

МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА РФ
НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ
АГРАРНЫЙ УНИВЕРСИТЕТ
ФАКУЛЬТЕТ ЭКОНОМИКИ И УПРАВЛЕНИЯ

**Основы организации баз данных в
среде СУБД**

**Методические указания для лабораторно-
практических занятий и самостоятельной работы студентов**

НОВОСИБИРСК 2021

УДК 004.65 (07)
ББК 32.972.134, я7
О – 753

**Кафедра Бухгалтерского учета и
автоматизированной обработки информации**

Составители: *Л.Г. Шишина, старший преподаватель
кафедры БУ и АОИ*

Рецензенты: *к.ф.н. О.Б. Кравченко, доцент кафедры
гражданского и гражданского процессуального права*

Основы организации баз данных в среде СУБД:
Методические указания для лабораторно-практических занятий
и самостоятельной работы студентов юридического факультета,
обучающихся по направлению подготовки 40.03.01
Юриспруденция / Новосиб. гос. аграр. ун-т; Составители:
Шишина Л.Г. - Новосибирск, 2021. – 21 с.

Методические указания предназначены для выполнения
индивидуального задания на лабораторно-практических
занятиях и для самостоятельной работы студентов
юридического факультета, обучающихся по направлению
подготовки 40.03.01 Юриспруденция. Методические указания
могут быть рекомендованы для студентов других направлений
подготовки. Методические указания содержат задания для
лабораторно-практических занятий, рекомендации и краткие

теоретические сведения по их выполнению и контрольные
вопросы по теме.

Методические указания обсуждены и одобрены на
заседании кафедры Бухгалтерского учета и
автоматизированной обработки информации (протокол № 10
от « 15» 06. 2021 г.).

Методические указания утверждены и рекомендованы
к изданию методической комиссией юридического
факультета (протокол № 10 от «15» 05. 2021 г.).

©
Новосибирски
й
государственн
ый аграрный
университет,
2021

Содержание

Введение	4
Задание	5
Порядок выполнения работы	5
1. Подготовительная часть работы	6
Разработка таблиц	6
Связи между таблицами	6
2. Выполнение задания в приложении MS Access	8
Создание новой базы данных	8
Создание новой таблицы	8
Режим Конструктора структуры таблицы	8
Создание схемы данных	9
Заполнение таблицы	11
Создание форм	11
Создание запросов	11
Режим конструктора запроса	11
Добавление вычисляемых полей в запрос	12
Создание отчетов	15
Режим Конструктора отчета	15
Структура отчета	16
Группировка данных в отчетах	17
Вычисление итогового значения в отчете	18
Контрольные вопросы	20
Список литературы	21

Введение

Основной целью дисциплин, изучающих информатику и информационные технологии является формирование у студентов основ информационной культуры, адекватной современному уровню и перспективам развития информационных процессов и систем, а также формирование у студентов знаний и умений, необходимых для свободной ориентировки в информационной среде и дальнейшего профессионального самообразования в области компьютерной подготовки.

Офисное приложение MS Access изучается в подобных дисциплинах и предназначено для приобретения практических навыков работы с реляционными базами данных любой сложности, а так же навыков документирования отчетных данных.

В данной методической разработке описаны этапы создания базы данных в MS Access на примере фирмы проката, подробно разобраны основные определения, понятия баз данных и этапы проектирования. Часть работы, которая выполняется на компьютере, описана обычным шрифтом, а часть подготовительной работы, которая выполняется в тетради - курсивом. Методические указания содержат задания для лабораторно-практических занятий, рекомендации, краткие теоретические сведения по их выполнению и контрольные вопросы по теме. Теоретический материал в указаниях выделен цветом, поэтому методические указания рекомендовано применять в электронном виде.

Методические указания могут быть использованы для аудиторной и самостоятельной работы студентов.

Знания и навыки, приобретенные при изучении темы, соответствуют требованиям универсальных и общепрофессиональных компетенций.

