

**НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ
АГРАРНЫЙ УНИВЕРСИТЕТ**
Факультет ветеринарной медицины

**Информационные технологии
в ветеринарии**

**Основные понятия и принципы
работы Access 2010**

Методические указания
к лабораторно-практическим занятиям

НОВОСИБИРСК 2015

УДК 619:004 (07)
ББК 48:32.81.27
О-752

Составители: *И.В.Наумкин*, канд. биол. наук, доц.
М.В. Лазарева, ст. препод.

Рецензент: *О.В. Агафонова*, канд. эконом. наук, доцент

Основные понятия и принципы работы Access 2010: методические указания к лабораторно-практическим занятиям для студентов факультета ветеринарной медицины очной и заочной форм обучения/ Новосиб. гос. аграр. ун-т; сост.: И.В. Наумкин, М.В. Лазарева.– Новосибирск: НГАУ, 2015.– 32 с.

Методические указания содержат материал для освоения и приобретения навыков работы с системой управления базами данных MS Access 2010. Рассматриваются основные понятия, возможности MS Access 2010. Прилагаются упражнения по практическому применению рассмотренных функций: способов создания таблиц базы и схемы взаимосвязей между ними. Дается методика разработки и использования форм для ввода и просмотра данных, конструирования отчетов.

Предназначено для студентов очной и заочной формы обучения, обучающихся по программам среднего профессионального и высшего образования в рамках УГС 36.00.00 Ветеринария и зоотехния (Ветеринария, Ветеринарно-санитарная экспертиза).

Утверждено и рекомендовано к изданию учебно-методической комиссией факультета ветеринарной медицины (протокол от 12.10.2015 №5).

Оглавление

Введение.....	4
1. Общая характеристика Microsoft Access 2010	6
1.1. Запуск программы.....	6
1.2. Интерфейс Microsoft Access 2010.....	7
2. Работа с объектами СУБД Access.....	8
2.1. Создание таблиц.....	8
2.1.1. Определение структуры таблицы.....	8
2.1.2. Установка главного ключа.....	10
2.1.3. Сохранение структуры таблицы.....	11
2.1.4. Установление связей между таблицами	11
2.1.5. Ввод записей.....	13
2.2. Создание запросов.....	13
2.3. Создание форм.....	15
2.4. Создание отчета.....	16
3. Задания для самостоятельной работы.....	18
3.1 Лабораторно-практическое задание.....	18
3.1.1. Определить перечень информации, подлежащей регистрации .	18
3.1.2. Разработка структуры базы и ее нормализация (оптимизация) .	18
3.1.3. Определение связей таблицы и справочников.....	22
3.1.4. Создание основной формы для заполнения базы «Прием».....	23
3.1.5. Создание запросов.....	25
3.1.6. Создание отчета.....	29
Библиографический список.....	31

Введение

Использование баз данных и информационных систем становится неотъемлемой составляющей деловой деятельности современного человека. Наша жизнь настолько насыщена различной информацией, что хранить ее без помощи средств вычислительной техники уже практически невозможно. Работа с большими объемами информации без помощи компьютера уже оказывается неприемлемой как с точки зрения затрат на ее хранение, так и с точки зрения управления информацией и скорости доступа к ней.

В связи с этим большую актуальность приобретает освоение принципов построения и эффективного применения систем управления базами данных (СУБД).

База данных (БД) представляет собой совокупность специальным образом организованных данных, хранимых в памяти вычислительной системы и отображающих состояние объектов и их взаимосвязей в рассматриваемой предметной области.

Примерами баз данных могут быть: база данных книжного фонда библиотеки, база данных кадрового состава учреждения, база данных законодательных актов в области уголовного права, база данных территориальных управлений ветеринарии и т.д.

Система управления базами данных (СУБД) – это комплекс языковых и программных средств, предназначенный для создания, ведения и совместного использования БД многими пользователями.

Классификация БД:

– по содержанию:

а) **фактографическая** – содержит краткую информацию об объектах некоторой системы в строго фиксированном формате. В фактографических БД сведения об описываемых объектах представляются в строго определенном формате (в виде числовых или текстовых характеристик объектов).

б) **документальная** – содержит документы самого разного типа: текстовые, графические, звуковые, мультимедийные. В документальных БД единицей хранения является документ, и пользователю выдается ссылка на документ или сам документ.

– по организации и технологии обработки данных:

а) **локальная (персональная)** – это база данных, предназначенная для локального использования одним пользователем;

б) **централизованная** – база данных, хранящихся на одном компьютере.

в) **распределённая** – база данных, разные части которой хранятся на различных компьютерах, объединённых в сеть;

– по используемым моделям данных:

а) **иерархическая** модель БД – изображают в виде дерева. Элементы дерева – вершины представляют совокупность данных, например логические записи.

