

Министерство образования и науки Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Магнитогорский государственный технический университет им. Г.И. Носова»  
Многопрофильный колледж



**МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ВЫПОЛНЕНИЮ  
ПРАКТИЧЕСКИХ ЗАНЯТИЙ**

ПМ.03 Техническое обслуживание и ремонт компьютерных систем и комплексов  
МДК.03.01. Техническое обслуживание и ремонт компьютерных систем и комплексов  
Раздел 5. Базы данных

для студентов специальности  
09.02.01 Компьютерные системы и комплексы  
базовой подготовки

Магнитогорск, 2017

**ОДОБРЕНО:**

Предметно-цикловой комиссией  
Информатика и вычислительная техника

Председатель И.Г.Зорина

Протокол № 7 от 14 марта 2017

Методической комиссией МпК  
Протокол №4 от «23» марта 2017г

**Составитель:**

преподаватель ФГБОУ ВО МГТУ МпК *И.Г. Зорина/*  
преподаватель ФГБОУ ВО МГТУ МпК *Р.А. Закирова/*

Методические указания по выполнению практических занятий разработаны на основе рабочей программы ПМ.03 Техническое обслуживание и ремонт компьютерных систем и комплексов.

Содержание практических работ ориентировано на формирование общих и профессиональных компетенций по основной профессиональной образовательной программе по специальности 09.02.01 Компьютерные системы и комплексы: МДК.03.01. Техническое обслуживание и ремонт компьютерных систем и комплексов.

## СОДЕРЖАНИЕ

1 ВВЕДЕНИЕ .....	4
2 МЕТОДИЧЕСКИЕ УКАЗАНИЯ .....	7
Практическая работа № 1 Создание таблиц и ввод исходных данных .....	7
Практическая работа № 2 Создание связей в базе данных	19
Практическая работа № 3 Проектирование структуры базы данных .....	28
Практическая работа № 4 Операции поиска и фильтрации данных .....	31
Практические работы № 5 Создание запросов .....	35
Практические работы № 6 Создание запросов с вычисляемыми полями .....	44
Практические работы № 7 Модификация базы данных с помощью запросов на изменение .....	49
Практические работы № 8 Работа с формами .....	52
Практические работы № 9 Применение отчетов для наглядного отображения данных .....	57
Практические работы № 10 Экспорт и импорт данных. ....	60
Практические работы № 11 Организация простейших SQL-запросов .....	63
Практические работы № 12 Выполнение SQL запросов с параметрами. ....	65
Практические работы № 13, 14 Использование функций. ....	67
Практические работы № 15 Использование вложенных подзапросов. ....	70
Практические работы № 16 Использование объединения таблиц .....	72

## 1 ВВЕДЕНИЕ

Важную часть теоретической и профессиональной практической подготовки студентов составляют практические занятия.

Состав и содержание практических работ направлены на реализацию действующего федерального государственного образовательного стандарта среднего профессионального образования по специальности 09.02.01 Компьютерные системы и комплексы.

Ведущей дидактической целью практических занятий является формирование практических умений - профессиональных (умений выполнять определенные действия, операции, необходимые в последующем в профессиональной деятельности), необходимых в последующей учебной деятельности по профессиональным модулям.

В соответствии с рабочей программой ПМ.03 Техническое обслуживание и ремонт компьютерных систем и комплексов, МДК.03.01. Техническое обслуживание и ремонт компьютерных систем и комплексов предусмотрено проведение практических работ.

В результате их выполнения, обучающийся должен:

*уметь:*

- формировать и настраивать схему базы данных;
- проектировать базу данных;
- устанавливать связи между таблицами;
- обеспечивать целостность данных;
- создавать поля со списками;
- создавать формы;
- добавлять различные элементы управления на форму;
- создавать отчеты;
- создавать запросы на выборку, обновление, добавление и удаление;
- создавать объекты баз данных в современных системах управления базами данных и управлять доступом к этим объектам;
- формировать и настраивать схему базы данных.

Содержание практических занятий ориентировано на формирование общих компетенций по профессиональному модулю основной профессиональной образовательной программы по специальности:

ОК 1. Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес.

ОК 2. Организовывать собственную деятельность, определять методы и способы выполнения профессиональных задач, оценивать их эффективность и качество.

ОК 3. Решать проблемы, оценивать риски и принимать решения

в нестандартных ситуациях.

ОК 4. Осуществлять поиск, анализ и оценку информации, необходимой для постановки и решения профессиональных задач, профессионального и личностного развития.

ОК 5. Использовать информационно-коммуникационные технологии для совершенствования профессиональной деятельности.

ОК 6. Работать в коллективе и команде, обеспечивать ее сплочение, эффективно общаться с коллегами, руководством, потребителями.

ОК 7. Ставить цели, мотивировать деятельность подчиненных, организовывать и контролировать их работу с принятием на себя ответственности за результат выполнения заданий.

ОК 8. Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации.

ОК 9. Быть готовым к смене технологий в профессиональной деятельности.

И овладению профессиональными компетенциями:

ПК 3.1. Проводить контроль, диагностику и восстановление работоспособности компьютерных систем и комплексов.

ПК 3.2. Проводить системотехническое обслуживание компьютерных систем и комплексов.

ПК 3.3. Принимать участие в отладке и технических испытаниях компьютерных систем и комплексов; инсталляции, конфигурировании программного обеспечения.

Выполнение студентами практических работ по программой ПМ.03 Техническое обслуживание и ремонт компьютерных систем и комплексов, МДК.03.01. Техническое обслуживание и ремонт компьютерных систем и комплексов направлено на:

- обобщение, систематизацию, углубление, закрепление, развитие и детализацию полученных теоретических знаний по конкретным темам междисциплинарных курсов;

- формирование умений применять полученные знания на практике, реализацию единства интеллектуальной и практической деятельности;

- формирование и развитие умений: наблюдать, сравнивать, сопоставлять, анализировать, делать выводы и обобщения;

- развитие интеллектуальных умений у будущих специалистов: аналитических, проектировочных, конструктивных и др.;

- выработку при решении поставленных задач профессионально значимых качеств, таких как самостоятельность, ответственность, точность, творческая инициатива.

Продолжительность выполнения практической работы составляет не менее двух академических часов и проводится после соответствующего занятия, которое обеспечивает наличие знаний, необходимых для ее выполнения.

## 2 МЕТОДИЧЕСКИЕ УКАЗАНИЯ

### Тема 03.01.05 Базы данных

#### Практическая работа № 1 Создание таблиц и ввод исходных данных

**Цель работы:** получение практических навыков по созданию таблиц в базе данных, заполнению таблиц исходными данными, обеспечение целостности данных, работа с масками.

**Выполнив работу, Вы будете:**

*уметь:*

- создавать таблицы в базе данных;
- заполнять таблицы исходными данными;
- применять маски для ввода исходных данных.

**Материальное обеспечение:**

Персональный компьютер, Access, методические указания для выполнения практических работ, вариант задания.

**Задание:**

Создать таблицы в базе данных Access.

**Порядок выполнения работы:**

Теория:

Необходимо создать базу данных студент. Информация, которая будет храниться в базе данных:

1. «Фамилия», «Имя» и «Отчество» студента.
2. «Название группы» – название группы, в которой учится студент.
3. «Номер зачетной книжки» – идентификатор, по которому можно однозначно идентифицировать студента.
4. «Домашний адрес» – место проживания студента
5. «Номер телефона» – номер телефона для связи со студентом.

Все это будет полями БД, однако так же необходимо задать типы полей:

1. «Фамилия», «Имя», «Отчество» – текстовый (можно ограничить размер полей в 20 символов).
2. «Название группы» – текстовый, (пока, для простоты считаем, что группа студентов состоит из букв и цифр формата КВТ-045, то есть 7 символов). Примечание: можно создать отдельную таблицу со списком групп и настроить выбор группы из созданной таблицы

3. «Номер зачетной книжки» – числовой (ключевое поле, т.к. номер зачетной книжки является уникальным для каждого студента).
4. «Домашний адрес» – текстовый, 100 символов.
5. «Номер телефона» – числовой.

Все поля, за исключением номера телефона, должны быть обязательными.

Для создания базы данных запустите Microsoft Access 2007, перед вами появится окно, представленное на рис. 1.1.

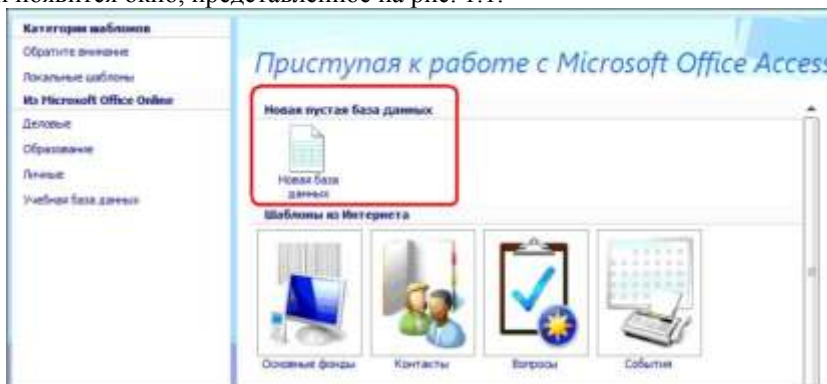


Рис. 1.1. Новая база данных

В нем выбираем шаблон Новая база данных. В правой части программы появится окно, в котором можно присвоить имя новой БД. Выбираем каталог «D:\STUDY\», указываем имя, например, «Студенты», и нажимаем на кнопку Создать (рис. 1.2).

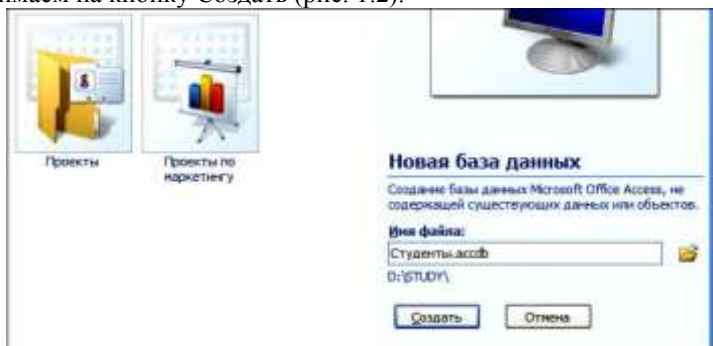


Рис. 1.2. Результат создания базы данных

В результате появится окно, в котором создается база данных (рис. 1.3).



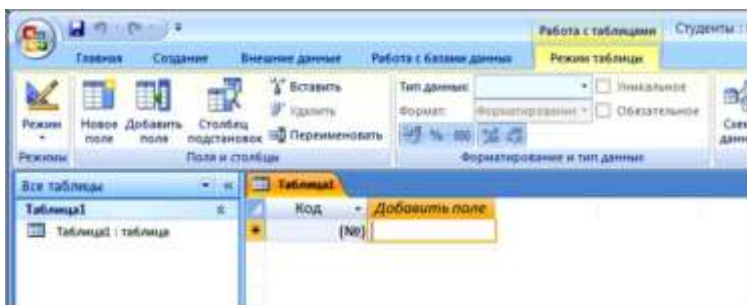


Рис. 1.3. Результат создания базы данных

На закладке Главная с помощью кнопки Вид перейдем в режим Конструктора (рис. 1.4).

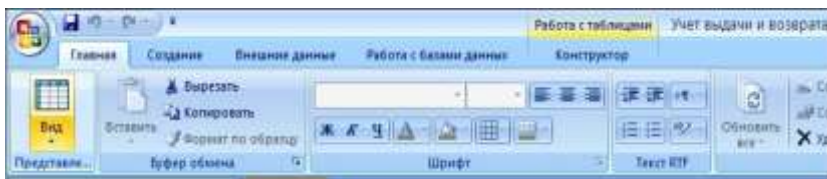


Рис. 1.4. Закладка «Главная»

Сохраним таблицу с именем «Основные сведения о студенте» (рис. 1.5).

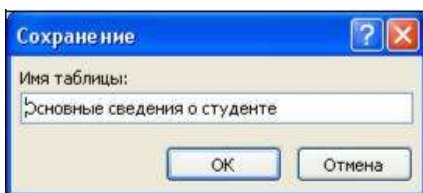


Рис. 1.5. Задание имени таблицы

Необходимо обратить внимание, что при создании таблицы обязательно задается ключевое поле. В первой таблице нашего примера ключевым полем будет поле «номер зачетной книжки».

Рассмотрим Маску ввода (InputMask). Маска ввода определяет способы ввода данных в БД. Например, с помощью маски ввода можно дать пользователю возможность вводить название группы, телефонные номера только в определенном формате.

Создание Маски ввода (InputMask). В Microsoft Access 2007

Маска ввода задается при создании поля таблицы на вкладке «Общие», пример представлен на рис. 1.6. Делается это в режиме конструктора таблицы, а результаты будут видны в процессе заполнения таблицы.

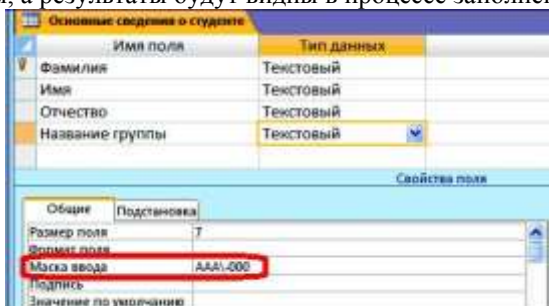


Рис. 1.6. Создание маски ввода

Маска ввода представляет собой набор текстовых констант и знаков маски, определяющий, какие данные и каким образом можно вводить. Например, с помощью маски ввода можно потребовать, чтобы пользователи вводили даты или телефонные номера так, как принято в определенной стране, например:

ДД.ММ.ГГГГ, где ДД – день, ММ – месяц, а ГГГГ – год. или  
 – (\_\_\_\_) \_\_\_\_–\_\_\_\_ доб. \_\_\_\_

Правильно задав маску ввода, можно заметить, что в режиме заполнения таблицы разрешается заменять знаки местозаполнителей собственными значениями, но нельзя изменить или переместить текстовые константы или знаки-разделители. Например, можно ввести дату, но нельзя изменить точки, разделяющие отдельные части даты: 09.01.2007.

Кроме того, с помощью маски ввода можно сделать одну или несколько частей поля или элемента управления обязательными для заполнения. Например, если пользователь ввел в маске телефонный номер, но не ввел междугородный код: (\_\_\_\_) 555-0187доб. \_\_\_\_, то он не сможет сохранить изменения, пока не введет междугородный код. Особенности работы маски зависят от ее параметров, заданных разработчиком базы данных.