Задание

Для эффективной работы фирмы проката необходимо автоматизировать следующие операции:

- учет клиентов,
- учет устройств, которые выдаются клиентам в прокат,
- учет оперативных данных по заказам (дата выдачи, срок возврата, оплата проката),
- получение информации о задолженностях,
- получение информации о заказах за разные временные промежутки ,
- получение информации по клиентам,
- получение статистической информации для анализа.

Порядок выполнения работы

База данных (БД) - совокупность данных, организованных по определенным правилам, предусматривающим общие принципы описания, хранения и манипулирования данными.

Система управления базами данных (СУБД) - программный продукт, обеспечивающий централизованную работу с данными в БД.

Большинство БД для хранения данных используют таблицы. Каждая таблица имеет уникальное имя и состоит из строк и столбцов.

Запись - это строка таблицы.

Графы таблицы называются **полями**.

Порядок проектирования БД

1. Определение **цели** создания БД
2. Определение **таблиц**, которые должна содержать БД
3. Определение структуры таблиц (состав **полей**)
4. Определение **ключевых** полей
5. Определение **связей** между таблицами
6. Ввод данных
7. Создание форм, запросов, отчетов
8. Анализ созданной БД

1. Подготовительная часть работы

Разработка таблиц

Информация в БД может храниться в нескольких таблицах, между которыми установлены связи.

В первую очередь надо решить, какие данные будет содержать создаваемая БД, и как разбить информацию на отдельные таблицы. Основной принцип деления данных в СУБД заключается в том, что по данным об объектах одной группы строится отдельная таблица.

В соответствии с заданием были разработаны три таблицы, внешний вид которых представлен ниже:

Таблица № 1 «Клиент»

Шифр Клиента	Фамилия	Имя	Отчество	Адрес	Тел.

Таблица № 2 «Склад»

Шифр устройства	Наименование устройства	Количество	Стоимость в сутки.

--	--	--	--

Таблица № 3 «Прокат»

номер заказа	Шифр клиента	Шифр устройства	Дата выдачи	Срок возврата	Оплата проката

Связи между таблицами

Связь между таблицами осуществляется посредством значений одного или нескольких совпадающих полей.

Каждая строка таблицы в реляционных базах данных уникальна. Для обеспечения уникальности строк используются ключи, которые содержат одно или несколько полей таблицы.

Ключевое поле (ключ) – одно или несколько полей, комбинация значений которых однозначно определяет каждую запись в таблице. Достоинство реляционных баз данных, таких как Access, в том, что они позволяют быстро найти и связать данные из разных таблиц при помощи запросов, форм и отчетов.

В Access используется три типа ключевых полей:

- Простой ключ
- Составной ключ
- Счетчик

Простой ключ – это одно поле, если оно содержит уникальные значения. Например, поле, содержащее номер зачетной книжки студента, можно сделать ключевым полем, и это будет простой ключ.

Составной ключ – ключ, состоящий из нескольких полей, когда невозможно гарантировать уникальность значений каждого поля.

Счетчик при добавлении каждой записи в таблицу вносит в поле порядковое число. Указание такого поля в качестве ключевого является наиболее простым способом создания ключевых полей.

. Поля в таблицах и связи между ними по заданию показаны на рисунке 1.

Таблица № 1 «Клиент»

Шифр Клиента	Фамилия	Имя	Отчество	Адрес	Телефон
--------------	---------	-----	----------	-------	---------

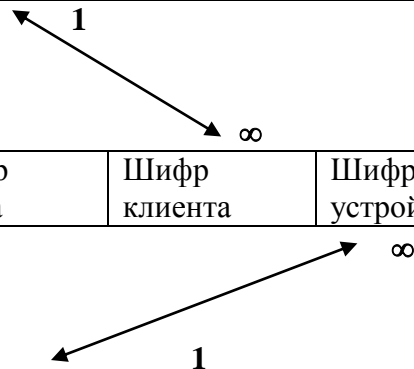
Таблица № 3 «Прокат»

Номер заказа	Шифр клиента	Шифр устройства	Дата выдачи	Срок возврата	Оплата проката
--------------	--------------	-----------------	-------------	---------------	----------------

Таблица № 2 «Склад»

Шифр устройства	Наименование устройства	Количество	Стоимость 1 ед. сутки
-----------------	-------------------------	------------	-----------------------

Рис.1 Ключевые поля на рисунке выделены жирным шрифтом.