б) **сетевая** модель БД – поддерживает структуру данных в виде сети. Сетевая модель данных позволяет отображать разнообразные взаимосвязи элементов данных в виде произвольного графа. Сетевая БД состоит из набора записей и набора соответствующих связей. На формирование связи особых ограничений не накладывается.

в) **реляционная** модель БД – база данных с табличной организацией данных.

Многие компьютерные БД являются Реляционными (от англ. Relation – отношение).

Реляционная модель – это модель данных, в которой все данные, доступные пользователю, организованы в виде двумерных таблиц, а все операции над данными сводятся к операциям над этими таблицами.

Microsoft Access 2010 является реляционной СУБД. Microsoft Access – это одно из приложений, входящих в состав пакета Microsoft Office. Приложение является мощной программой обработки баз данных, которую можно использовать для хранения, сортировки и управления практически любыми типами данных.

Описание типов файлов СУБД MS Access

.accdb – расширение файлов стандартной базы данных СУБД MS Access начиная с MS офиса версий 2007/2010. Пришло на смену расширению *.mdb* в старых версиях MS офиса до 2007.

.accdw – расширение файл базы данных Access (.ACCDB), который хранятся удаленно и содержит связующую информацию для загрузки их с сервера Microsoft SharePoint. Дает возможность пользователям совместно работать с файлами баз данных.

.accde – расширение файлов базы данных СУБД MS Access версий 2007/2010, работающих в режиме «исполнения». Изменение структуры таких баз невозможно, т.к. из них удален весь исходный код, поэтому идет исполнение кода языка VBA. Данное расширение пришло на смену аналогичному типу файлов *.mde* из старых версий.

.accdt – расширение файлов, содержащих шаблоны баз данных СУБД MS Access версий 2007/2010;

.accdr – расширение закрытого файла базы данных, созданного программой связанных баз данных Microsoft Access, в котором содержится база данных (.ACCDB). Дает возможность открывать и запускать базу данных, без возможности внесения в неё изменений, в режиме, доступном только для чтения.

1. Общая характеристика Microsoft Access 2010

1.1. Запуск программы

Запуск MS Access можно осуществить следующими способами:

- 1) меню **Пуск** системы Windows → **Все программы** → **Microsoft Office** → **Microsoft Access 2010**;
- 2) запустить файл с расширением *.accdb, *.accdw, *.accde, *.accdt, *.accdr, *.mdw.

После запуска MS Access на экране появляется диалоговое окно с наименованием Microsoft Access в строке заголовка (рис. 1).

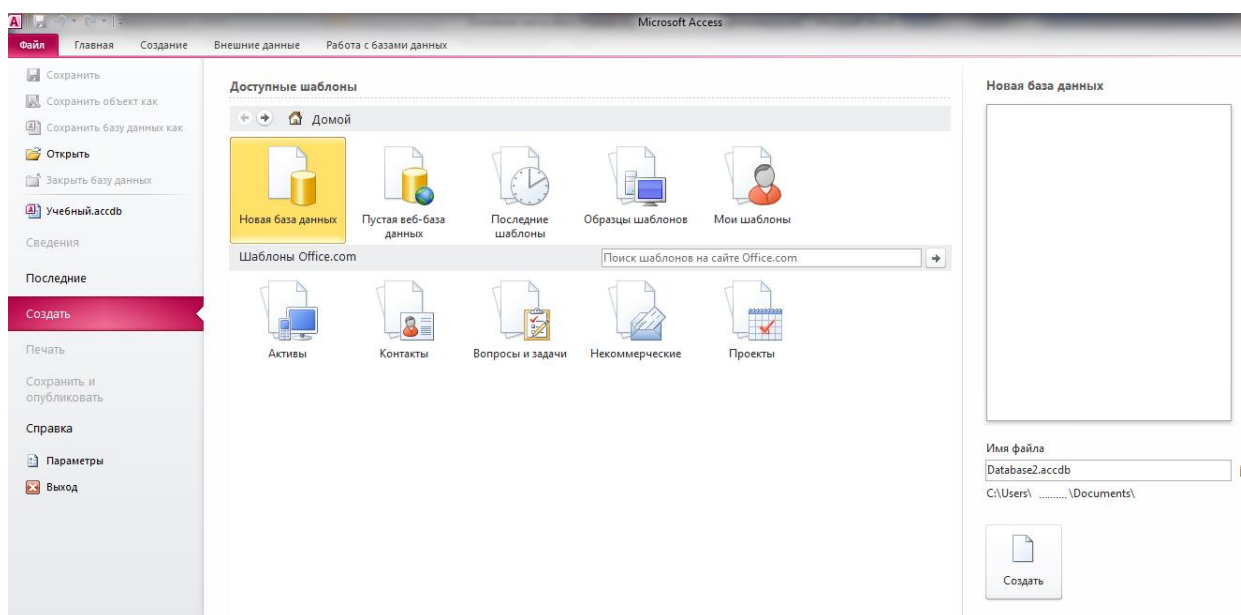


Рис. 1. Создание базы данных в MS Access 2010

В этом окне следует выбрать одно из предлагаемых действий:

- 1) открыть существующую базу данных;
- 2) создать новую (пустую);
- 3) создать базу данных с помощью прилагаемых **Шаблонов**.