Маски ввода выполняют значительную часть задач проверки данных, поскольку позволяют вводить данные только определенным образом. Это означает, что они могут предотвратить ввод пользователем неверных данных (например, телефонного номера в поле для даты). Кроме того, маски ввода помогают поддерживать единообразие вводимых дан-

ных. Это единообразие упрощает поиск данных и управление базой данных.

Знание правил создания маски ввода позволяет сделать работу в Microsoft Access более грамотной, создаваемые БД будут выглядеть профессионально и работать с ними будет значительно проще.

Чтобы добавить маску ввода к полям таблицы, помимо создания вручную, можно запустить Мастер масок ввода и выбрать готовый вариант для используемого типа данных. Таким же образом маска ввода добавляется к элементу управления на форме. Например, если используются инструменты, предусмотренные в Access для автоматического создания формы, то соответствующий элемент управления наследует маску ввода, которая была применена к полям таблицы. Можно также вручную добавить маски ввода к элементам управления в форме и, как и в случае с таблицами, они добавляются к свойству Маска ввода элемента управления.

Поле «Фамилия» – выбираем тип данных Текстовый (рис. 1.7) определяем свойства поля: Размер поля – 30, Обязательное поле Да, Пустые строки –нет.

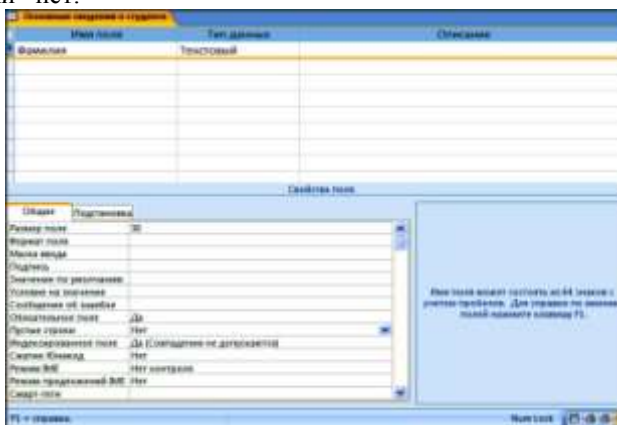


Рис. 1.7. Создание поля «Фамилия»

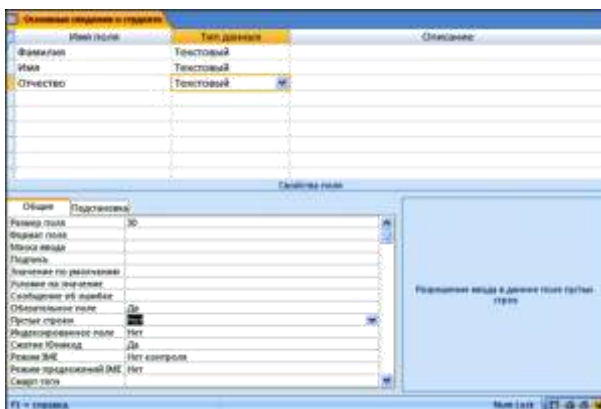


Рис. 1.8. Создание полей «Имя» и «Отчество»

Аналогичным образом добавляем поля «Имя» и «Отчество»(рис. 1.8). При задании поля «Название группы» создадим маску ввода вида «AAA\-000» (рис. 1.9), где «AAA» – это буквы в названии группы (латиница – раскладка клавиатуры «Английская»), а «000»– обязательные числа. Все символы, которые могут использоваться при создании маски ввода, описаны в Приложении.

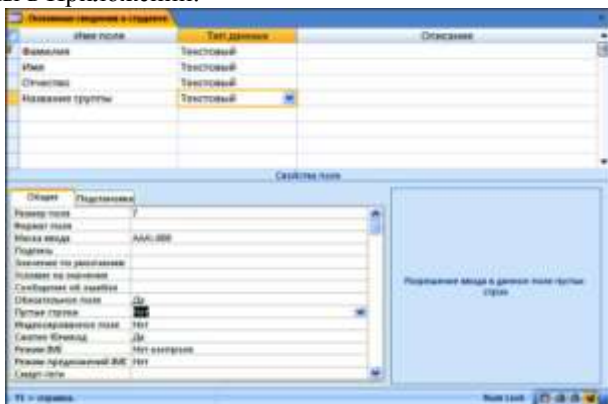


Рис. 1.9. Создание поля «Название группы»

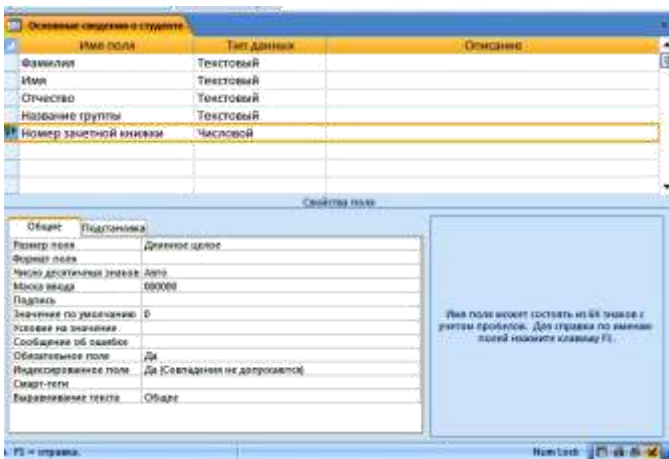


Рис. 1.10. Создание поля «Номер зачетной книжки»

Для поля «Номер зачетной книжки» зададим числовой тип – Целый, с числом десятичных знаков 6 (шесть нулей в поле маска ввода) и указанием обязательного поля. Определяем значение по умолчанию и определяем это поле как ключевое. Обратите внимание, оно становится индексированным полем, в котором совпадения не допускаются (рис. 1.10).

Учитывая, что «Домашний адрес» является обязательным и «структурировать» его достаточно трудно (хотя можно, создав отдельные поля с городом, улицей, домом и квартирой, но для нашей базы данных это не обязательно), то зададим Размер поля – 100, Обязательное поле – Да, Пустые строки – Нет (рис. 1.11).

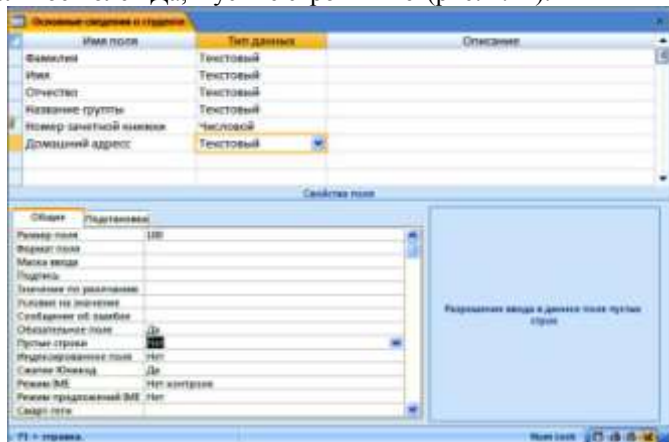


Рис. 1.11. Создание поля «Домашний адрес»

«Номер телефона» зададим с числовым типом,определив десяти-значную маску.

После проделанных операций необходимо сохранить таблицу. Для перехода в режим таблицы воспользуемся функцией Режим на панели инструментов (рис. 1.13).

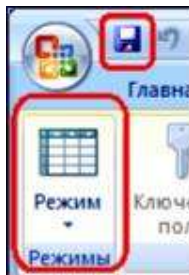


Рис. 1.13. Сохранение и переход в режим таблицы

Режим таблицы с одной введенной записью (строкой) с информацией о студенте представлен на рис. 1.14.



Рис. 1.14. Отображение данных в режиме таблицы

При попытке ввести номер телефона вида 9888888888 может выйти предупреждение «Введенное значение не подходит для данного поля» (рис. 1.15).

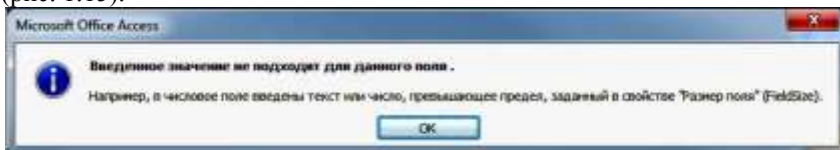


Рис. 1.15. Предупреждение при вводе данных

Это означает, что не совсем удачно был выбран тип поля данных «Номер телефона» – длинное целое (четырёхбайтовый тип данных), в

котором невозможно сохранить десятизначный номер. Поэтому нужно сменить тип, который бы позволил хранить длинные номера, например на текстовый тип.

Таким образом, можно снова перейти к вводу данных, создайте несколько строк в режиме заполнения таблицы и сохраните свою работу (рис. 1.16).

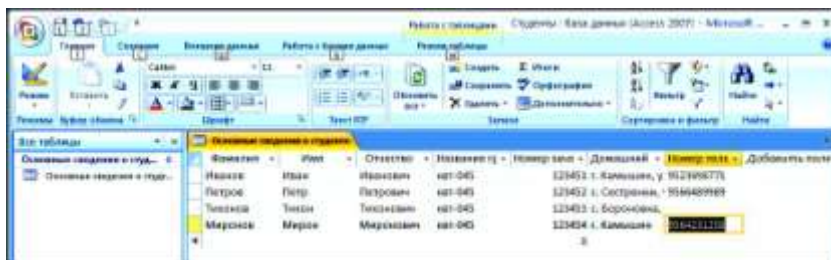


Рис. 1.16. Ввод данных в таблицу

### Ход работы:

1. Изучить среду разработки приложений баз данных MicrosoftAccess 2007
2. Создайте маску для полей «Фамилия», «Имя», «Отчество» так, чтобы первая буква в таблице для данных полей была заглавной.
3. Сделайте маску поля «Название группы» таким образом, чтобы можно было ввести любое имя группы (не менее 2 букв, не более 8 букв) с сохранением трехзначного номера.
4. Попробуйте доработать базу данных. Добавьте недостающую информацию о студенте, например пол, дату рождения и тому подобное.
5. Дополнительное задание. Создайте еще одну таблицу «Список групп», где будет храниться название группы, краткое описание группы, фамилия, имя, отчество куратора группы. Сделайте так, чтобы в основной таблице «Основные сведения о студенте» в поле «Название группы» название группы выбиралось из таблицы «Список групп» и был запрещен ручной ввод.

Таблица. 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 A до Z или от A до Я, ввод обязателен).
?	Буква (от A до Z или от A до Я, ввод не обязателен).
A	Буква или цифра (ввод обязателен).

В таблице ниже приведены примеры использования масок ввода.

<b>Маска ввода</b>	<b>Обеспечивает ввод значения в виде</b>	<b>Примечания</b>
(000) 000-0000	(206) 555-0199	В данном случае пользователи должны вводить код города, так как в соответствующем разделе маски (000 в скобках) используется заполнитель 0.
(999) 000-0000!	(206) 555-0199 ( ) 555-0199	В этом случае в области междугородного кода используется местозаполнитель 9, поэтому междугородные коды необязательны. Кроме того, восклицательный знак (!) обуславливает заполнение маски слева направо.
(000) AAA-AAAA	(206) 555-ТЕЛЕ	Предоставляется возможность заменить буквами четыре последних цифры телефонного номера в формате США. Обратите внимание на местозаполнитель 0 в области кода города, который делает междугородный код обязательным.
#999	-20 2000	Любое положительное или отрицательное число, включаю-



		шее не более четырех знаков и не имеющее разделителей тысяч и дробной части.
>L????L?000LO	ЗЕЛЕНЫЙ339М3 МАЙ Р 452Б7	Сочетание обязательных (L) и необязательных (?) букв и обязательных цифр (0). Знак "больше" требует вводить все буквы в верхнем регистре. Чтобы использовать маску ввода этого типа, необходимо задать для типа данных поля таблицы значение <b>Текстовый</b> или <b>Поле МЕМО</b> .
00000-9999	98115- 98115-3007	Обязательный почтовый индекс и необязательная область четырехзначного расширения.
>L<???????????????	Мария Вадим	Имя или фамилия, первая буква которой автоматически вводится заглавной.
ISBN 0- &&&&&&&&&-0	ISBN 1-55615- 507-7	Учетный номер книги с текстовой константой, обязательными первой и последней цифрами и произвольной комбинацией букв и знаков между ними.
>LL00000-0000	БД51392-0493	Сочетание обязательных букв и цифр, все в верхнем регистре. Данный тип маски ввода помогает, например, правильно вводить инвентарные номера или заполнять другие формы складского учета.

### Форма представления результата:

Необходимо предоставить созданную базу данных с заполненными таблицами и созданными масками.

### Критерии оценки:

Оценка «отлично» ставится, если задание выполнено верно.

Оценка «хорошо» ставится, если ход выполнения задания верный, но была допущена одна или две ошибки, приведшие к неправильному ответу.

Оценка «удовлетворительно» ставится, если в работе не получен ответ и приведено неполное выполнение задания, но ход выполнения задания верный

Оценка «неудовлетворительно» ставится, если задание не выполнено или если приведен правильный ответ, но решение отсутствует.

## Тема 03.01.05 Базы данных

### Практическая работа № 2 Создание связей в базе данных

**Цель работы:** получение практических навыков по освоению операций создания таблиц с помощью **Конструктора** и **Мастертаблиц**.  
Получение практических навыков по освоению операций подстановок.

#### **Выполнив работу, Вы будете:**

*уметь:*

- устанавливать связи между таблицами;
- обеспечивать целостность данных;
- создавать поля со списками;
- формировать и настраивать схему базы данных.

#### **Материальное обеспечение:**

Персональный компьютер, Access, методические указания для выполнения практических работ, вариант задания.

#### **Задание:**

Спроектировать схему базы данных. По спроектированной схеме создать базу данных в Access. Освоить навыки подстановки значений.

#### **Краткие теоретические сведения:**

Проектирование базы данных заключается в ее многоступенчатом описании с различной степенью детализации и формализации, в ходе которого производится уточнение и оптимизация структуры базы данных. Проектирование начинается с описания предметной области и задач информационной системы, идет к более абстрактному уровню логического описания данных и далее – к схеме физической (внутренней) модели базы данных. Трех основным уровням моделирования системы – концептуальному, логическому и физическому соответствуют три последовательных этапа детализации описания объектов базы данных и их взаимосвязей.