2. Выполнение задания в приложении MS Access

Создание новой базы данных

При запуске MS Access предлагает открыть готовую БД или создать новую. Выбрав вариант создания новой базы данных, нужно указать имя БД и выбрать папку, где она будет храниться.

Все объекты одной БД хранятся в одном файле.

Создание новой таблицы

Предварительно надо определить структуру таблицы, т.е. определить количество и структуру полей и их имена, типы полей и их длину.

Чтобы создать таблицу необходимо выполнить следующие действия:

- в окне базы данных выберите вкладку **Создать**
- нажмите кнопку **Таблицы**
- выберите режим **Конструктор**

Режим Конструктора структуры таблицы

В режиме конструктора создается макет таблицы (структура), в котором описываются следующие элементы:

- имя поля
- тип данных

- описание поля
- определяется ключевого поля

. Окно режима конструктора делится на две части. В верхней – таблица для задания характеристик полей. В столбцы ее заносятся имена, типы данных и комментарии (свободный текст по усмотрению разработчика). В нижней – выводятся свойства выделенного поля.

Имя поля может содержать не более 64 любых символов но оно должно быть коротким и максимально информативным. Следует избегать пробелов в именах полей. При необходимости можно заменять пробел символом подчеркивания. Одинаковым полям в разных таблицах лучше давать одинаковые имена.

Тип поля выбирается из списка.

Размер поля нужно выбирать так, чтобы не было избыточности.

После определения всех полей и их свойств можно назначить ключевое поле, что обязательно, если таблица будет связана с другими таблицами БД. Чтобы объявить поле ключевым, достаточно выделить соответствующую ему строку и щелкнуть по кнопке с изображением ключа.

Если ключевое поле не было определено, то при сохранении созданной таблицы появится сообщение о создании ключевого поля. При ответе **Да** будет создано дополнительное ключевое поле с типом **Счетчик**.



Загрузите приложение MS Access.

Создайте пустую БД и три таблицы в ней разобранных выше:

- Таблица № 1 «Клиент»
- Таблица № 3 «Прокат»
- Таблица № 2 «Склад»

Структуры таблиц в режиме Конструктора

клиент : таблица		
	Имя поля	Тип данных
?	шифр клиента	Числовой
	фамилия	Текстовый
	имя	Текстовый
	отчество	Текстовый
	адрес	Текстовый
	телефон	Текстовый

показаны на рис. 2, 3

Рисунок 2. Таблица "Клиент" в режиме конструктора:

склад : таблица		
	Имя поля	Тип данных
?	шифр устройства	Числовой
	наименование устройства	Текстовый
	количество	Числовой
	стоимость 1 ед	Денежный

Рисунок 3. Таблица "Склад" в режиме конструктора

Аналогично создайте в режиме конструктора таблицу "Прокат".¹

Создание схемы данных

Реляционные базы данных состоят из множества взаимосвязанных двумерных таблиц. Связи обеспечивают поиск информации, целостность данных и их каскадное обновление, то есть невозможность исправлять данные в зависимой таблице без их обновления в главной и возможность изменения данных во всех зависимых таблицах при их обновлении в главной таблице. Режим **Схемы данных** отражает структуру таблиц в БД и их взаимосвязи. Связь устанавливается между полем первичного ключа одной из таблиц с соответствующими полями (не обязательно ключевыми) других таблиц (чаще всего используется **связь «один ко многим»**).

Схему данных можно корректировать в любой момент, удалять, строить новую.

Для создания схемы данных нужно выбрать вкладку

Работа с базами данных/Схема данных



. Откроется окно, в котором отобразится макет связей, если он был ранее сохранен. Иначе откроется диалоговое окно **Добавление таблицы**, в котором нужно выбрать таблицы или запросы для добавления в окно схемы данных.

¹ Необходимо учесть, что поля «шифр клиента» и «шифр устройства» в разных таблицах должны иметь одинаковые имена и характеристики.