Для создания пустой базы данных выберите в диалоговом окне **Доступные шаблоны** выберите значение **Новая база данных**. В открывшемся окне диалога **Новая база данных** выполните:

1. В строке **Имя файла** задайте имя новой базы данных.
2. В конце строки **Имя файла** нажмите знак Папка и задайте папку, в которой предполагается сохранить создаваемую базу данных.
3. Нажмите кнопку **Создать**.

1.2. Интерфейс Microsoft Access 2010

Главный элемент пользовательского интерфейса MS Access 2010 представляет собой *Ленту*, которая идет вдоль верхней части окна каждого приложения (рис. 2).

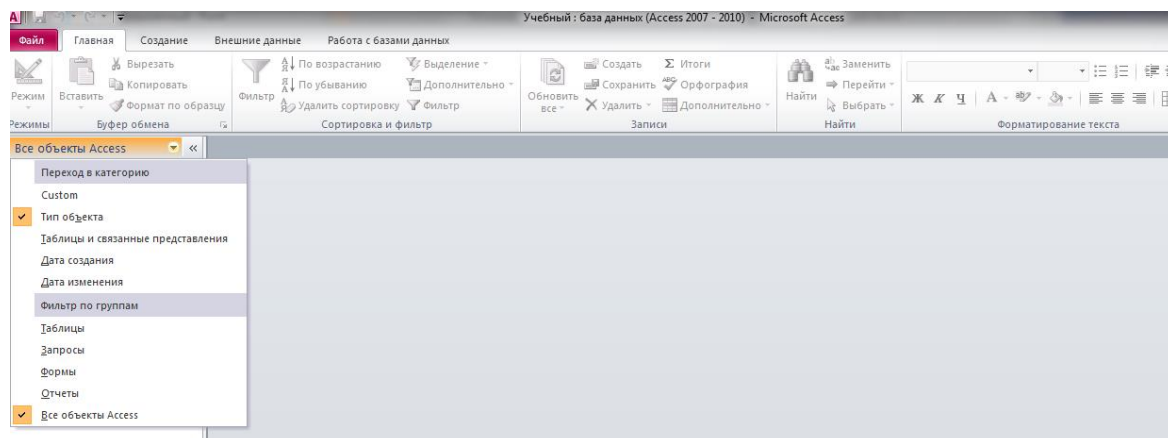


Рис. 2. Интерфейс MS Access 2010

Лента управления содержит вкладки. По умолчанию их пять: *Файл*, *Главная*, *Создание*, *Внешние данные*, *Работа с базами данных*. Каждая вкладка связана с видом выполняемого действия.

Панель быстрого доступа, находящаяся по умолчанию над лентой, обеспечивает доступ к командам одним щелчком мыши. Набор по умолчанию включает команды *Сохранение*, *Отмена* и *Возврат*, при этом можно настроить панель быстрого доступа для добавления в нее наиболее часто используемых команд.

Область навигации, расположенная по левому краю окна Access. Она предназначена для отображения объектов или групп объектов открытой базы данных, а также для перехода от объекта к объекту. Чтобы раскрыть группу объектов следует щелкнуть мышкой по кнопке. Управлять объектами можно командами ленты и командами контекстного меню.

СУБД Access работает с несколькими видами объектов:

Таблицы – основная единица хранения данных в базе. Реляционная база данных может иметь много взаимосвязанных таблиц. Основные операции над таблицами: просмотр и обновление, сортировка, фильтрация и печать.

Запросы – представляют собой формализованное требование на отбор данных из таблиц. С помощью запросов данные упорядочивают, фильтруют, отбирают, изменяют, объединяют, то есть обрабатывают. Существует много различных видов запросов, и самые простые из них и, к тому же наиболее часто используемые – это запросы на выборку. Цель запроса на выборку

состоит в создании результирующей таблицы, в которой отображаются только нужные по условию запроса данные из базовых таблиц.

Формы – это объекты, с помощью которых в базу вводят новые данные или просматривают имеющиеся.

Отчеты – это объекты, с помощью которых данные выдают на принтер в удобном и наглядном виде.

Макросы – это макрокоманды. Если какие-то операции с базой производятся особенно часто, то имеет смысл сгруппировать несколько команд в один макрос и назначить его выделенной комбинации клавиш.

Модули – это программные процедуры, написанные на каком-либо языке программирования.

Область документов, в которой отображается каждый объект базы данных, открываемый в любом режиме.

Строка состояния, расположенная вдоль нижней границы окна Access, отображающая кнопки переключения в различные режимы работы с активным объектом.