На концептуальном уровне проектирования производится смысловое описание информации предметной области, определяются ее границы, производится абстрагирование от несущественных деталей. В результате определяются моделируемые объекты, и их свойства и связи. Выполняется структуризация знаний о предметной области, стандартизируется терминология. Затем строится концептуальная модель, описыва-

ваемая на естественном языке. Для описания свойств и связей объектов применяются различные диаграммы.

На следующем шаге принимается решение о том, в какой конкретно СУБД будет реализована база данных. Выбор СУБД является сложной задачей и должен основываться на потребностях с точки зрения информационной системы и пользователей. Определяющими здесь являются вид программного продукта и категория пользователей (или профессиональные программисты, или конечные пользователи, или то и другое). Другими показателями, влияющими на выбор СУБД, являются:

- Удобство и простота использования;
- Качество средств разработки, защиты и контроля базы данных;
- Уровень коммуникационных средств (в случае применения ее в сетях);
- Фирма-разработчик;
- Стоимость.

Каждая конкретная СУБД работает с определенной моделью данных. Под моделью данных понимается способ их взаимосвязи: в виде иерархического дерева, сложной сетевой структуры или связанных таблиц. В настоящее время большинство СУБД использует табличную модель данных, называемую *реляционной*.

На логическом уровне производится отображение данных концептуальной модели в логическую модель в рамках той структуры данных, которая поддерживается выбранной СУБД. Логическая модель не зависит от конкретной СУБД и может быть реализована на любой СУБД реляционного типа.

На физическом уровне производится выбор рациональной структуры хранения данных и методов доступа к ним, которые обеспечивает выбранная СУБД. На этом уровне решаются вопросы эффективного выполнения запросов к БД, для чего строятся дополнительные структуры, например индексы. В физической модели содержится информация обо всех объектах базы данных (таблицах, индексах, процедурах и др.) и используемых типах данных. Физическая модель *зависит* от конкретной СУБД. Одной и той же логической модели может соответствовать несколько разных физических моделей. Физическое проектирование является начальным этапом реализации базы данных.

#### **Порядок выполнения работы:**

1. Создать базу данных;
2. Заполнить таблицы базы данных;
3. Импортировать таблицу из Excel;
4. Создать мастер подстановок;

5. Предоставить схему полученной базы данных.

### Ход работы:

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 *Kafedra*.

Для создания таблицы базы данных:

- в окне базы данных выберите вкладку *Создание*, а затем щелкните по кнопке <Конструктор таблиц> (рис. 1).

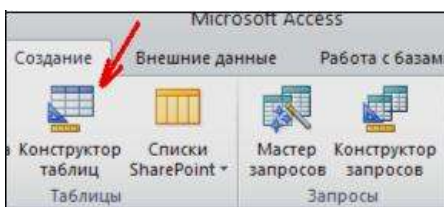


Рис. 1. Создание таблиц в режиме Конструктора

1.3. Для определения полей таблицы:

- введите в строку столбца *Имя поля* имя первого поля - *kod\_kaf*.
- В строке столбца «Тип данных» щелкните по кнопке списка и выберите тип данных *Числовой*. Поля вкладки *Общие* оставьте такими, как предлагает Access.

Таблица 1. Описание свойств полей таблицы

Имя поля	Тип данных	Размер поля (вкладка <i>Общие</i> )
Kod_kaf	Числовой	
Name_kaf	Текстовый	20
Nom_tel	Числовой	Длинное целое (по умолчанию)
Nom_aud	Числовой	Длинное целое (по умолчанию)
Col_sotr	Числовой	Целое
Zav_kaf	Текстовый	50 (Максимальное значение)

### 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 Kod\_kaf ключевым. Для этого откройте таблицу в режиме Конструктора и, поместив курсор на имя этого поля, щелкните по

кнопке  - Ключевое поле:

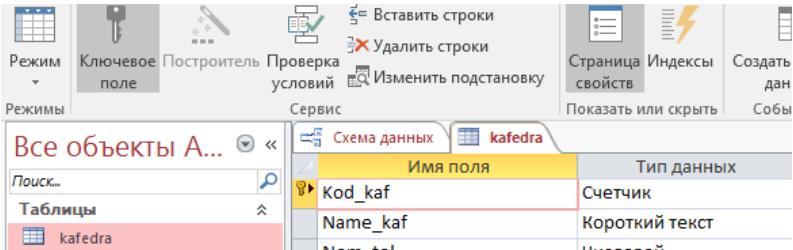


Рис. 2 Назначение поля ключевым

### Импорт данных

Таблицу «Zav\_kafedra» (используется для поля «Zav\_kaf») создайте в Excel, чтобы импортировать её в нашу базу данных. Поля таблицы:

Заведующий кафедрой
Иванов
Петров
Сидоров

Рис. 3 Таблица «Zav\_kafedra»

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

- Определите формат ячеек: для поля – текстовый.
- Сохраните таблицу в той же папке, где находится БД.
- В базе данных 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»,

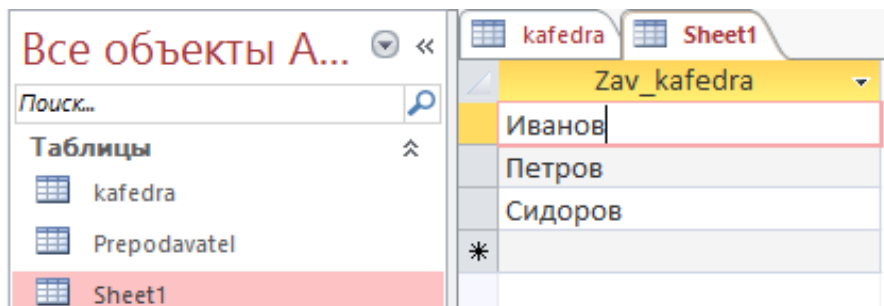


Рис. 4 Таблица «Zav\_kafedra»

Таблица «Numberofauditoriya» содержит следующие поля:

- ID\_audit (Ключевое поле)
- Number
- Description

Необходимо заполнить таблицу данными.

Используя «Мастер подстановок» заполняем поле Nom\_auditoria и Zav\_kaf в таблице kafedra.

Создаем таблицу Prepodavatel. Поля таблицы имеют следующий вид:

Имя поля	Тип данных
Id_prepod	Счетчик
Name_prep	Короткий текст
Kod_kaf	Числовой
Subject	Длинный текст

Рис. 5 Таблица «Prepodavatel»

Поле Kod\_kaf заполнить через Мастер подстановок. Вывести на экран схему получившейся базы данных

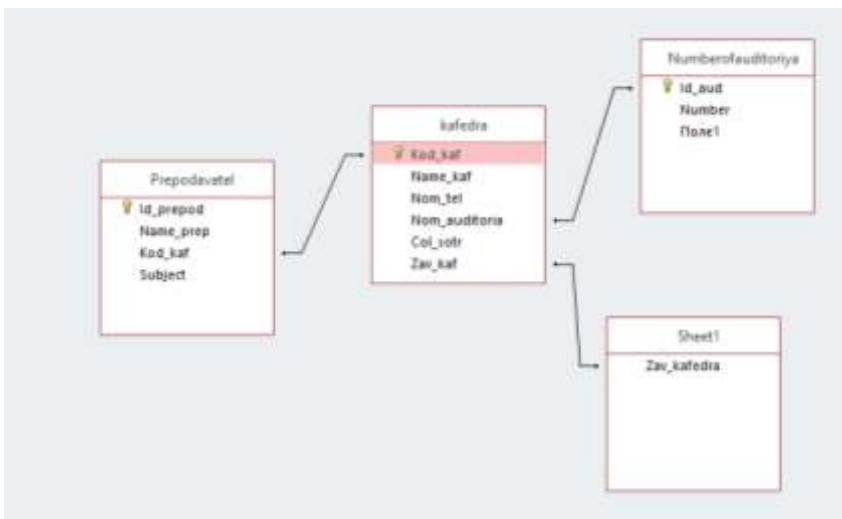


Рис. 6 Получившаяся схема базы данных

**Задание 2.** Создать базу данных и таблицы базы данных, заполнить таблицы данными.

#### Краткие теоретические сведения:

Связи *автоматические* устанавливаются **Мастером подстановок**. Посмотреть, установить, отредактировать связи можно командой *Работа с базой данных – Схема данных*.

Если связи устанавливаются первично, то откроется окно *Таблицы*, а если повторно, то окно *Связи*. Двойной щелчок на нужной таблице позволит перенести их в окно *Связи*.

Для установки связи между таблицами *вручную* нужно перетянуть связываемое поле из главной таблицы и наложить его на соответствующее поле подчиненной таблицы.

Удаление и изменение связей производится с помощью контекстного меню на линии связи, а также клавишей DEL.

В окне *Схема данных* двойной щелчок по линии связи позволит открыть окно *Связи*. В нем можно установить флажок у опции *Целостность данных*, линия связи станет гораздо темнее и появятся значки «1» и «∞», означающие отношение «один» или «многие».

Если система определила тип связи (в нижней части диалогового окна) «один-к-одному» или «один-ко-многим», то можно поставить флажок «Поддерживать целостность данных».



**Целостность данных** – это набор правил, защищающих данные от случайных изменений или удалений с помощью механизма поддержки корректности связей между связанными таблицами.

Если связь определена и система взяла на себя поддержку целостности данных, то при просмотре главной таблицы (отношение «один») слева, рядом с полосой выделения появится колонка со знаками «+». Щелчок на «+» позволит открыть подчиненную таблицу (отношение «много» или «один»).

#### **Порядок выполнения работы:**

1. Создайте в своей рабочей папке папку с именем *База данных*.
2. Запустите *MSAccess*. Создайте в созданной папке новую базу данных с именем *Туризм*.

#### Создание таблиц с помощью Конструктора

3. Создайте таблицу **Сотрудники** в режиме Конструктора. Наименования и типы полей представлены в приведенной таблице.

<b>Название поля</b>	<b>Тип данных</b>
Код сотрудника	Числовой
ФИО	Текст
Должность	Текст
Дата найма	Дата/Время
Дата рождения	Дата/Время
Домашний телефон	Текст
Адрес	Текст
Размер оклада	Числовой

4. Для поля *Домашний телефон* задайте маску, набрав, например, следующий шаблон (999) 999-99-99.
5. Для поля *Оклад* задайте условие, что он больше 5000 р., но не больше 10000. Для этого в свойстве «Условие на значение» установите (>5000) AND (<10000). Предусмотрите выдачу сообщения при ошибке ввода данных.
6. Установите для *Даты рождения* и *Даты найма* маску ввода. Используйте краткий формат даты.
7. Создайте первичный ключ.
8. Перейдите в режим просмотра таблицы.
9. Спрячьте некоторые столбцы. Сделайте их опять видимыми командами **Контекстное меню — Скрыть/Показать столбцы**.
10. Зафиксируйте столбцы, содержащие фамилию и имя. Освободите столбцы.
11. Поменяйте тип шрифта и его начертание (**Формат — Шрифт**).

12. Закройте окно таблицы **Сотрудники**, сохранив изменения.
13. Создайте новую таблицу **КлиентыСтраны**. В качестве первичного ключа задайте **Код Клиента** и **Код Тура**.

Название поля	Тип данных
Код клиента	Числовой
Название клиента	Текст
Контактное лицо	Текст
Признак группы	Логический
Телефон	Текст
Адрес	Текст

#### Использование Мастера подстановок

14. Создайте в режиме **Конструктора** таблицу **Договоры**, которая должна иметь следующие поля:

Название поля	Тип данных
Номер договора	Число
Код клиента	Числовой
Код тура	Число
Дата начала тура	Дата/Время
Дата окончания тура	Дата/Время
Число туристов	Числовой
Цена тура	Денежный
Дата платежа	Дата/Время
Код Сотрудника	Числовой

Поля *Код сотрудника*, *Код клиента*, *Код тура* являются полями подстановки. Для их задания используется Мастер подстановок:

- в Типе данных поля *Код сотрудника* раскрыть список типов и выбрать **Мастер подстановок**;
- указать, что столбец подстановки получает свои значения из таблицы **Сотрудники**;
- выбрать поля *Код сотрудника* и *Фамилия*;
- установить мышью подходящую ширину столбца;
- согласиться с предлагаемой подписью столбца подстановок *Фамилия*;
- сохранить таблицу с именем **Договоры**.

Аналогично для подстановки *Кода клиента* и *Кода тура* вызывается **Мастер подстановок**. При этом для *Кода клиента* выбираем поля *Код клиента* и *Название клиента* из таблицы **Клиенты**, а для *Кода тура* — поля *Код тура* и *Страна* из таблицы **Страны**.

19. Просмотрите и проанализируйте уже установленные при работе с **Мастером подстановки** связи в окне **Схема данных**.
20. В окне **Схема данных** двойным щелчком по линии связи откройте окно Связи и установите *Целостность данных*, *Каскадное обновление данных*, *Каскадное удаление данных*.
21. Введите в таблицы 10 разнообразных записей. В таблице **Сотрудники** *существите* ввод заведомо некорректных данных для проверки работоспособности условия на значение.
22. Просмотрите главную таблицу каждой связи (с помощью «+») и вызовите *подчиненную* таблицу для каждой записи.

#### **Форма представления результата:**

Необходимо предоставить схему полученную базу данных с созданными связями между таблицами.

#### **Критерии оценки:**

Оценка «отлично» ставится, если задание выполнено верно.

Оценка «хорошо» ставится, если ход выполнения задания верный, но была допущена одна или две ошибки, приведшие к неправильному ответу.

Оценка «удовлетворительно» ставится, если в работе не получен ответ и приведено неполное выполнение задания, но ход выполнения задания верный

Оценка «неудовлетворительно» ставится, если задание не выполнено или если приведен правильный ответ, но решение отсутствует.

## Тема 03.01.05 Базы данных

### Практическая работа № 3 Проектирование структуры базы данных

**Цель работы:** получение практических навыков по проектированию структуры базы данных по заданной предметной области.

**Выполнив работу, Вы будете:**

*уметь:*

- проектировать базу данных.

**Материальное обеспечение:**

Персональный компьютер, Access, Erwin, методические указания для выполнения практических работ, вариант задания.

**Задание:**

Выполнить проектирование базы данных в соответствии с выданным вариантом.