Для создания связи нужно установить курсор на ключевое поле, нажать левую клавишу мыши и, не отпуская, перевести курсор в другую таблицу на поле, с которым устанавливается связь. Появится окно **Изменение связей**, в котором нужно нажать кнопку **Создать**.

Появится ломаная линия, соединяющая два связанных поля. Линии связи можно изменять, удалять и создавать новые.

Связь между парой таблиц может быть построена только одной парой полей. При попытке повторно связать те же таблицы, но через другие поля, приложение MSAccess предложит заменить ранее построенную связь на новую.

Чтобы при вводе в связанные поля допустимость значений отслеживалась автоматически, связи между таблицами нужно строить до ввода данных.



В режиме **Схемы данных** создайте связи между таблицами.

Процесс создания связей показан на рис.4.

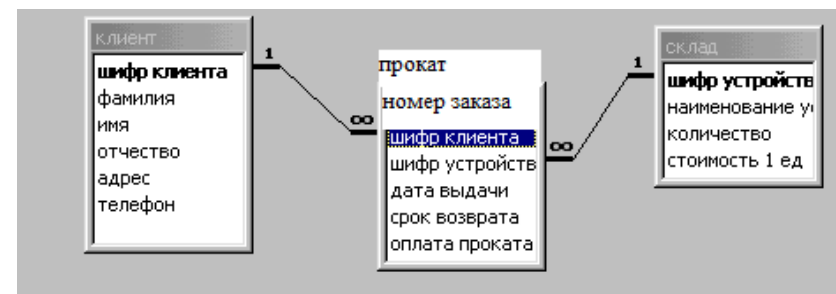
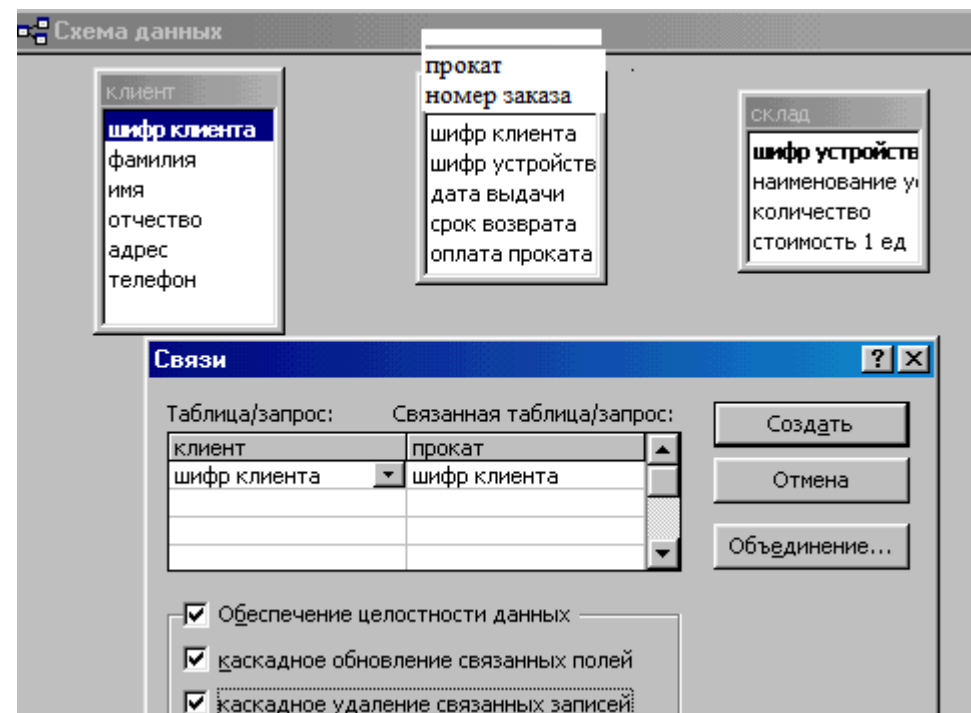


Рисунок 4. Процесс создания связей между таблицами.