Также следует знать следующие понятия:

Сущность – объект любого происхождения, хранящийся в базе данных.

Атрибут – свойство, характеризующее сущность. В базе данных представляет собой название графы (заголовки столбца).

2. Работа с объектами СУБД Access

2.1. Создание таблиц

2.1.1. Определение структуры таблицы

В реляционных БД *строка* таблицы называется *записью*, а *столбец* – *полем*.

Поля – это различные характеристики (иногда говорят – атрибуты) объекта. Значения полей в одной строчке относятся к одному объекту. Каждое поле таблицы имеет имя. Например, в таблице «Животное» именами полей могут быть: Вид, Пол, Порода, Кличка, Возраст, Масть животного и др.

В отличие от электронной таблицы *Excel*, в которой в каждой новой строке одного столбца могут храниться значения разных типов (числовые, текстовые, формулы, даты, логические выражения и т.д.), в базе данных – в каждой новой строке одного столбца могут быть сохранены данные только одного типа, который определяется при создании поля.

Одна *запись* содержит информацию об одном объекте.

Например, одна запись о каком либо объекте – это информация об одном животном, о работе одного врача, успеваемости конкретного студента и т.д.

Для создания полей в таблице в окне базы данных необходимо выбрать вкладку **Создание** и нажать на кнопку **Конструктор таблиц**.

В этом режиме открывается диалоговое окно, в котором таблица содержит три столбца: *Имя поля*, *Тип данных*, *Описание* (рис. 3).

В столбец *Имя поля* введите в нужной последовательности имена полей (например: Код вида животного, порода, кличка, возраст животного, код владельца и т.д.).

В столбце *Тип данных*, напротив каждого имени поля, необходимо выбрать нужный тип данных, используя кнопку выпадающего списка.

Тип определяет множество значений, которые может принимать данное поле в различных записях.

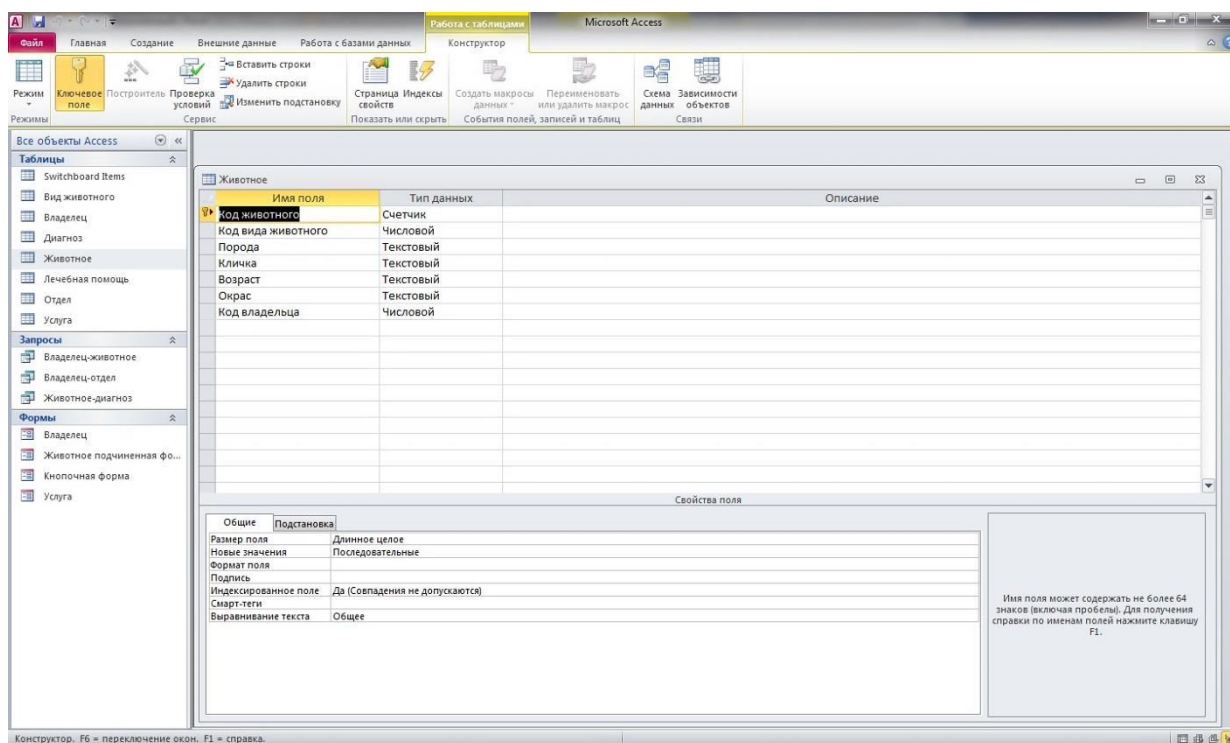


Рис. 3. Таблица в режиме конструктора

В реляционных базах данных используются следующие основные типы полей:

- *текстовый* (содержит текстовую строку, максимальная длина задаётся параметром «размер», но не может быть больше 255);
- *поле МЕМО* (текст длиной до 65535 символов);
- *числовой* (имеет поля, значения которых могут быть только числами);
- *дата/время* (имеет поля, содержащие календарные даты в форме «день/месяц/год» или в формате времени «час:мин:сек»);

- *денежный* (специальный формат для финансовых нужд, по сути являющийся числовым);

- *счётчик* (при добавлении новой записи внутренний счётчик таблицы увеличивается на единицу и записывается в данное поле новой записи, таким образом, значения этого поля гарантированно различны для разных записей), тип предназначен для ключевого поля;

- *логический* (может принимать всего два значения: да или нет, правда или ложь, включен или выключен);

- *объект OLE* (в этом поле могут храниться документы, картинки, звуки и т.п.);

- *гиперссылка* (используется для хранения ссылок на ресурсы Интернета);

- *вычисляемый* (позволяет создавать поля, значения которых будут вычисляться на основе данных, хранящихся в полях той же таблицы, а также с помощью встроенных функций Access.). В выражениях для вычисления **нельзя** использовать поля из других таблиц или запросов. Результаты вычислений доступны только для чтения.

Описание поля не является обязательным параметром таблицы. В него заносят информацию характеризующую хранящиеся в поле данных. Позволяет разработчику или пользователю быстро разобраться в структуре данных, хранящихся в таблице.

2.1.2. Установка главного ключа

Главным ключом в базах данных называют поле (или совокупность полей), значение которого не повторяется у разных записей. Например, ключевым полем может являться инвентарный номер животного.

Ключ предназначен для:

- для идентификации записей в таблице;
- установления связей между таблицами;
- создания ограничений ссылочной целостности;

Например, для базы данных клиентов ветеринарной клиники, в качестве ключевых полей могут быть назначены поля: *Фамилия, Имя, Отчество, Дата рождения*. В редких случаях все значения этих полей могут совпасть. В этом случае к ключевым полям можно добавить *Место рождения*.

Для определения главного ключа таблицы, выделите поле *Код*, щелкнув правой кнопкой мыши на области маркировки (слева от имени поля), в контекстном меню выберите пункт **Ключевое поле**, или соответствующую кнопку во вкладке **Конструктор**. Признаком установки ключа является изображение ключа слева от имени поля.

Следует помнить, что:

В реляционной базе данных ключ реализуется в виде *индекса*.

Индекс – указатель на данные, размещенные в таблице и представляет собой информацию о точном их физическом расположении.

Одной из особенностей базы данных является предоставление механизма быстрого поиска информации по заданным параметрам. Исполнение этой особенности СУБД обеспечивают *индексы*, которые можно сравнить с алфавитным указателем в справочнике лекарственных средств.

Индекс содержит отсортированные значения указанного поля и ссылки на номера записей в таблице, что позволяет быстро находить искомые значения.

2.1.3. Сохранение структуры таблицы

После определения перечня всех полей в текущей таблице нажмите кнопку **Сохранить**, которая располагается над вкладкой **Файл**. В появившемся диалоговом окне «**Сохранение**» введите имя таблицы, нажмите кнопку ОК. Таблица с соответствующим именем появится в списке объектов **Таблицы** в окне **Все объекты Access**.

2.1.4. Установление связей между таблицами

Создавая базу данных, в которой будет храниться информация о животных мы должны создать не менее трех таблиц: вид животного, сведения о животном и сведения о владельце. Следовательно таблицы должны быть связаны между собой. Установка связей позволяет использовать *поля-подстановки (списки)*.

Существует три основные типы связей: «*один-к одному*», «*один-ко многим*», «*многие ко многим*».

Рассмотрим особенности связей между таблицами:

- связи «*один к одному*» (1 : 1) (животное – паспорт);

Отношение один-к-одному подразумевает, что одной записи в главной таблице соответствует одна запись в подчиненной таблице. Так студенты факультета ветеринарной медицины могут быть включены в состав только одной группы на данном факультете.

- связи «*один ко многим*» (1 : М) (владелец – животное);

Одна запись из одной таблицы может иметь связь с множеством записей из другой таблицы. У одного владельца может быть несколько животных.

- связи «*многие ко многим*» (М : М) (врач - животное);

Отношение многие-ко-многим подразумевает, что одна запись из главной таблицы может иметь связь со многими записями из подчиненной таблицы. А одна запись из подчиненной таблицы может иметь связь со многими записями главной таблицы. Это значит, что одно животное может попасть на лечение к нескольким специалистам (врачу-эпизоотологу, врачу акушеру-гинекологу, терапевту и т.д.). А каждый врач проводит обследование и лечение многих животных.