#### **Краткие теоретические сведения:**

Проектирование базы данных заключается в ее многоступенчатом описании с различной степенью детализации и формализации, в ходе которого производится уточнение и оптимизация структуры базы данных. Проектирование начинается с описания предметной области и задач информационной системы, идет к более абстрактному уровню логического описания данных и далее – к схеме физической (внутренней) модели базы данных. Трех основным уровням моделирования системы – **концептуальному, логическому и физическому** соответствуют три последовательных этапа детализации описания объектов базы данных и их взаимосвязей.

На **концептуальном уровне** проектирования производится смысловое описание информации предметной области, определяются ее границы, производится абстрагирование от несущественных деталей. В результате определяются моделируемые объекты, и их свойства и связи. Выполняется структуризация знаний о предметной области, стандартизируется терминология. Затем строится концептуальная модель, описываемая на естественном языке. Для описания свойств и связей объектов применяют различные диаграммы.

На следующем шаге принимается решение о том, в какой конкретной СУБД будет реализована база данных. **Выбор СУБД** является сложной задачей и должен основываться на потребностях с точки зрения информационной системы и пользователей. Определяющими здесь являются

вид программного продукта и категория пользователей (или профессиональные программисты, или конечные пользователи, или то и другое).

Другими показателями, влияющими на выбор СУБД, являются:

- Удобство и простота использования;
- Качество средств разработки, защиты и контроля базы данных;
- Уровень коммуникационных средств (в случае применения ее в сетях);
- Фирма-разработчик;
- Стоимость.

Каждая конкретная СУБД работает с определенной моделью данных. Под моделью данных понимается способ их взаимосвязи: в виде иерархического дерева, сложной сетевой структуры или связанных таблиц. В настоящее время большинство СУБД использует табличную модель данных, называемую *реляционной*.

На **логическом уровне** производится отображение данных концептуальной модели в логическую модель в рамках той структуры данных, которая поддерживается выбранной СУБД. Логическая модель не зависит от конкретной СУБД и может быть реализована на любой СУБД реляционного типа.

На **физическом уровне** производится выбор рациональной структуры хранения данных и методов доступа к ним, которые обеспечивает выбранная СУБД. На этом уровне решаются вопросы эффективного выполнения запросов к БД, для чего строятся дополнительные структуры, например индексы. В физической модели содержится информация обо всех объектах базы данных (таблицах, индексах, процедурах и др.) и используемых типах данных. Физическая модель *зависит* от конкретной СУБД. Одной и той же логической модели может соответствовать несколько разных физических моделей. Физическое проектирование является начальным этапом реализации базы данных.

#### **Порядок выполнения работы:**

1. Выполнить анализ предметной области.
2. Выполнить анализ данных.
3. Определить набор атрибутов для данной предметной области.
4. Определить набор таблиц.  
При проектировании таблиц рекомендуется руководствоваться следующими основными принципами:
  - каждая таблица должна содержать данные только на одну тему;
  - данные не должны дублироваться.
5. Создать словарь имен.
6. Определить состав и типы полей.

## 7. Создать связи между таблицами.

Темы для самостоятельного проектирования базы данных:

1. Проектирование базы данных для учета домашних финансов
2. Проектирование базы данных кинотеатра
3. Проектирование базы данных для агентства недвижимости
4. Проектирование базы данных риэлторского агентства
5. Проектирование базы данных туристического агентства
6. Проектирование и разработка базы данных соревнований по настольному теннису
7. Проектирование базы данных футбольных клубов
8. Проектирование базы данных центра по продаже автомобилей
9. Проектирование и разработка базы данных книжного магазина
10. Проектирование и разработка базы данных интернет магазина
11. Проектирование базы данных аэропорта
12. Проектирование базы данных книжной библиотеки
13. Проектирование базы данных для начисления стипендии.

### **Форма представления результата:**

Оформленная схема базы данных.

### **Критерии оценки:**

Оценка «отлично» ставится, если задание выполнено верно.

Оценка «хорошо» ставится, если ход выполнения задания верный, но была допущена одна или две ошибки, приведшие к неправильному ответу.

Оценка «удовлетворительно» ставится, если в работе не получен ответ и приведено неполное выполнение задания, но ход выполнения задания верный

Оценка «неудовлетворительно» ставится, если задание не выполнено или если приведен правильный ответ, но решение отсутствует.

## Тема 03.01.05 Базы данных

### Практическая работа № 4 Операции поиска и фильтрации данных

**Цель работы:** получение практических навыков по освоению операций сортировки, поиска и фильтрации данных.

**Выполнив работу, Вы будете:**

*уметь:*

- осуществлять сортировку данных;
- выполнять поиск и замену данных;
- осуществлять фильтрацию данных.

**Материальное обеспечение:**

Персональный компьютер, MSAccess, методические указания для выполнения практических работ.

**Задание:**

Освоить операции сортировки, поиска и фильтрации данных на примере базы данных *Туризм*.

**Краткие теоретические сведения:**

Поиск и представление данных из базы данных – одна из основных задач СУБД. В зависимости от информационной потребности можно использовать простые приемы поиска данных или более сложные, позволяющие формировать непростые критерии отбора.

К простейшим видам поиска относится использование команд *Найти и Заменить*. В условиях поиска могут быть использованы операции сравнения (>, <, <=, >=, =, <>), а также подстановочные символы:

\* - любая цифра или символ. Может быть первым или последние символом текстовой строки. Например, Wh\* - поиск слова What, white и why.

? – любой тестовый символ. Например – B?ll – поиск слова ball, bell и bill.

[ ] – любой один символ из заключенных в скобки. Например – B[ae]ll – поиск слов ball, bell, но не bill.

! – любой один символ, кроме заключенных в скобки. Например - b[!ae]ll – поиск слова billbull, но не bell и ball.

- -любой символ из диапазона. Нужно указывать по возрастанию (от A до Z, но не от Z До A). Например, b[a-c]d – поиск слов bad, bbd и bcd.

# - любая цифра. Например, 1#3 – поиск значений 103, 113, 123.

## **ИСПОЛЬЗОВАНИЕ ФИЛЬТРОВ**

**Фильтр** – это способ показать в окне только те записи базы данных, которые удовлетворяют требованиям пользователя. Фильтры – это однократные запросы, без имени. Они просты в использовании. Можно применять фильтры к таблице, запросу или форме, но фильтруются всегда данные только одной таблицы. В фильтре отображаются все поля.

В СУБД MSAccess несколько видов фильтров.

### **1. Фильтр по выделенному**

Необходимо выделить фрагмент содержимого нужного поля и установить фильтр одним из способов: **Фильтр – Фильтр по выделенному**, контекстное меню - **Фильтр по выделенному**. В результате останутся записи, совпадающие по этому полю или его части.

### **2. Фильтр по форме или изменение фильтра**

При использовании *фильтра по форме* получается свернутая в строку пустая таблица с пиктограммой списка в каждом поле, где можно задать критерий отбора. В критерии можно использовать и логические операторы AND, OR, NOT.

Инструментом сортировки можно найденные записи упорядочить.

Например, если нужно в базе данных **Туризм** просмотреть только те записи, в которых **Дата начала тура** после 15.02.02, то нужно открыть таблицу **Договоры**, **Фильтр – Изменить фильтр**, в этом поле набрать



условие >#15.02.02#, имея в виду, что константы типа Дата/Время заключаются в #. После этого нужно нажать кнопку **Применить фильтр**.

В результате на экране останутся только соответствующие критерию записи.

### 3. **Фильтр по вводу**

Устанавливается при помощи вызова контекстного меню на нужном поле таблицы. Может применяться в таблицах и формах. Позволяет найти записи, удовлетворяющие нескольким условиям одновременно.

### 4. **Расширенный фильтр**

Вызывается командой **Фильтр – Расширенный фильтр**. В приведенном окне бланка фильтра пользователь имеет возможность создать фильтр, введя условия отбора, с помощью которых из всех записей в открытой форме или таблице выделяется подмножество, удовлетворяющее данным условиям. Кроме того, в бланке фильтра задается порядок сортировки для одного или нескольких полей.

### **Порядок выполнения работы:**

1. Откройте базу данных **Туризм**.
2. Откройте таблицу **Сотрудники**.  
Поиск данных
3. Осуществите следующие операции поиска:
  - найдите все записи о служащих в должности «Менеджер»;
  - определите домашний телефон, который начинается на цифру 5;
  - выберите телефоны, содержащие цифру 5;
  - определите фамилии, имеющие вторую букву «а» или «о».Замена данных
4. Замените все должности «Менеджер» на «Менеджер по продажам».  
Сортировка данных в таблицах
5. Отсортируйте фамилии сотрудников по алфавиту.
6. Отсортируйте записи по должностям, а для одинаковых должностей — по фамилиям. Для этого расположите поле **Должность** слева от поля **Фамилия**, выделите оба поля и выполните сортировку.
7. Использование фильтров
8. Откройте таблицу **Сотрудники**.  
Установите фильтры по выделенному (снимая фильтр каждый раз после получения результата):
  - конкретная фамилия (например, Иванов);
  - записи, в которых фамилии заканчиваются на «вич», «ов»;
  - выборка менеджеров по продажам;

- выборка проживающих в одном районе (по первым трем цифрам телефона).
9. Установите фильтрацию данных с помощью исключения данных (вместо включения). Для этого выберите предложенные в предыдущем пункте критерии как исключающие («все, кроме этих» — контекстное меню — **Исключить выделенное**).
10. Установите фильтр по форме:
- фамилии, начинающиеся на «О» или «К»;
  - сотрудники не старше 25 лет;
  - с окладом меньше 1500.
11. Установите фильтр по вводу (контекстное меню на нужном поле таблицы):
- фамилии, начинающиеся на букву А;
  - договоры, заключенные в 2008 году;
  - клиенты, зарегистрированные как групповые.
12. Выберите команду **Фильтр— Расширенный фильтр**:
- конкретная фамилия сотрудника;
  - договор в конкретную страну, оформленный заданным сотрудником (например, «Какие договоры на посещение Испания заключил Сидоров?»);
  - номера телефонов, которые содержат цифру 9.

### **Форма представления результата:**

Отчет по выполненной практической работе

#### **Критерии оценки:**

Оценка «отлично» ставится, если задание выполнено верно.

Оценка «хорошо» ставится, если ход выполнения задания верный, но была допущена одна или две ошибки, приведшие к неправильному ответу.

Оценка «удовлетворительно» ставится, если в работе не получен ответ и приведено неполное выполнение задания, но ход выполнения задания верный

Оценка «неудовлетворительно» ставится, если задание не выполнено или если приведен правильный ответ, но решение отсутствует.

## Тема 03.01.05 Базы данных Практические работы №5 Создание запросов

**Цель работы:** получение практических навыков по созданию запросов к базе данных.

### **Выполнив работу, Вы будете:**

*уметь:*

- создание конструктора запросов;
- задания условий для запросов.

### **Материальное обеспечение:**

Персональный компьютер, MSAccess, методические указания для выполнения практических работ.

### **Задание:**

1. К созданной таблице в практической работе 1 добавить в базу данных таблицу «Группы», состоящую из следующих полей:  
«Название группы» – тип поля должен полностью совпадать с соответствующим полем из таблицы «Основные сведения о студенте» (первичный ключ);

«Факультет» – тестовый, 30 символов (обязательное поле).

«Описание» – МЕМО (необязательное поле к заполнению).

2. Заполнить таблицу «Группы».

3. В таблице «Основные сведения о студенте» значения поля «Название группы» выбиралось из соответствующего поля таблицы «Группы».

4. Создайте запрос, который выводит все данные из таблицы «Основные сведения о студенте» и поле «Факультет» из таблицы «Группы».

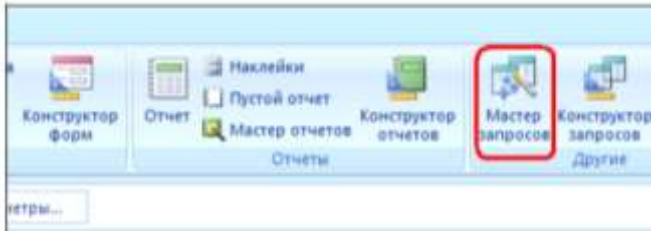
5. Создать запрос, который выводит сведения о студентах, обучающихся на факультете «Информационные технологии».

6. На основе практической работы №5 создать запрос, отображающий студентов определенного года рождения, учащихся на факультете «Информационные технологии». Примечание: взять диапазон – Ваш год рождения  $\pm 1$ .

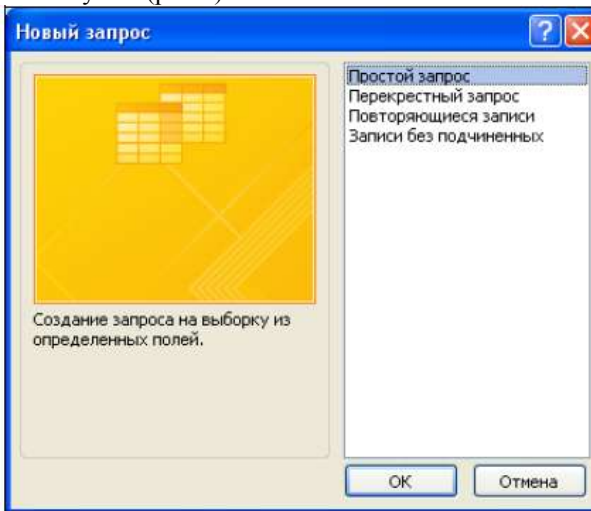
### **Ход работы:**

С помощью Мастера запросов. Наиболее просто создается запрос при помощи Мастера запросов. Рассмотрим создание простого запроса на примере таблицы «Основные сведения о студенте». Чтобы создать простой запрос с помощью Мастера запросов, необходимо:

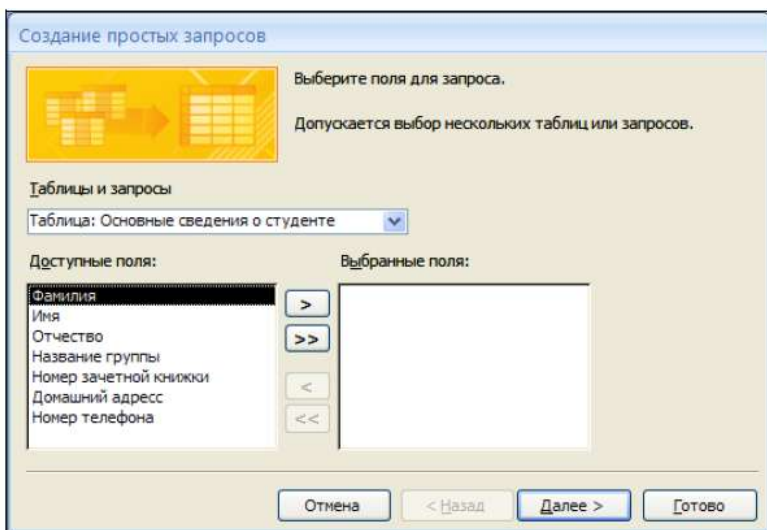
1. В окне базы данных на панели инструментов Создание выбрать значок Мастер запросов (рис.1).