Заполнение таблицы

В созданную таблицу необходимо внести данные, для этого открыть таблицу в режиме **Таблицы**. Записи можно добавлять, изменять и удалять.

При вводе данных нужно поддерживать тот тип, который был для них определен. Десятичный знак – запятая.



Заполните таблицы данными. Данные в таблицы введите самостоятельно (не менее 10 записей в каждой таблице) в режиме **Таблицы** или в режиме **Формы**.

Создание форм

Форма позволяет расположить на экране данные, относящиеся к одной записи или строке запроса, в наиболее удобном виде; при этом форму можно использовать для ввода или корректировки табличных данных. Записи можно просматривать через форму последовательно.

Для ввода данных можно выбирать типовые виды форм, создавать формы с помощью **Мастера форм** и корректировать их в режиме **Конструктора**.

Для создания формы используется кнопка **Мастер Форм** на вкладке **Создать** окна базы данных. Указывается имя таблицы или запроса, для которого создается форма. В его окнах необходимо указать все требуемые параметры формы и вывести ее на экран.



Создайте формы для всех таблиц с соответствующим названием. Используя форму «Прокат» введите 6 новых записей: 3 записи с одним клиентом, но разными устройствами, и 3 записи с одним устройством, заказанным разными клиентами.

Создание запросов

Запрос позволяет выбрать из одной или нескольких таблиц необходимые данные, соответствующие заданным критериям, и расположить их в заданной последовательности. Запросы можно использовать как обычные таблицы, можно на их основе создавать формы и отчеты.

Запросы создаются с помощью вкладки **Создать**, группы команд **Запросы**.

Режим конструктора запроса

Запрос удобнее создавать в режиме **Конструктора**. В окне **Добавление таблицы** последовательно добавляются необходимые таблицы и запросы.

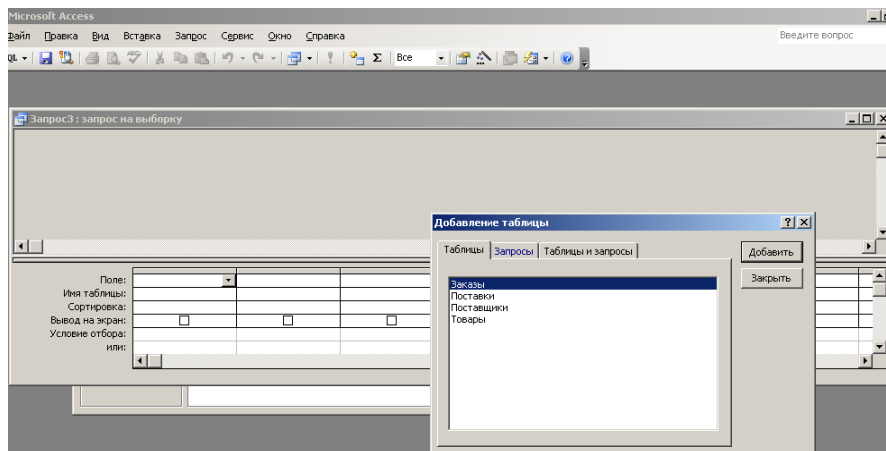


Рисунок 5. Внешний вид Конструктора запросов.

Окно конструктора запроса состоит из двух частей.

В верхней – показаны таблицы и другие объекты со связями, на основе которых строится запрос, в нижней – бланк запроса с описанием полей.

Названия полей из таблиц располагаются в первой строке столбцов бланка, куда их проще всего перенести с помощью мыши. Во второй строке бланка заносится имя таблицы, из которой было взято поле. Если убрать флажок в строке **Вывод на экран**, то соответствующее поле не будет выводиться в запросе.

Условие отбора может быть задано для всех полей, используемых в запросе. Существует синтаксис записи для выражений условий отбора. Тип поля тоже накладывает свои ограничения, например, значения символьного выражения берутся в кавычки.

Добавление вычисляемых полей в запрос

В бланке запроса можно создавать новые поля, которых нет в исходных таблицах. Чаще всего это расчётные поля. Например, если известна цена и количество товара, то можно рассчитать сумму стоимости этого товара.

