.А.

Защищен _____ Оценка _____
(дата)

Члены комиссии _____

Краснодар

201_

Создать в Excel и Access формы ввода и печати документов

1. Создать форму ввода приходного ордера на склад и его печать.
2. Создать форму ввода платежного требования и его печать.
3. Создать форму ввода акта взаиморасчетов и его печать.
4. Создать форму ввода приказа о приеме на работу и его печать.
5. Создать форму ввода иска и его печать.
6. Создать форму ввода платежного поручения и его печать.
7. Создать форму ввода счет-фактуры и ее печать.
8. Создать форму ввода доверенности и ее печать.
9. Создать форму ввода товарной накладной и ее печать.
10. Создать форму ввода счета на оплату и его печать.
11. Создать форму ввода внутренней накладной и ее печать.
12. Создать форму ввода приходного кассового ордера и его печать.
13. Создать форму ввода расходного кассового ордера и его печать.
14. Создать форму ввода акта на списание основных средств и его печать.
15. Создать форму ввода акта на списание материальных ценностей и его печать.
16. Создать форму ввода авансового отчета и его печать.
17. Создать форму ввода наряда на выполнение работ и его печать.
18. Создать форму ввода платежной ведомости и ее печать.
19. Создать форму ввода объявления о взносе наличных и его печать.
20. Создать форму ввода ведомости на выдачу продукции и ее печать.
21. Создать форму ввода договора поставки продукции и его печать.
22. Создать форму ввода договора покупки продукции и его печать.
23. Создать форму ввода договора отпуска продукции и его печать.

24. Создать форму ввода приказа об увольнении и его печать.
25. Создать форму ввода товарно-транспортной накладной и её печать.
26. Создать форму ввода акта оказания услуг и его печать.
27. Создать форму ввода требование-накладной и её печать.