Для установки связей между таблицами необходимо выполнить следующие действия:

1. Выберите вкладку **Работа с базами данных**.
2. В группе Отношения выберите **Схема данных**.
3. В появившемся диалоговом окне Добавление таблицы выберите таблицы, которые должны быть связаны. Названия каждой из таблиц со списками полей появятся в окне Схема данных (рис. 4).

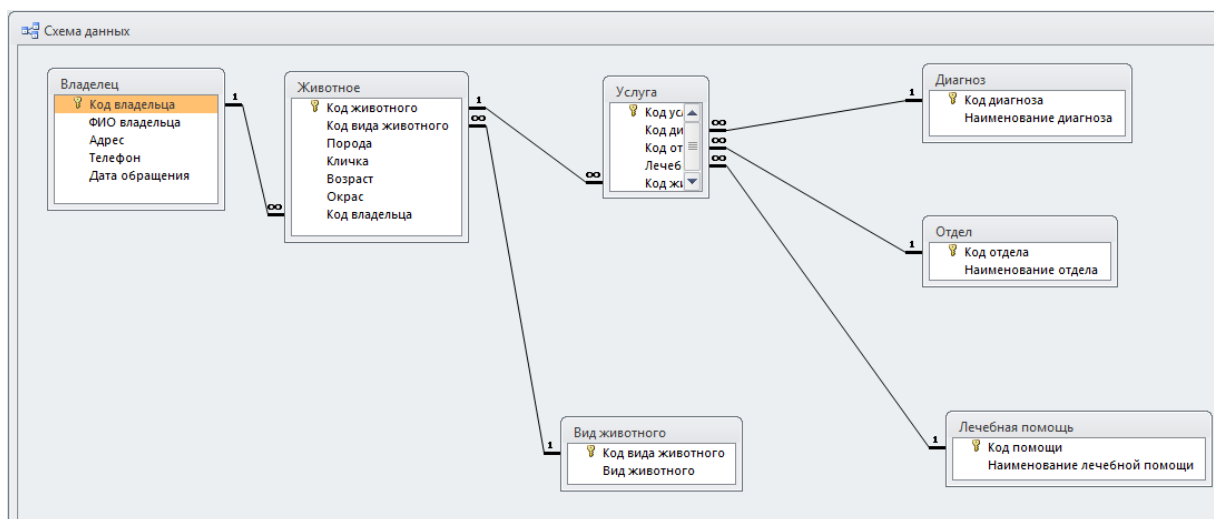


Рис. 4. Схема данных

4. Установите курсор в любую из таблиц на поле, по которому будет установлена связь и "перетащите" это поле на связующее поле другой таблицы. Тип данных, значения и свойства связываемых полей должны совпадать.

5. Активизируйте флажок *Обеспечение целостности данных*.

От полей, указанных при определении связи зависит тип создаваемой связи, который отображается в этом же окне (рис. 5).

Для удаления связи в окне **Схема данных** выделите ненужную связь и нажмите клавишу Delete.

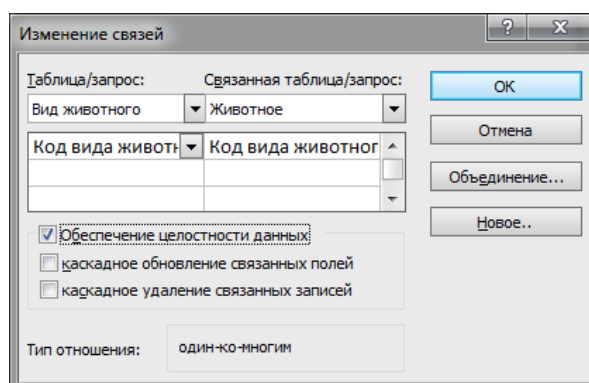


Рис. 5. Изменение связей

2.1.5. Ввод записей

Ввод записей может выполняться в режиме работы с таблицами. Переход к табличному представлению БД осуществляется с помощью вкладки **Главная** → группа **Режимы** → кнопка **Режим таблицы** (рис. 6).