2. При щелчке по этому значку появится диалоговое окно Новый запрос (NewQuery). Выбрать Простой запрос (SimpleQueryWizard) и нажать на кнопку ОК (рис.2).



3. В появившемся окне Создание простых запросов (SimpleQueryWizard) (рис. 3) в поле со списком Таблицы и запросы (Tables/Queries) выбрать таблицу или запрос, которые будут служить источником данных для создаваемого запроса.



4. С помощью стрелок вправо и влево переместить из списка Доступные поля (AvailableFields) в список Выбранные поля (SelectedFields) те поля, которые необходимы в конструируемом запросе. При этом порядок полей в запросе будет соответствовать порядку полей в списке Выбранные поля (SelectedFields). Если нужно включить в запрос все поля, можно воспользоваться кнопкой с двумя стрелками вправо.

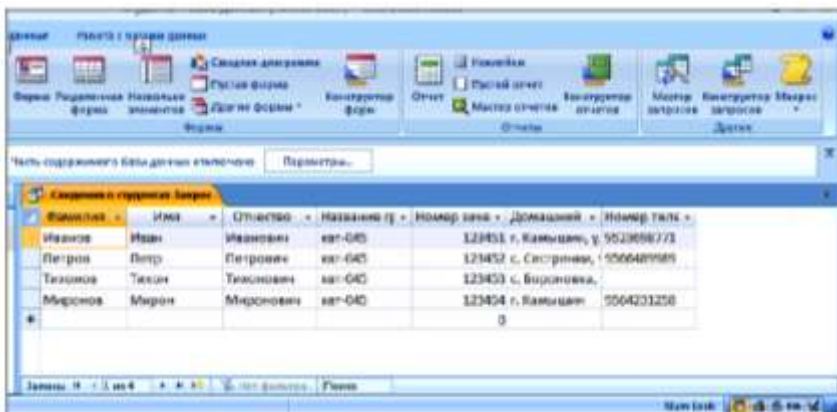
5. Нажать кнопку Далее (Next).

6. Следующее диалоговое окно будет последним. В нем нужно ввести имя создаваемого запроса в поле Задайте имя запроса (Whattitledoyouwanttoyourquery?) (например: «Сведения о студентах Запрос») и выбрать дальнейшие действия: Открыть запрос для просмотра данных (Openthequerytoviewinformation) или Изменить макет запроса (Modifythedyquerydesign).

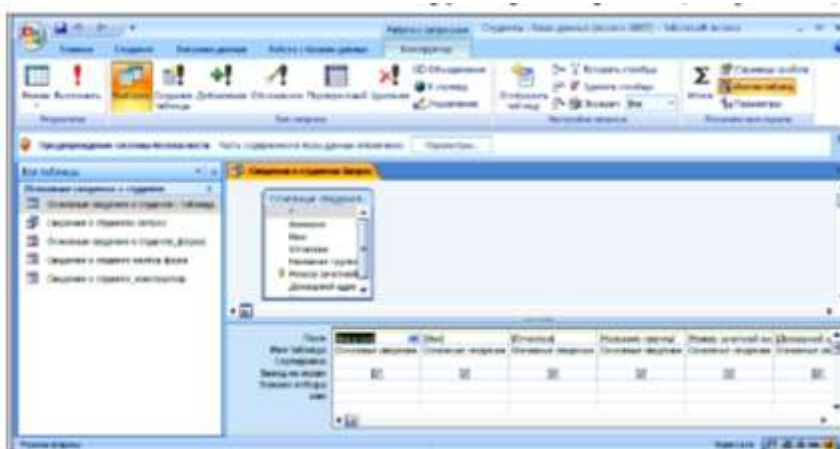
7. При необходимости можно установить флажок Вывести справку по работе с запросом? (DisplayHelponworkingwiththequery) для вывода справочной информации по работе с запросами.

8. Нажать на кнопку Готово (Finish).

По окончании работы Мастера простых запросов в зависимости от выбора способа дальнейшей работы с запросом откроется или окно запроса в режиме просмотра (рис. 4), или окно Конструктора запросов, в котором можно модифицировать запрос (рис. 5).



Создание и изменение запроса с помощью Конструктора запросов. Для изменения уже существующих запросов и для создания новых запросов используется Конструктор запросов. Для того чтобы открыть запрос в режиме Конструктора, выделите в списке один из существующих запросов, например, только что созданный запрос — Сведения о студентах Запрос1, и выберите режим Конструктор (Design) на панели инструментов окна База данных (Database). В итоге появится окно Конструктора запросов (см. рис.5).



В верхней части окна отображается таблица (или несколько таблиц, если запрос многотабличный, однако в нашем случае пока доступна только одна таблица) в том виде, в каком таблицы отображаются в окне

Схема данных (Relationship). Таблицы – источники данных для запроса, будем называть *базовыми* таблицами запроса. В нижней части окна находится *бланк запроса* – таблица, ячейки которой используются для определения запроса. В бланке отображаются все столбцы, включенные в результирующее множество запроса. Для того чтобы просматривать полностью бланк запроса и все исходные таблицы, используют линейки прокрутки. В области панелей инструментов Microsoft Access 2007 отображается панель инструментов Конструктор запросов (QueriesDesign). Эта панель представлена на рис. 6.

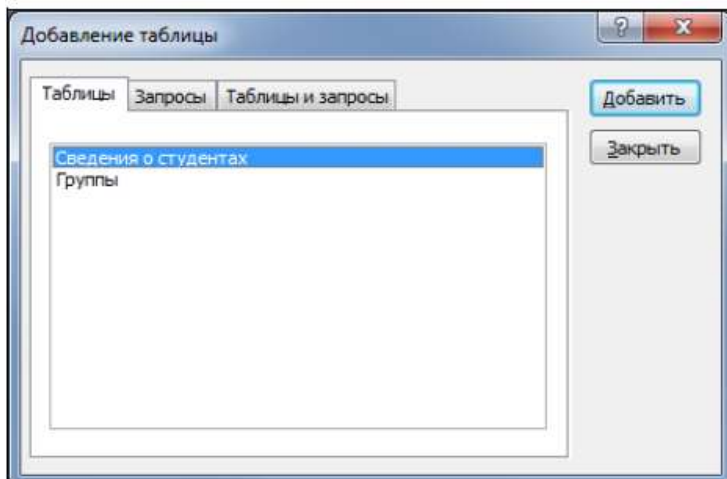


Для создания нового запроса с помощью Конструктора запросов необходимо:

1. На панели инструментов закладки Создание выбрать ярлык Конструктор запросов.

2. В окне Добавление таблицы (ShowTable) (рис. 4.7) выбрать одну или несколько таблиц или запросов для построения нового запроса и нажать кнопку Добавить (Add). В нашем случае добавляем таблицу «Основные сведения о студенте». Для удобства выбора таблиц и запросов в окне существуют следующие вкладки: Таблицы (Tables), на которой отображается список таблиц; Запросы (Queries), на которой отображается список запросов; Таблицы и запросы (Both), на которой отображается список таблиц и запросов вместе.

3. После добавления всех необходимых таблиц нажать кнопку Закрыть (Close) в окне Добавление таблицы (см. рис. 4.7). Все выбранные таблицы оказываются помещенными на верхней панели окна Конструктора запросов. Если таблицы связаны между собой, то есть связи присутствуют явно на схеме данных, то эти связи также отображаются. Если связи на схеме данных не установлены, то Конструктор запросов автоматически устанавливает связи между таблицами, если они содержат поля, которые имеют одинаковые имена и согласованные типы.



*Примечание: Иногда Конструктор устанавливает лишние связи, основываясь только на именах и типах полей. Это может привести к некорректным результатам запроса, поэтому нужно обязательно проверять, как отображаются связи между таблицами в окне Конструктора запросов, и удалить вручную лишние связи. Для этого выделите лишнюю связь, щелкнув по ней левой кнопкой мыши, и нажмите клавишу <Delete>.*

4. Затем нужно указать, какие поля из базовых таблиц будут отображаться в запросе. Включать в запрос можно поля из любой таблицы. Способов включения полей в запрос существует несколько:

Выделите нужное поле в таблице-источнике (можно выделить несколько полей, пользуясь клавишами <Shift> и <Ctrl>). Если требуется включить в запрос все поля базовой таблицы, выделите поле, обозначенное звездочкой (\*). Дважды щелкните левой кнопкой мыши на выделенном поле. При этом в бланке запроса появится столбец, соответствующий выбранному полю. Затем аналогично добавьте другие поля. Столбцы в бланке запроса при этом заполняются слева направо.

Можно подвести указатель мыши к выделенному полю (одному из выделенных полей), нажать на левую кнопку мыши и перетаскивать поле (поля) в нужное место бланка запроса (указатель мыши при этом должен принять вид трех прямоугольников). Последний способ позволяет помещать поля в любое место бланка запроса.

Вместо перетаскивания полей в бланк запроса из таблицы можно просто использовать раскрывающийся список полей в строке Поле (Field) бланка запроса.



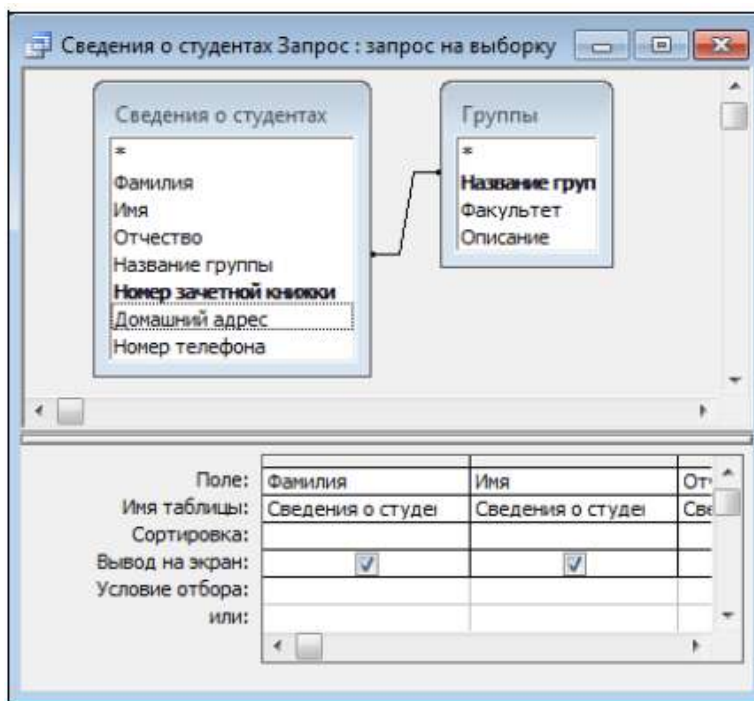
*Примечание: Перед продолжением рекомендуется выполнить 1 и 2 пункты задания на проведение лабораторной работы. Чтобы добавить в запрос еще одну таблицу или другой запрос, необходимо (находясь в режиме Конструкторе запроса):*

1. Нажать кнопку Добавить таблицу на панели инструментов или выполнить команду меню Запрос, Добавить таблицу.

2. В окне Добавление таблицы выбрать вкладку, содержащую требуемые объекты. (Если нужная таблица находится в другой базе данных или другом приложении, необходимо сначала при-соединить эту таблицу к текущей базе данных.).

3. Выбрать имя объекта, добавляемого в запрос.

4. Нажать кнопку Добавить, а затем кнопку Закреть (рис. 8).



Чтобы удалить базовую таблицу из запроса, необходимо выделить ее, щелкнув на любом месте в списке ее полей, и нажать клавишу <Delete>. Чтобы удалить поле из запроса, выделите нужный столбец в бланке запроса, а затем нажмите клавишу <Delete>. Поля в таблице, являющейся результатом запроса, отображаются в том порядке, в котором они следуют в бланке запроса. Если требуется изменить порядок их сле-

дования, то достаточно пере-ставить соответствующим образом столбцы в бланке запроса. В режиме Конструктора запросов можно изменять имена по-лей запроса. Чтобы переименовать поле, необходимо установить курсор в бланке запроса перед первой буквой его имени и ввести новое имя и символ двоеточия. В строке Условие отбора (Criteria) и в строке ИЛИ (Or) указываются условия отбора записей. Такими условиями могут быть логические выражения. Например, (>30), (=Петров'), (=10) и т. п. (рис. 9).

Выражения активно используются в запросах для описания критериев выборки записей. Если критерий отбора очень сложный, можно вставлять до-полнительные строки условий. Для этого необходимо:

1. Выделить строку, которая должна оказаться под вставляемой строкой. Для этого нужно подвести указатель мыши к левой границе строки (он должен превратиться в жирную стрелку, указывающую вправо) и щелкнуть левой кнопкой мыши.

2. Выполнить команду Вставка, Строки (Insert, Rows) или нажать клавишу <Ins>. Новая пустая строка вставляется над выбранной. В нее можно вносить условия отбора.

Для удаления строки условий отбора:

1. Щелкните левой кнопкой мыши в любом месте строки.
2. Выполните команду Правка, Удалить строки (Edit, DeleteRows).

Чтобы установить порядок сортировки записей в запросе, используйте строку Сортировка (Sort). Для каждого поля, по которому должны сортироваться записи, выберите из списка соответствующий порядок сортировки: по возрастанию (Ascending) или по убыванию (Descending). По умолчанию во всех полях запроса устанавливается значение отсутствует (Notsorted). После формирования бланка запроса его можно сохранить, нажав на кнопку Сохранить (Save) на панели инструментов или выполнив команду меню Файл, Сохранить (File, Save). При этом появляется диалоговое окно, в которое нужно ввести имя сохраняемого запроса «Запрос» можно сохранить и закрыв его. Результаты выполнения запроса можно увидеть, переключившись в режим Таблицы (DatasheetView) с помощью кнопки Вид (View) или нажав кнопку Запуск (Run) на панели инструментов. 4.2.

#### **Форма представления результата:**

Предоставить реализованную базу данных с созданными запросами.

#### **Критерии оценки:**

Оценка «отлично» ставится, если задание выполнено верно.

Оценка «хорошо» ставится, если ход выполнения задания верный, но была допущена одна или две ошибки, приведшие к неправильному ответу.