Формат формулы имеет следующий вид:

Имя нового вычисляемого поля, двоеточие, исходные данные и операции для расчета (имена полей указываются в квадратных скобках).

Например: **Сумма: [цена]*[количество]**

В новом поле с именем Сумма будет выводиться результат произведения цены и количества.

Если запрос многотабличный, то в формуле перед именем поля нужно указать имя таблицы: **Сумма: Заказы![цена]*[количество]**

Вычисляемые поля проще всего создавать с помощью **Построителя выражений**, для вызова которого нужно нажать

кнопку  . Появится окно

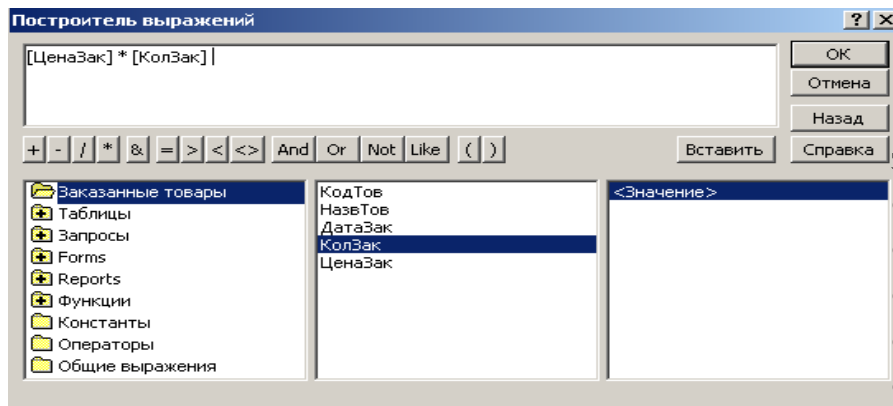


Рисунок 6. Внешний вид построителя выражений

Окно построителя выражений состоит из трех разделов, располагающихся сверху вниз.

Поле выражения. В верхней части окна построителя расположено поле, в котором создается выражение.

Кнопки операторов. В средней части окна построителя находятся кнопки с часто используемыми операторами.

Элементы выражения. В нижней части окна построителя находятся три поля.

В *левом* поле выводятся папки, содержащие таблицы, запросы, формы, объекты базы данных, встроенные и определенные пользователем функции, константы, операторы и общие выражения.

В *среднем* поле задаются определенные элементы или типы элементов для папки, заданной в левом поле.

В *правом* поле выводится список значений (если они существуют) для элементов, заданных в левом и среднем полях.



Создайте запрос 1 «Заказы». В этот

запрос выведите следующие данные:

- номер заказа
- шифр клиента
- фамилия клиента
- имя клиента
- отчество клиента
- шифр устройства
- наименование устройства
- дата выдачи –



. Создайте запрос 2 «Заказы за неделю»

Источником данных выберете запрос «Заказы». В этот запрос выведите ту же информацию, как в запросе «Заказы». В строке **Условие отбора** поля «дата выдачи» задайте условие, ограничивающее вывод данных за любые 5 дней.



Создайте запрос 3 «Стоимость заказа». В

этом запросе должны быть следующие данные:

- номер заказа
- фамилия клиента
- наименование устройства
- оплата проката
- стоимость заказа

Поле «стоимость заказа» является вычисляемым:
«стоимость заказа» = [стоимость 1 ед. в сутки] * ([срок возврата] - [дата выдачи])



Создайте запрос 4 «Задолженность

организации».

- номер заказа
- фамилия клиента
- наименование устройства
- оплата проката
- стоимость заказа
- задолженность организации

Поле «задолженность организации» - вычисляемое.

«задолженность организации» = [оплата проката] - [стоимость заказа]

В строке **Условие отбора** поля «задолженность организации» задайте условие,: > 0



Создайте запрос 5 «Задолженность

клиента».

- номер заказа
- фамилия клиента
- наименование устройства
- оплата проката
- стоимость заказа
- задолженность клиента

Поле «задолженность клиента» - вычисляемое.