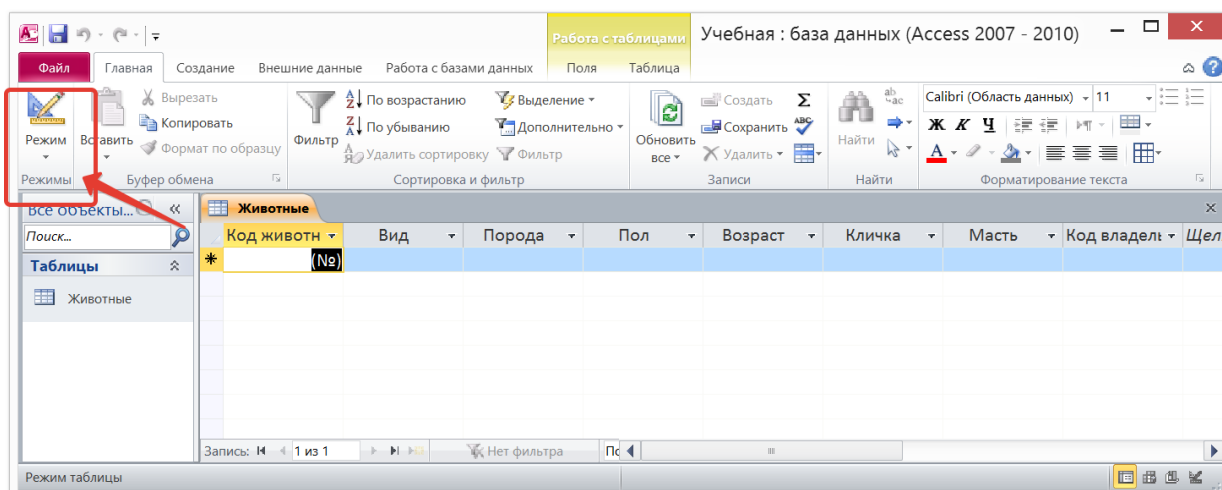


Рис. 6. Ввод данных в режиме таблицы

После заполнения всех полей первой записи, новая запись добавляется после нажатия на клавишу «вниз».

2.2. Создание запросов

Запросы предназначены для просмотра, изменения и анализа данных. Они используются также в качестве источника записей при создании форм и отчётов. Одним из наиболее распространённых запросов является запрос на выборку, который выполняет отбор данных из одной или нескольких таблиц в соответствии с заданными пользователем критериями.

Для создания нового запроса:

1. На вкладке **Создание** в группе **Запросы** выберите кнопку **Конструктор запросов** (рис. 7).

2. В диалоговом окне *Добавление таблицы* укажите имена таблиц, по полям которых будет производиться запрос, нажимая кнопку *Добавить* после каждого указанного имени таблицы.

3. Нажмите кнопку *Заккрыть*.

4. В строке *Поле* щелкните в правой части клетки на стрелке, указывающей вниз и выберите имя поля, по которому будет осуществляться запрос. Если запрос осуществляется по полям из разных таблиц, то сначала щелкните в строке *Таблица* и укажите нужную таблицу, что позволит ограничить список полей в строке *Поле*. Если запрос будет осуществляться по нескольким полям, отобразите их имена в свободных клетках строки *Поле*.

5. Проследите, чтобы в строке *Вывод на экран* флажок отображался бы галочкой.

6. Если это нужно, сохраните запрос для последующего использования. Для выполнения запроса нажмите кнопку с восклицательным знаком **Выполнить** группа **Результаты**.

Поле:	Имя таблицы:	Сортировка:	Вывод на экран:	Условие отбора:
ФИО владельца	Владелец		<input checked="" type="checkbox"/>	
Код вида животного	Животное		<input checked="" type="checkbox"/>	
Порода	Животное		<input checked="" type="checkbox"/>	
Кличка	Животное		<input checked="" type="checkbox"/>	
Возраст	Животное		<input checked="" type="checkbox"/>	
Окрас	Животное		<input checked="" type="checkbox"/>	
			<input type="checkbox"/>	
			<input type="checkbox"/>	

Рис. 7. Конструктор запросов

2.3. Создание форм

Формы Access позволяют создавать пользовательский интерфейс для таблиц базы данных. Хотя для выполнения тех же самых функций можно использовать режим таблицы, формы предоставляют преимущества для представления данных в упорядоченном и привлекательном виде.

Форма представляет собой некий электронный бланк, в котором имеются поля для ввода данных (рис. 8). В Форме каждое поле можно разместить в точно заданном месте, выбрать для него цвет и заливку.

Форма строится на основе *таблицы* или *запроса*. Формы могут быть выведены на экран в трех видах: режим формы, режим макета и режим конструктора. Для перехода из одного режима в другой используются команды группы **Режимы**.

Создайте форму с помощью **Мастера форм**.

1. Выберите команду **Мастер форм** (вкладка Создание группа Формы).
3. В появившемся диалоговом окне укажите имена полей для главной и подчиненной форм и порядок их размещения в новой форме, выбрав имя таблицы из раскрывающегося списка Таблицы/Запросы. Нажмите кнопку *далее*.
4. В следующем окне выберите переключатель **Подчиненные формы**.
5. Далее выберите вид подчиненной формы.
6. Озаглавьте главную и подчиненную формы и нажмите кнопку *Готово*.

Код животного	Животное	Порода	Кличка	Возраст	Окрас	Владелец
3	Собака	мопс	Рекс	2 года	белый	1
5	Корова	мясо-молочная	Зоря	3 года	черно-пестрая	1
*	(№)					1

Рис. 8. Форма MS Access

2.4. Создание отчета

Отчет – это форматированное представление данных, которое выводится на экран, в печать или файл. Они позволяют извлечь из базы нужные сведения и представить их в виде, удобном для восприятия, а также предоставляют широкие возможности для обобщения и анализа данных.