Оценка «удовлетворительно» ставится, если в работе не получен ответ и приведено неполное выполнение задания, но ход выполнения задания верный

Оценка «неудовлетворительно» ставится, если задание не выполнено или если приведен правильный ответ, но решение отсутствует.

## Тема 03.01.05 Базы данных

### Практические работы №6 Создание запросов с вычисляемыми полями

#### Формируемые компетенции:

ПК 3.1. Проводить контроль, диагностику и восстановление работоспособности компьютерных систем и комплексов.

ПК 3.2. Проводить системотехническое обслуживание компьютерных систем и комплексов.

**Цель работы:** получение практических навыков по освоению операций создания запросов с помощью Конструктора.

#### Выполнив работу, Вы будете:

*уметь:*

- создавать запросы с помощью Конструктора;
- создавать запросы на выборку, с вычисляемыми полями, параметрические, итоговые и перекрестные.

#### Материальное обеспечение:

Персональный компьютер, MSAccess, методические указания для выполнения практических работ.

#### Задание:

Разработать запросы для базы данных.

#### Краткие теоретические сведения:

Запрос является объектом базы данных. Он представляет собой сформулированную информационную потребность.

При работе с запросом можно выделить два этапа: формирование (проектирование) и выполнение. При выполнении запроса выбирается информация из всех таблиц базы данных в соответствии с критерием запроса.

В верхней части окна **Конструктора** размещаются нужные таблицы посредством команды **Запрос – Добавить таблицу** или та же команда в контекстном меню. В нижней части окна расположен бланк запроса, информация в него заносится путем перетаскивания нужных полей из таблиц в верхней части окна в строку «поле» или двойным щелчком мыши. При этом имя таблицы в бланке подставляется автоматически.

Наличие «галочки» в строке «Вывод на экран» означает присутствие данного поля в таблице результатов поиска. Критерии запроса устанавливаются

ливаются в строке «Условия отбора» и последующих строках, связанных логическим оператором OR. Все критерии отбора, указанные в одной строке, объединяются оператором AND.

В качестве «Условия отбора» могут быть выражения (вычисляемое поле) даты, текст, которые либо вносятся вручную, либо инструментом, либо с помощью команды контекстного меню **Построить**. Константы типа Дата/Время заключаются в #.

Запросы бывают разных типов: на выборку, на создание, на обновление, на добавление, на удаление, перекрестный, итоговый, параметрический и др. По умолчанию формируется запрос на выборку. Тип запроса может быть преобразован в любой другой командой **Запрос**.

При создании критерия можно использовать инструмент **Построить** или такую же команду контекстного меню для категории «условие отбора».

### **1. Вычисляемые поля в запросах**

С помощью запросов можно задать вычисления над данными и сделать вычисляемое поле новым полем в наборе данных. Для создания нового поля в пустой ячейке строки **Поле** в бланке запроса вводится формула:

**Имя поля: выражение**

Для построения выражений имеется специальное средство – **Построитель** выражений, вызываемый правой кнопкой мыши на поле или кнопкой **Построить**.

В верхней части размещается область ввода. Нижняя содержит три списка для выбора имен полей и функций. В папке **Функции** размещаются встроенные функции, сгруппированные по категориям.

### **2. Параметрические запросы**

Условия запроса могут быть включены непосредственно в бланк запроса, но для того чтобы сделать его более универсальным, можно вместо конкретного значения отбора включить в запрос параметр, т.е. создать параметрический запрос. Для этого в строку «условия отбора» вводится фраза в квадратных скобках, которая будет выводиться в качестве «подсказки» в процессе диалога, например [введите фамилию]. Таких параметров может быть несколько, каждый для своего поля.

### **3. Итоговые запросы**

При выборе данных может понадобиться найти какую-либо функцию, например сумму значений или максимальное значение в поле. Запросы, выполняющие вычисления над группой записей, называются итоговыми. В бланке запроса появится новая строка с наименованием «Групповая операция», в ней содержится слово «Группировка». В этой строке следует указать, какое вычисление необходимо выполнить.

Возможные операции в строке «Групповые операции»:

SUM – сложение;  
AVG – среднее значение;  
MIN – минимальное значение;  
MAX – максимальное значение;  
COUNT – количество записей со значениями (без пустых значений);  
STDEV – стандартное отклонение;  
VAR – дисперсия;  
FIRST – значение в первой записи;  
LAST- значение в последней записи.

#### 4. *Перекрестные запросы*

Особый тип итоговых запросов, представляющих результаты поиска в виде матрицы, называется перекрестным.

Для каждого поля такого запроса может быть выбрана одна из установок: «Заголовки строк», «Заголовки столбцов», «Значение», которое выводится в ячейках таблицы, и «Не отображаются».

Для перекрестного запроса надо обязательно определить хотя бы по одному полю в качестве заголовка строк, заголовка столбцов и значения. Можно использовать дополнительные условия отбора и сортировка.

#### **Порядок выполнения работы:**

##### *Запросы с вычисляемыми полями*

1. Создайте запрос для расчета ведомости заработной платы для сотрудников агентства, включив в нее следующие поля: ***Фамилия сотрудника, Размер оклада, Стаж, Надбавка, Налог, На руки.***

Для поля ***Стаж*** нужно использовать формулу, построенную с помощью кнопки **Построить**, в которой учитывается сегодняшняя дата и **Дата найма** на работу:

$$\text{Стаж} = (\text{Date}() - \text{Сотрудники!ДатаНайма}) / 365$$

Для поля ***Надбавка*** нужно исходить из того, что она составляет 20% от ***Размера оклада***, если ***Стаж*** меньше 5 лет, и 30% — если стаж больше 5 лет:

$$\text{If} ([\text{стаж}] < 5; 0,2 * [\text{Сотрудники!Размер оклада}]; 0,3 * [\text{Сотрудники!Размер оклада}])$$

Поле ***Налог*** рассчитывается как 13% от **Размера оклада**:  
$$[\text{Сотрудники!Размер оклада}] * 0,13$$

Поле **На руки** рассчитывается как:  
$$[\text{Размер оклада}] + [\text{надбавка}] - [\text{налог}].$$

В результате выполнения запроса будет получена новая ведомость.

2. Создайте запрос для определения стоимости путевок корпоративных клиентов, включив в него поля Клиент, Стоимость путевки

Стоимость путевки = Sum(договоры![Цена тура]

\*договоры![Число туристов])

#### Параметрические запросы

3. Сформируйте запрос для выборки всех туров по названию страны.
4. Создайте запрос для получения данных на сотрудников, работающих по турам в конкретную страну.
5. Создайте запрос по всем клиентам, оформившим договоры в определенную страну и регион.

#### Итоговые запросы

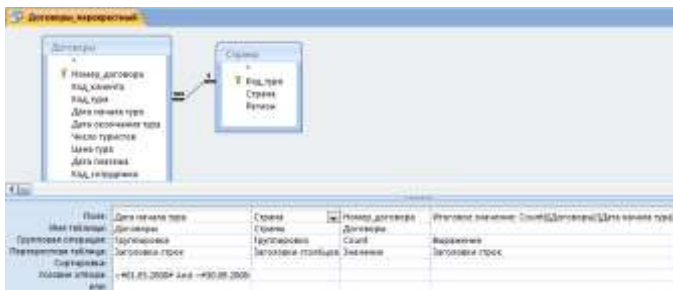
6. Создайте запрос, используя подходящие функции, найдите наибольший и средний размеры цены тура.
7. Создайте запрос для подсчета объема продаж путевок в конкретную страну. Для этого:
  - добавьте в Конструкторе запросов таблицу **ДоговорыСтраны**
  - добавьте в бланк запроса поля **Название страны** (из таблицы Страны) и расчетное поле Цена тура \* Число туристов, которому присвоим название Стоимость путевок;
  - выберите команду **Вид — Групповые операции** в выпадающем списке в строке «Группировка» для поля **Стоимость путевок** установите функцию SUM;
  - запустите запрос и просмотрите результаты.
8. Создайте запрос для определения средней цены и общей суммы туров за 2005 год.
9. Для объединения записей в группы и получения итоговых значений по каждой группе используется опция «Группировка». Создайте новый запрос для базы данных **Туризм**, в котором определите общие суммы продаж путевок по годам:
  - добавьте таблицу Договоры в окно запроса;
  - в первый столбец поместите поле Год начала тура, рассчитав его с помощью функции **Year**, во второй — сумма общих продаж путевок — Sum(договоры![Цена тура]\*договоры![Число туристов]);
  - установите для первого столбца в строке «Групповая операция» — «Группировка», для второго — **Выражение**;
  - выполните запрос и прокомментируйте результаты.
10. Дополните предыдущий запрос критерием, который включает в выборку только те заказы, которые оформлены в 2008 г.

и позже. Для этого следует добавить в бланк запроса поле *Дата заказа* из таблицы «Заказы». В строке «Групповая операция» выберите пункт «Условие». В строке «Условие отбора» укажите условие на дату. Обязательно снимите флажок «Вывод на экран» для этого поля. Выполните запрос и проанализируйте результаты.

11. Выберите записи, стоимость перевозок в которых превышает заданное значение.
12. Найдите записи, в которых для каждого вида доставки было оформлено более 5 заказов («Доставка» — «Группировка», Код заказа – COUNT, «Условие отбора» в поле *Кодзаказа*>=5).

#### Перекрестные запросы

13. Составьте запрос для выяснения: сколько туров организовано в каждую страну в конкретный регион.
14. Составьте перекрестный запрос по теме: сколько туров начались с мая по сентябрь 2008 г. в разные страны.



Составьте перекрестный запрос для определения предпочтений клиентов разным регионам (сколько клиентов, в каком регионе побывали).

#### **Форма представления результата:**

Отчет по выполненной практической работе.



## Тема 03.01.05 Базы данных

### Практические работы №7 Модификация базы данных с помощью запросов на изменение

#### Формируемые компетенции:

ПК 3.1. Проводить контроль, диагностику и восстановление работоспособности компьютерных систем и комплексов.

ПК 3.2. Проводить системотехническое обслуживание компьютерных систем и комплексов.

**Цель работы:** получение практических навыков по освоению запросов на создание, обновление, добавление и удаление.

#### Выполнив работу, Вы будете:

*уметь:*

- создавать запросы на обновление, добавление и удаление.

#### Материальное обеспечение:

Персональный компьютер, MSAccess, методические указания для выполнения практических работ.

#### Задание:

Разработать модифицирующие запросы к базе данных.

#### Краткие теоретические сведения:

##### *Модификация базы данных с помощью запросов на изменение*

##### ➤ **Запрос на обновление**

Запрос этого типа используется при необходимости внесения изменений во множество записей базы данных, поэтому целесообразно сделать резервную копию таблицы.

Выполняется этот вид запроса в два этапа: сначала проверяется правильность отбора обновляемых записей с помощью запроса на выборку, затем он преобразуется в запрос на обновление и выполняется повторно.

При обновлении полей следует иметь в виду, что если при проектировании таблицы в свойствах поля было указано «условие на значение», то при обновлении этого поля условие может быть нарушено, чего не допустит MSAccess. Поэтому нужно: или изменить условие на значение, или удалить это условие в Конструкторе.

##### ➤ **Запрос на добавление**

Периодически убирая в архивные таблицы «старые» записи, можно увеличить быстродействие основных частей и улучшить обзорность базы данных.

Кроме того, при необходимости добавить данные в таблицу базы данных из другой базы можно также использовать запросы на добавление.

➤ **Запрос на удаление**

«Старые» или неиспользуемые записи таблиц можно удалить, но обязательно сначала произвести выборку и проверить ее. Целесообразно сделать копию.

**Порядок выполнения работы:**

Запрос на создание

1. Создайте обобщенную таблицу **Договоры по странам**, включив в нее следующие поля:

Из таблицы **Договоры**: **Номер договора; Название клиента**

Из таблицы **Страны**: **Название страны; Регион**.

Для этого:

- Создайте запрос на выборку этих данных, выполните его и проверьте результаты;
- Если результаты корректны, то поменяйте статус у запроса: **Запрос – Создания таблицы** – укажите новое имя таблицы **Договоры по странам**;
- Выполните запрос с новым статусом еще раз;
- Перейдите на вкладку **Таблицы** и убедитесь, что появилась новая таблица. Просмотрите ее.

Запрос на обновление

2. Увеличьте **Размер оклада** у менеджеров по продажам на 15%.

Для этого:

- Составьте новый запрос на выборку, включив в него поля **Фамилия, Должность и Размер оклада**;
- Проверьте составленный запрос;
- Видоизмените запрос, установив ему статус «**Обновление**» (**Запрос – Обновление**). В появившейся в бланке запроса строке «**Обновление**» для поля **Размер оклада** внесите с помощью **Построить** выражение [Размер оклада]\*1,15;
- Выполните запрос, подтвердите обновление; сохраните запрос, дав ему имя и обратив внимание на появившийся значок у его имени; просмотрите результаты.

Запрос на добавление

3. Создайте путем копирования дубликат таблицы **Договоры** без данных, назвав ее **Договоры2008 года**. Для этого в контекстном меню для таблицы **Договоры** выберите **Копировать**, затем вы-

полните команду *Вставить*, в параметрах вставки укажите «*Только структуру*». Просмотрите таблицу **Договоры 2008 года** – она должна быть пустой и иметь такую же структуру, как и таблица **Договоры**.

4. Отберите в таблице **Договоры 2008 года** записи обо всех договорах этого года.

Для этого:

- Создайте запрос на выборку, включив в него все поля таблицы **Договоры** в любой последовательности, и критерий по дате, выполните его для проверки правильности;
- Измените статус запроса на «*Добавление*», в появившемся окне задайте имя таблицы для добавления **Договоры 2008 года**, обратите внимание на появление строки «*Добавление*» в бланке запроса;
- Выполните запрос и подтвердите добавление; просмотрите результаты архивации и сохраните запрос, обратив внимание на значок у его имени.

Запросна удаление

5. Удалите из таблицы **Договоры** записи о договорах 2008 года, используя копию сохраненного запроса на добавление в таблицу **Договоры 2008 года**, изменив его статус на «*Удаление*».

#### **Форма представления результата:**

Отчет по выполненной практической работе.

**Тема 03.01.05 Базы данных**  
**Практические работы №8 Работа с формами**

**Цель работы:** получение практических навыков по освоению операций создания форм разными способами.

**Выполнив работу, Вы будете:**

*уметь:*

- создавать формы;
- добавлять различные элементы управления на форму.