«задолженность клиента» = [стоимость заказа] - [оплата проката]

В строке **Условие отбора** поля «задолженность клиента» задайте условие,: > 0



Запрос 6. На основе запроса 4 выведете информацию о задолженностях организации перед клиентом превышающей 5000рублей.



Запросы 7-9. На основе запроса 5 сформируйте информацию о максимальной задолженности клиента перед организацией, минимальной и средней соответственно.



Запрос 10. Создайте запрос с общей суммой задолженностей клиентов перед организацией.

Создание отчетов

Представление данных в **отчете** аналогично форме, но в отчет включаются данные из всех записей, он разбит на страницы и полностью подготовлен для печати.

Стандартные отчеты делятся на столбцовые и ленточные. В столбцовом каждое поле выводится в отдельной строке (аналогично, как в форме) последовательно друг за другом. В ленточном - каждая запись в отдельной строке. В режиме предварительного просмотра можно в деталях увидеть, как будет выглядеть отчет.

Отчет можно создать тремя различными способами:

► При помощи автоотчета на основе таблицы или запроса.

Автоотчет служит для создания отчета, в котором выводятся все поля и записи базовой таблицы или запроса.

► При помощи мастера на основе одной или нескольких таблиц или запросов.

► Вручную в режиме конструктора.

Макет отчета целесообразно создавать с помощью **Мастера отчетов**, а затем использовать **Конструктор**.

Мастер отчетов задает вопросы пользователю и формирует отчет на основе его ответов:

- на каких таблицах или запросах будет основан отчет
- какие поля из этих источников данных должны использоваться.
- следует ли группировать данные
- выполнять сортировку и обобщение данных.
- О внешнем виде отчета (макет, ориентация, выбор одного из шести визуальных представлений)
- название отчета.

Режим Конструктора отчета

Режим конструктора позволяет вносить различные изменения, начиная от источника базовых данных и заканчивая цветом текста. Для перехода в этот режим нужно выделить

элемента управления или сделать его прозрачным. Кроме того, можно изменить внешний вид элемента управления, применяя эффекты оформления. Например, можно представить элемент управления приподнятым, утопленным или с вдавленными границами. В форме или отчете можно отобразить элемент управления с тенью или как рельефный.

► **Границы.** Можно задать цвет или ширину границ. Кроме того, можно выбрать тип линии границы. Например, предлагается выбор точечных или пунктирных границ.









► **Вид текста.** Для текста в элементе управления можно выбрать шрифт и размер шрифта. Можно выделить текст в элементе управления полужирным шрифтом, курсивом или подчеркиванием.

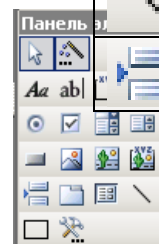
► **Положение.** Имеется возможность перемещать элементы управления или выравнивать их по другим элементам управления. Можно увеличить или уменьшить промежутки между элементами управления или задать равные промежутки между элементами.

В режиме конструктора выводится панель элементов, с помощью которой форматируются существующие элементы управления или добавляются новые.

Чтобы ее вызвать, необходимо нажать кнопку 

Таблица 4 Значение команд панели элементов


	Команда Выбор объектов. По умолчанию эта Команда активна. Используется для выделения, изменения размера, перемещения и редактирования элемента управления.
	Команда Мастера элементов для активизации мастера по созданию элементов управления.
	Команда Надпись используется для создания элемента управления Надпись , содержащего постоянный текст
	Команда Поле используется для создания элемента управления Поле для отображения текста, чисел, дат, времени, а также для вычисления значения выражений
	Команда Рисунок позволяет поместить в отчет статический рисунок
	Команда Прямоугольник используется для включения в отчет прямоугольников.
	Команда Линия используется для получения прямых линий.
	Команда Конец страницы позволяет вставлять разрыв страницы в многостраничный отчет.



Группировка данных в отчетах

Данные можно группировать в отчете путем выбора Рис одного или нескольких значений. элементов Например, путем выбора даты можно группировать все заказы за определенную дату.