**Материальное обеспечение:**

Персональный компьютер, MSAccess, методические указания для выполнения практических работ.

**Задание:**

Разработать формы для базы данных.

**Порядок выполнения работы:**

- 1) Создать формы для ввода данных с последующей их модификацией.
- 2) Создать главную форму «Заставка».

**Ход 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.



Kod_kaf	<input type="text" value="1"/>
Name_kaf	<input type="text" value="Физика"/>
Nom_tel	<input type="text" value="222222"/>
Nom_auditoria	<input type="text" value="201"/>
Col_sotr	<input type="text"/>
Zav_kaf	<input type="text" value="Фирсов.В.В."/>

### 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.

Открывается вкладка Конструктор - панель элементов, которая позволяет создавать элементы управления на форме и осуществлять необходимые действия при конструировании:

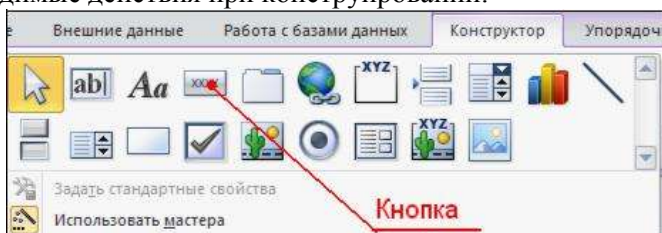



Рис. Элементы управления

Выберите на панели инструментов и активируйте Кнопку.

- Создайте заголовок формы. Для этого выберите кнопку  – Надпись, щелкнув по ней, расширьте область заголовка формы и введите в поле надпись *База данных«MainForm»*. 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. Выберите форму *Kafedra*, которая будет открываться этой кнопкой


Далее, оставьте переключатель в положении «Открыть форму и показать все записи»

В следующем окне поставьте переключатель в положение Текст.

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

**САМОСТОЯТЕЛЬНО:** Добавьте кнопку выхода или закрытия главной формы.

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, выберите Свойства. Открывается диалоговое окно. Во вкладке Макет добавьте фон или рисунок.

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

**Форма представления результата:**

Предоставить реализованную базу данных с сохраненными формами.

**Критерии оценки:**

Оценка «отлично» ставится, если задание выполнено верно.

Оценка «хорошо» ставится, если ход выполнения задания верный, но была допущена одна или две ошибки, приведшие к неправильному ответу.

Оценка «удовлетворительно» ставится, если в работе не получен ответ и приведено неполное выполнение задания, но ход выполнения задания верный

Оценка «неудовлетворительно» ставится, если задание не выполнено или если приведен правильный ответ, но решение отсутствует.



## Тема 03.01.05 Базы данных

### Практические работы №9 Применение отчетов для наглядного отображения данных

**Цель работы:** получение практических навыков по освоению операций создания отчетов.

**Выполнив работу, Вы будете:**

*уметь:*

- создавать отчеты;
- создавать итоговые отчеты.

**Материальное обеспечение:**

Персональный компьютер, MSAccess, методические указания для выполнения практических работ.

**Задание:**

Разработать отчеты для базы данных.

**Краткие теоретические сведения:**

Для конечного пользователя данные, хранимые в таблицах и запросах, могут быть оформлены в виде отчета. Отчет позволит представить информацию на печать в желаемом формате. Его можно дополнить рисунками и графиками, которые сделают данные более привлекательными. В отчете легче всего группировать и сортировать данные.

Существует два основных режима работы с отчетами:

- *Конструктор* – для создания новых отчетов и изменения существующих;
- *Предварительный просмотр* – внешний вид отчета при печати.

К разделам отчета относятся – область данных, где размещаются записи из источника данных, заголовок и примечание отчета, которые только однажды размещаются соответственно в начале и в конце отчета. Дополнительные разделы **Заголовок** и **Примечание группы** связаны с группировкой записей. Количество этих разделов – по числу уровней группировки.

Группа – это набор записей, отобранных по определенному критерию. Группа состоит из заголовка, включенных в нее записей и примечания. Группировка позволяет разбить записи на логически группы и напечатать вводную и итоговую информацию для каждой.

Группировка может быть:

- для чисел - по десяткам, сотням и другим диапазонам значений, которые задаются в свойстве **Интервал**;
- для текстовых полей – по первой букве, по двум первым буквам и т.д.
- для полей типа Дата – по годам, кварталам, месяцам и т.д.

В отчетах, как и формах, можно использовать вычисляемые поля, которые в режиме **Мастера** устанавливаются с помощью кнопки **Итоги**, а в режиме **Конструктора** кнопкой.

Почтовые наклейки являются разновидностью отчета.

### Порядок выполнения работы:

1. Создайте отчет **Ведомость зарплаты** на основе запроса **Расчет зарплаты**. Включите все поля запроса. Установите группировку по полю **Должность**. Подведите итоги по полям **Оклад**, **Надбавка**, **Налоги Наруки**. Всем денежным полям назначьте формат денежный
2. Составьте для базы данных **Туризм** отчет, в котором указаны фамилии сотрудников, а для каждого сотрудника — договоры, которые он заключил, сведения о клиенте, стоимости тура. Сохраните отчет с именем «**Отчет по сотрудникам**».
3. Создайте отчет «**Общая стоимость договоров по странам**», в котором будут использованы вычисляемые поля. Для этого следует выполнить указанные ниже действия.
  - создайте отчет для таблицы **Договоры** произведите группировку по полю **Код тура**.
  - откройте этот отчет в режиме **Конструктора**;
  - подведите итог по каждой группе (по каждому туру). Для этого:
  - добавьте область итогов по группе в отчет **Вид — Сортировка/Группировка — Примечание группы — Да**;
  - в появившейся области «**Примечания группы Код тура**» вставьте новое поле;
  - в свойствах названия поля укажите подпись «**Общая сумма договоров**»;
  - в свойствах поля **Все — Данные** с помощью **Построить** сформируйте формулу  

$$=SUM([Цена тура] * [Число туристов])$$
  - на этой же вкладке установите **Формат** поля **Основной**.
4. Задайте в макете **Цвет текста** — яркий.
5. Просмотрите полученный отчет.
6. Измените отчет для подсчета общих итогов по всем сотрудникам.

7. На основе таблицы **Сотрудники** создайте отчет с группировкой по годам рождения сотрудников: каждое десятилетие должно обрывать группу (1950—1959, 1960—1969 и т.д.).

Для этого нужно задать группировку по полю **Дата рождения** установить свойства: *группировка* — по годам, *интервал* — 10. В разделе *Область данных* разместить все поля и вычисляемое поле

**Возраст** = Round(((Date()-[ДатаРождения])/365)).

В разделе *Заголовок группы* поместить:

Годы рождения с [МинГр] по [МаксГр]

, где «Годы рождения с» и «по» — надписи

[МинГр] и [МаксГр] — вычисляемые поля:

[МинГр] = (Min(DatePart(«уууу»;[ДатаРождения]))\10) \* 10

DatePart(«уууу»;[ДатаРождения]) — возвращает год даты,

Min() — возвращает минимальный год рождения в группе.

Целочисленное деление и последующее умножение на 10 дает значение года, кратное 10.

[МаксГр] = [МинГр] + 9

8. Создайте почтовые наклейки на основе таблицы **Клиенты**. Включите в наклейки поля **Фамилия** и **Адрес** клиента, а также произвольный текст.

### **Форма представления результата:**

Отчет по выполненной практической работе.

#### **Критерии оценки:**

Оценка «отлично» ставится, если задание выполнено верно.

Оценка «хорошо» ставится, если ход выполнения задания верный, но была допущена одна или две ошибки, приведшие к неправильному ответу.

Оценка «удовлетворительно» ставится, если в работе не получен ответ и приведено неполное выполнение задания, но ход выполнения задания верный

Оценка «неудовлетворительно» ставится, если задание не выполнено или если приведен правильный ответ, но решение отсутствует.

## Тема 03.01.05 Базы данных

### Практические работы №10 Экспорт и импорт данных.

**Цель работы:** получение практических навыков по освоению операций экспорта и импорта данных.

**Выполнив работу, Вы будете:**

*уметь:*

- экспортировать данные из базы данных;
- импортировать данные в базу данных.

**Материальное обеспечение:**

Персональный компьютер, MSAccess, методические указания для выполнения практических работ.

**Задание:**

Выполнить экспорт и импорт данных.

**Краткие теоретические сведения:**

MSAccess позволяет импортировать данные из других прикладных программ, таких, как MSExcel, dBase, MSFoxPro, Lotus 1-2-3, команду **Внешние данные –Импорт**. Запускаемый при этом **Мастер Импорта** создаст импортируемого файла таблицу MSAccess.

Для сохранения объекта в файле другого приложения, например MSWord, нужно выделить объект и выполнить команду **Связи с Office – Публикация в MSWord**.

Если нужно использовать таблицу в MSAccess для создания составного документа с помощью функции слияния, то используется команда **Связи с Office – Слияние с MSWord**.

Экспорт таблиц MSAccess можно осуществлять в текстовый файл, файл электронных таблиц или баз данных других форматов командами **Экспорт** –и указать тип файла, в который нужно преобразовать данные. Если есть необходимость экспортировать таблицу в формат MSExcel, то можно использовать команду **Связи с Office – Анализ данных в MicrosoftExcel**.

При желании можно установить связь с таблицей из другой программы для просмотра и обработки имеющихся там данных.

**Порядок выполнения работы:**

Экспорт данных в Excel

1. Экспортируйте **Сотрудники** таблицу в файл типа Excel.

2. Запустите Excel и откройте полученный файл. При необходимости поменяйте шрифт. Обратите внимание на наличие установленного системой примечания в первой ячейке. Измените подпись рабочего листа на «**Адреса**».  
*Импорт данных в Access*
3. Создайте с Excel на основе таблицы *Адреса* новую таблицу *Картоотека адресов*, исключив из таблицы *Адреса* поля *Код сотрудника*, *Должность*, *Размероклада*, *ДатаНайма*. Сохраните ее в файле с именем *Address.xls*.
4. Импортируйте ее (создайте на ее основе новую таблицу) в базу данных **Туризм**.
5. Присвойте новой таблице имя «**Адреса**».
6. Проверьте наличие новой таблицы на вкладке **Таблицы**. Откройте и просмотрите ее.
7. Закройте базу данных.  
*База данных как источник при слиянии документов в MSWord*
8. Для всех клиентов, которые являются групповыми, нужно подготовить и разослать письмо с сообщением о новом открывающемся туре в экзотическую страну.
9. Откройте *Word*. Создайте в окне следующее письмо:

Адрес: «Адрес»

Телефон: «Телефон»

Получатель «Клиент»

Уважаемый «Контактное лицо!»

Спешим информировать Вас, что наша фирма с 1 июня 2009 года открывает новый маршрут в экзотическую страну Острова Зеленого Мыса. Вас ждут приключения и неожиданности в роскошном природном оазисе Африки. Приглашаем Вас принять участие.

Менеджер отдела продаж Петров А.А.

10. Выделенные и заключенные в кавычки поля должны соответствовать полям таблицы **Клиенты**.
11. Отправьте каждому групповому клиенту созданное письмо, осуществив слияние документов – текста письма и атрибутов адреса, взятых из базы данных. Для этого в окне *Word* выполните:
  - **Письма и рассылки – Мастер слияния;**
  - Далее нужно следовать указаниям *Мастера слияния* в нижней части окна;
  - **Источник данных: Получить данные – Выбор получателей** – выберите базу данных **Туризм** – в нем таблицу **Клиенты**;

- Для выбора из таблицы только групповых клиентов установите автофильтр по **Признаку группы**;
- Посредством кнопки **Другие Элементы** внесите в письмо на место названий, заключенных в кавычки, соответствующие поля из таблицы **Клиенты**;
- Осуществите слияние данных, используя кнопку **Просмотр писем** панели инструментов **Слияние**.

### **Форма представления результата:**

Сохранённый отчет в базе данных.

### **Критерии оценки:**

Оценка «отлично» ставится, если задание выполнено верно.

Оценка «хорошо» ставится, если ход выполнения задания верный, но была допущена одна или две ошибки, приведшие к неправильному ответу.

Оценка «удовлетворительно» ставится, если в работе не получен ответ и приведено неполное выполнение задания, но ход выполнения задания верный

Оценка «неудовлетворительно» ставится, если задание не выполнено или если приведен правильный ответ, но решение отсутствует.

## Тема 03.01.05 Базы данных

### Практические работы №11 Организация простейших SQL-запросов

**Цель работы:** получение практических навыков по освоению операций создания запросов с помощью языка SQL.

**Выполнив работу, Вы будете:**

*уметь:*

- создавать запросы с помощью SQL;

**Материальное обеспечение:**

Персональный компьютер, MSAccess, методические указания для выполнения практических работ.

**Задание:**

Необходимо создать запросы при помощи язык структурированных запросов SQL.

**Краткие теоретические сведения:**

Разработать запросы для базы данных.

Запрос – это оператор, который посылает команду Системе Управления Базой Данных (СУБД) произвести манипуляцию или отобразить определенную информацию. Все запросы по выборке данных в SQL конструируются с помощью оператора SELECT. Он позволяет выполнять довольно сложные проверки и обработку данных.

Запрос может выводить данные из определенного столбца или из всех столбцов таблицы. Чтобы создать простейших SELECT запрос, необходимо указать имя столбца и название таблицы.

Синтаксисоператора SELECT

```
SELECT column_list
```

```
FROM table_name
```

```
[WHERE условие]
```

```
[GROUP BY условие]
```

```
[HAVING условие]
```

```
[ORDER BY условие]
```

**Порядок выполнения работы:**

1. Показать все содержимое таблицы EMP.

2. Выбрать всех сотрудников с номерами отделов, в которых они работают и кодами их руководителей.
3. Выбрать имена служащих, их годовой доход и премию. При расчете годового дохода использовать арифметическую операцию – умножение.
4. Выбрать номера отделов (deptno), должности (job) служащих этих отделов и месячную зарплату (sal). Отсортировать результат в обратном порядке по номеру отдела.

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	17.12.1980	800		20
7499	ALLEN	SALESMAN	7698	20.02.1981	1600	300	30
7521	WARD	SALESMAN	7698	22.02.1981	1250	500	30
7566	JONES	MANAGER	7839	02.04.1981	2975		20
7654	MARTIN	SALESMAN	7698	28.08.1981	1250	1400	30
7698	BLAKE	MANAGER	7839	01.05.1981	2850		30
7782	CLARK	MANAGER	7839	09.06.1981	2450		10
7788	SCOTT	ANALYST	7566	09.12.1982	3000		20
7839	KING	PRESIDENT		17.11.1981	5000		10
7844	TURER	SALESMAN	7698	08.09.1981	1500	0	30
7876	ADAMS	CLERK	7788	12.12.1983	1100		20
7900	JAMES	CLERK	7698	03.12.1981	950		30
7902	FORD	ANALYST	7566	03.12.1981	3000		20
7934	MILLER	CLERK	7782	23.12.1982	1300		10

### Форма представления результата:

Предоставить сохраненные запросы в текстовом документе.

### Критерии оценки:

Оценка «отлично» ставится, если задание выполнено верно.

Оценка «хорошо» ставится, если ход выполнения задания верный, но была допущена одна или две ошибки, приведшие к неправильному ответу.

Оценка «удовлетворительно» ставится, если в работе не получен ответ и приведено неполное выполнение задания, но ход выполнения задания верный

Оценка «неудовлетворительно» ставится, если задание не выполнено или если приведен правильный ответ, но решение отсутствует.



## Тема 03.01.05 Базы данных

### Практические работы №12 Выполнение SQL запросов с параметрами.

**Цель работы:** получение практических навыков по освоению операций создания запросов с помощью языка SQL.

**Выполнив работу, Вы будете:**

*уметь:*

- создавать запросы с помощью SQL;

**Материальное обеспечение:**

Персональный компьютер, MSAccess, методические указания для выполнения практических работ.

**Задание:**

Необходимо создать запросы при помощи язык структурированных запросов SQL.

**Краткие теоретические сведения:**

Разработать запросы для базы данных.

Запрос – это оператор, который посылает команду Системе Управления Базой Данных (СУБД) произвести манипуляцию или отобразить определенную информацию. Все запросы по выборке данных в SQL конструируются с помощью оператора SELECT. Он позволяет выполнять довольно сложные проверки и обработку данных.

Запрос может выводить данные из определенного столбца или из всех столбцов таблицы. Чтобы создать простейших SELECT запрос, необходимо указать имя столбца и название таблицы.

Синтаксисоператора SELECT

```
SELECT column_list
```

```
FROM table_name
```

```
[WHERE условие]
```

```
[GROUP BY условие]
```

```
[HAVING условие]
```

```
[ORDER BY условие]
```

**Порядок выполнения работы:**

По таблице EMP, представленная на рисунке 1 создать запросы:

1. Получить перечень должностей в каждом отделе и отсортировать результат по возрастанию номеров отделов. В 20 и 30 отделе присутствуют служащие с одинаковыми должностями. Исключить повторения
2. Выбрать всех служащих работающих клерками (CLERK).
3. Выбрать служащих, номер руководителя которых 7902 или 7566.
4. Выбрать служащих, оклад которых больше или равен 1000 и меньше или равен 2000.
5. Выбрать служащих, оклад которых больше или равен 1000 и меньше или равен 2000 и работающих клерками.
6. Найдите всех служащих, имена которых содержат комбинации символов TH или LL.
7. Покажите всех служащих, имеющих руководителя.
8. Покажите всех служащих работающих клерками и аналитиками.

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7566	SMITH	CLERK	7902	17.12.1980	800		20
7499	ALLEN	SALESMAN	7698	20.02.1981	1600	300	30
7521	WARD	SALESMAN	7698	22.02.1981	1250	500	30
7566	JONES	MANAGER	7839	02.04.1981	2975		20
7654	MARTIN	SALESMAN	7698	28.08.1981	1250	1400	30
7698	BLAKE	MANAGER	7839	01.05.1981	2850		30
7782	CLARK	MANAGER	7839	09.06.1981	2450		10
7788	SCOTT	ANALYST	7566	09.12.1982	3000		20
7839	KING	PRESIDENT		17.11.1981	5000		10
7844	TURER	SALESMAN	7698	08.09.1981	1500	0	30
7876	ADAMS	CLERK	7788	12.12.1983	1100		20
7900	JAMES	CLERK	7698	03.12.1981	950		30
7902	FORD	ANALYST	7566	03.12.1981	3000		20
7934	MILLER	CLERK	7782	23.12.1982	1300		10

### Форма представления результата:

Предоставить сохраненные запросы в текстовом документе.

### Критерии оценки:

Оценка «отлично» ставится, если задание выполнено верно.

Оценка «хорошо» ставится, если ход выполнения задания верный, но была допущена одна или две ошибки, приведшие к неправильному ответу.

Оценка «удовлетворительно» ставится, если в работе не получен ответ и приведено неполное выполнение задания, но ход выполнения задания верный

Оценка «неудовлетворительно» ставится, если задание не выполнено или если приведен правильный ответ, но решение отсутствует.

**Практические работы №13,14 Использование функций.**

**Цель работы:** получение практических навыков по освоению операций создания запросов с помощью SQL.

**Выполнив работу, Вы будете:**

*уметь:*

- создавать итоговые запросы с помощью SQL.

**Материальное обеспечение:**

Персональный компьютер, MSAccess, методические указания для выполнения практических работ.

**Задание 1:**

Разработать запросы для базы данных на языке SQL.

**Краткие теоретические сведения:**

Агрегирующие функции позволяют получать из таблицы сводную (агрегированную) информацию, выполняя операции над группой строк таблицы. Для задания в SELECT-запросе агрегирующих операций используются следующие ключевые слова:

- COUNT определяет количество строк или значений поля, выбранных посредством запроса и не являющихся NULL-значениями;
- SUM вычисляет арифметическую сумму всех выбранных значений данного поля;
- AVG вычисляет среднее значение для всех выбранных значений данного поля;
- MAX вычисляет наибольшее из всех выбранных значений данного поля;
- MIN вычисляет наименьшее из всех выбранных значений данного поля.

Предложение GROUPBY позволяет группировать записи в подмножества, определяемые значениями какого-либо поля, и применять агрегирующие функции уже не ко всем записям таблицы, а отдельно к каждой сформированной группе.

При необходимости часть сформированных с помощью GROUPBY групп может быть исключена с помощью предложения HAVING.

Предложение HAVING определяет критерий, по которому группы следует включать в выходные данные, по аналогии с предложением WHERE, которое осуществляет это для отдельных строк.

В условии, задаваемом предложением HAVING, указывают только поля или выражения, которые на выходе имеют единственное значение для каждой выводимой группы.

### **Порядок выполнения работы:**

#### ***По таблице EMP.***

1. Составить запрос для получения минимальной, максимальной и средней зарплаты в компании.
2. Составить запрос для получения максимальной зарплаты по каждой должности.
3. Составить запрос для подсчёта количества менеджеров, работающих в компании.
4. Составить запрос, вычисляющий разницу между наибольшим и наименьшим окладами в компании.
5. Составить запрос, позволяющий найти отделы, в которых работает более трёх служащих.
6. Составьте запрос, позволяющий показать, что коды служащих (столбец EMPNO) уникальны.

### **Задание 2:**

Разработать запросы для базы данных на языке SQL.

#### ***По «Учебной базе данных».***

1. Составить запрос для подсчёта количества студентов, сдававших экзамен по предмету обучения с идентификатором, равным 22 по таблице EXAMMARKS.
2. Составить запрос, который выполняет выборку для каждого студента значения его идентификатора и минимальной из полученных им оценок по таблице EXAM MARKS, используя предложение GROUP BY.
3. Составить запрос, выполняющий вывод фамилии первого в алфавитном порядке (по фамилии) студента, фамилия которого начинается на букву «П» по таблице STUDENT.
4. Составить запрос, который выполняет вывод (для каждого предмета обучения) наименования предмета и максимального значения номера семестра, в котором этот предмет преподаётся по таблице SUBJECT.
5. Составить запрос для определения количества студентов, сдававших каждый экзамен.
6. Составить запрос для определения количества студентов, проживающих в Воронеже.
7. Составить запрос, выполняющий вывод номера студента, фамилию студента и стипендию, увеличенную на 20%. Выходные

- данные упорядочить по значению последнего столбца(величине стипендии).
8. Составить запрос, выполняющий вывод списка предметов обучения в порядке убывания семестров. Поле семестра в выходных данных должно быть первым, за ним должны следовать имя предмета обучения и идентификатор предмета.
  9. Составить запрос, который выполняет вывод суммы баллов всех студентов для каждой даты сдачи экзаменов и представляет результаты в порядке убывания этих сумм.

**Форма представления результата:**

Предоставить сохраненные запросы в текстовом документе.

**Критерии оценки:**

Оценка «отлично» ставится, если задание выполнено верно.

Оценка «хорошо» ставится, если ход выполнения задания верный, но была допущена одна или две ошибки, приведшие к неправильному ответу.

Оценка «удовлетворительно» ставится, если в работе не получен ответ и приведено неполное выполнение задания, но ход выполнения задания верный

Оценка «неудовлетворительно» ставится, если задание не выполнено или если приведен правильный ответ, но решение отсутствует.

## Тема 03.01.05 Базы данных

### Практические работы №15 Использование вложенных подзапросов.

**Цель работы:** получение практических навыков по освоению операций создания запросов с помощью языка SQL.

**Выполнив работу, Вы будете:**

*уметь:*

- создавать подзапросы с помощью SQL;

**Материальное обеспечение:**

Персональный компьютер, MSAccess, методические указания для выполнения практических работ.

**Задание:**

Разработать запросы для базы данных на языке SQL.

**Краткие теоретические сведения:**

Вложенный запрос – это запрос, который находится внутри другого SQL запроса и встроен внутри условного оператора WHERE. Данный вид запросов используется для возвращения данных, которые будут использоваться в основном запросе, как условие для ограничения получаемых данных.

**Порядок выполнения работы:**

Разработать запросы для базы данных.

По таблице EMP, представленная на рисунке 1 создать запросы:

1. Найти служащих, получающих самую маленькую заработную плату в компании.
2. Выбрать сотрудников отдела продаж (SALES), занимающих ту же должность, что и служащий по фамилии SMITH.
3. Получить список сотрудников отдела 30, получающих зарплату больше минимальной в 20 отделе.
4. Получить список сотрудников, получающих зарплату больше любого, работающего в отделе 30.

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
736	SMITH	CLERK	7902	17.12.1980	800		20
7499	ALLEN	SALESMAN	7698	20.02.1981	1600	300	30
7521	WARD	SALESMAN	7698	22.02.1981	1250	500	30
7566	JONES	MANAGER	7839	02.04.1981	2975		20
7654	MARTIN	SALESMAN	7698	28.08.1981	1250	1400	30
7698	BLAKE	MANAGER	7839	01.05.1981	2850		30
7782	CLARK	MANAGER	7839	09.06.1981	2450		10
7788	SCOTT	ANALYST	7566	09.12.1982	3000		20
7839	KING	PRESIDENT		17.11.1981	5000		10
7844	TURER	SALESMAN	7698	08.09.1981	1500	0	30
7876	ADAMS	CLERK	7788	12.12.1983	1100		20
7900	JAMES	CLERK	7698	03.12.1981	950		30
7902	FORD	ANALYST	7566	03.12.1981	3000		20
7934	MILLER	CLERK	7782	23.12.1982	1300		10

### Форма представления результата:

Предоставить сохраненные запросы в текстовом документе.

### Критерии оценки:

Оценка «отлично» ставится, если задание выполнено верно.

Оценка «хорошо» ставится, если ход выполнения задания верный, но была допущена одна или две ошибки, приведшие к неправильному ответу.

Оценка «удовлетворительно» ставится, если в работе не получен ответ и приведено неполное выполнение задания, но ход выполнения задания верный

Оценка «неудовлетворительно» ставится, если задание не выполнено или если приведен правильный ответ, но решение отсутствует.

## Тема 03.01.05 Базы данных

### Практические работы №16 Использование объединения таблиц

**Цель работы:** получение практических навыков по освоению операций создания запросов с помощью языка SQL.

**Выполнив работу, Вы будете:**

*уметь:*

- создавать запросы с помощью SQL;
- использовать язык SQL с целью объединения таблиц

**Материальное обеспечение:**

Персональный компьютер, MSAccess, методические указания для выполнения практических работ.

**Задание:**

Разработать запросы для базы данных.

**Порядок выполнения работы:**

По таблице EMP, представленная на рисунке 1 создать запросы:

1. Вывести имена, должности сотрудников и названия отделов в которых они работают.
2. Вывести имена и должности сотрудников, а также номера и названия отделов в которых они работают. Результат отсортировать по номерам отделов.
3. Вывести имена и должности сотрудников, а также номера и названия отделов в которых они работают. Результат отсортировать по номерам отделов.
4. Получить список руководителей и подчиненных им сотрудников.
5. Получить список сотрудников компании, номера и названия отделов в которых они работают.
6. Определить к какой категории относится уровень зарплаты каждого сотрудника компании.
7. Выбрать должности, которые есть в отделах 10 и 30. Дубликаты не исключаются.
8. Выбрать должности, которые есть в отделах 10 и 30. Дубликаты исключаются.
9. Выбрать должности сотрудников отдела 10, которых нет в отделе 30.



EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	17.12.1980	800		20
7499	ALLEN	SALESMAN	7698	20.02.1981	1600	300	30
7521	WARD	SALESMAN	7698	22.02.1981	1250	500	30
7566	JONES	MANAGER	7839	02.04.1981	2975		20
7654	MARTIN	SALESMAN	7698	28.08.1981	1250	1400	30
7698	BLAKE	MANAGER	7839	01.05.1981	2850		30
7782	CLARK	MANAGER	7839	09.06.1981	2450		10
7788	SCOTT	ANALYST	7566	09.12.1982	3000		20
7839	KING	PRESIDENT		17.11.1981	5000		10
7844	TURER	SALESMAN	7698	08.09.1981	1500	0	30
7876	ADAMS	CLERK	7788	12.12.1983	1100		20
7900	JAMES	CLERK	7698	03.12.1981	950		30
7902	FORD	ANALYST	7566	03.12.1981	3000		20
7934	MILLER	CLERK	7782	23.12.1982	1300		10

### Форма представления результата:

Предоставить сохраненные запросы в текстовом документе.

### Критерии оценки:

Оценка «отлично» ставится, если задание выполнено верно.

Оценка «хорошо» ставится, если ход выполнения задания верный, но была допущена одна или две ошибки, приведшие к неправильному ответу.

Оценка «удовлетворительно» ставится, если в работе не получен ответ и приведено неполное выполнение задания, но ход выполнения задания верный

Оценка «неудовлетворительно» ставится, если задание не выполнено или если приведен правильный ответ, но решение отсутствует.