Для группировки данных в отчете можно выбирать до 10 значений.

Группы данных можно создавать или изменять в режиме конструктора, выбирая команду **Сортировка и группировка**  на панели инструментов **Конструктора отчетов**. При этом в макет отчета можно добавить дополнительные разделы: название группы, примечание группы используя Контекстное меню.

Вычисление итогового значения в отчете.

Итоговые значения вычисляются в **Примечании** отчета или группы. Добавить дополнительные раздел можно с помощью Контекстного меню. Для создания итоговой строки необходимо увеличить область **Примечание отчета**. В отрывшейся области разместить элементы управления

Надпись и пустое **Поле** для формулы. При создании элемента управления **Поле** автоматически появляется связанный с ним элемент управления **Надпись**, который лучше удалить.

Рисунок 8. Вычисление итоговых значений в отчете.

Для создания формулы лучше использовать **Построитель выражений**. Формула для итоговых вычислений имеет следующий формат: **=Sum([Имя поля])**. Можно воспользоваться **построителем выражений** (команда **Свойства\вкладка Данные**).



Создайте следующие отчеты:

Задолженности : отчет																	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Заголовок отчета																	
Задолженности																	
Верхний колонтитул																	
	фамилия	наименование устройства			оплата проката									Задолженность			
Область данных																	
	фамилия	именование устройст			оплата проката									Задолженность			
Нижний колонтитул																	
	=Now()													"Страница " & [Page] & " из " & [Pages]			
Примечание отчета																	
	итого:													sum([Задолженность])			

Отчет 1 «Заказы клиентов» на основе запроса 3 «Стоимость заказа» с группировкой данных по клиентам.

Отчет 2 «Заказы устройств» на основе запроса 3 «Стоимость заказа» с группировкой данных устройствам.

Отчет 3 «Долги организации» на основе запроса 4 «Задолженность организации» с группировкой по клиентам, расчетом промежуточных итогов и общей задолженности.

Отчет 4 «Долги клиентов» на основе запроса 5 «Задолженность клиентов» с группировкой по клиентам, расчетом промежуточных итогов и общей задолженности

Отчет 5 «Статистика» на основе запросов 7-10

Оформите отчеты в виде документов согласно основным требованиям УСД² к стандартной форме построения документов.

² Унифицированная система документации

Контрольные вопросы

1. Что такое СУБД?
2. Что такое реляционные базы данных?
3. Какая структура используется в базах данных?
4. Что называется полем в базе данных?
5. Что называется записью в базе данных?
6. Какие объекты входят в состав базы данных?
7. Какие операции можно выполнять в режиме таблиц?
8. Что такое структура и как ее создать?
9. Как описываются поля?
10. Что такое ключевое поле и как оно задается?
11. Виды связей таблиц.
12. Что такое запросы?
13. Как запрос создается?
14. Структура макета запроса в режиме Конструктор..
15. Что такое формы?
16. Как создаются формы?
17. Структура макета формы в режиме Конструктор.
18. Что такое отчет?
19. Как создаются отчеты?
20. Структура макета отчета в режиме Конструктор.
21. Группировка данных в отчете.
22. Вычисления в отчете.

Список литературы

1. Гвоздева В. А. Базовые и прикладные информационные технологии : учебник / В.А. Гвоздева. — Москва : ИД «ФОРУМ» : ИНФРА-М, 2021. — 383 с. — (Высшее образование). - ISBN 978-5-8199-0885-3. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1406486>
2. Информатика и информационные технологии в профессиональной деятельности : учебник / А. А. Бабкин. С. В. Видов. С. А. Грязнов [и др.] ; под ред. В. П. Корячко, М. И. Купцова. - Рязань : Академия ФСИИ России, 2016. - 354 с. - ISBN 978-5-7743-0767-8. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1247750>
3. Озерский, С. В. Информатика и информационные технологии в профессиональной деятельности. Часть 1. Информатика : практикум / С. В. Озерский, Н. И. Улендеева. - Самара : Самарский юридический институт ФСИИ России, 2020. - 124 с. - ISBN 978-5-91612-314-2. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1322824>