Располагая базой данных можно распечатать любую таблицу, запрос или форму. Однако результаты печати не будут выглядеть профессионально, так как эти инструменты не предназначены для печати. С помощью отчета можно получить результаты в высококачественном варианте. В Access 2010 отчет представляет собой форму специального типа, предназначенную для вывода на печать. Но в отличие от форм отчеты не предназначены для вывода в окне и предназначены только для печати, т.е. создают не экранные, а печатные документы.

В Microsoft Access можно создавать отчеты различными способами:

- отчет
- конструктор отчетов
- мастер отчетов
- пустой отчет
- наклейки

Для создания отчета с помощью **Мастера Отчетов**:

1. На вкладке **Создание** группа **Отчеты** выберите команду **Мастер Отчетов** (рис. 9).
2. Укажите имя таблицы или запроса, на основе которых создаете отчет.
3. Выберите поля, данные которых будут помещены в отчет.
4. Задайте требуемый порядок сортировки полей.
5. Выберите вид макета отчета.
6. Задайте имя отчета.
7. Нажмите кнопку **Готово**.

Отчет открывается в режиме **Предварительного просмотра**, который позволяет увидеть, как будет выглядеть отчет в распечатанном виде.

Редактирование и форматирование отчета можно выполнить в режиме **Конструктора**.

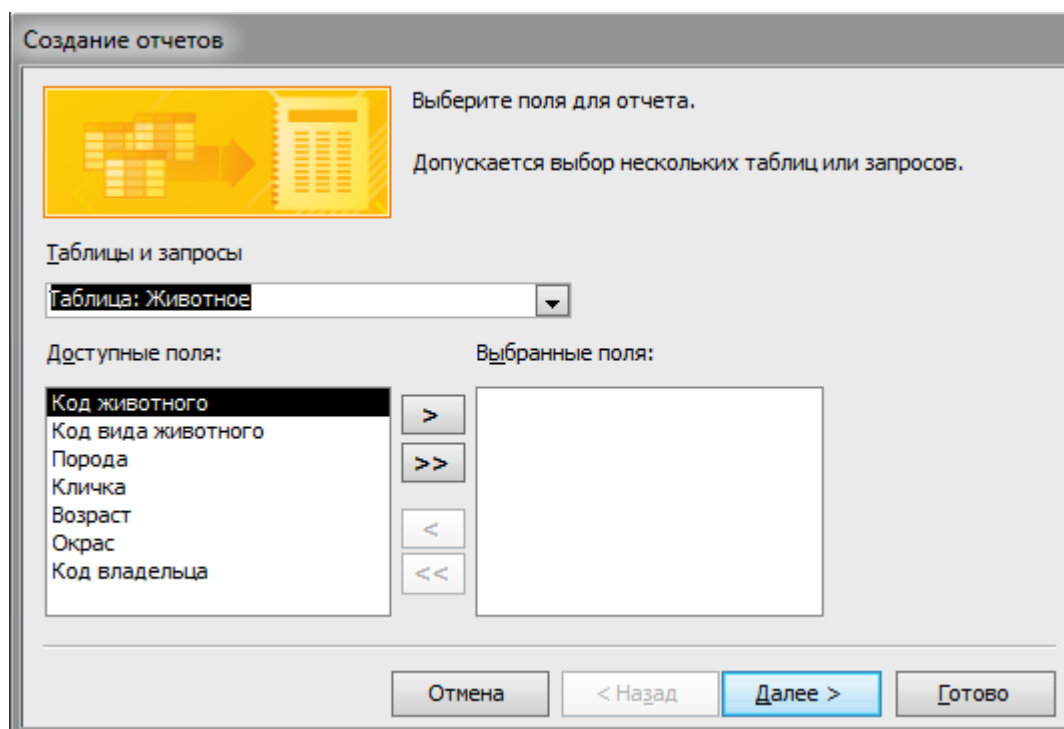


Рис. 9. Создание отчетов с помощью Мастера Отчетов

Для создания отчета самостоятельно без помощи Мастера Отчетов:

1. На вкладке **Создание** группа **Отчеты** выберите команду **Конструктор** отчетов.
2. На вкладке **Конструктор** группа **Сервис** выберите команду **Добавить** поля.
3. Укажите имя таблицы, на которой должен базироваться отчет и выберите поля, данные которых будут отображаться в отчете.

Заголовок отчета — печатается только в начале отчета, используется на титульной странице.

Верхний колонтитул — печатается вверху каждой страницы.

Заголовок группы — печатается перед обработкой первой записи группы.

Область данных — печатается каждая запись таблицы или динамического набора данных запроса.

Примечание группы — печатается после обработки последней записи группы.

Нижний колонтитул — печатается внизу каждой страницы.

Примечание отчета — печатается в конце отчета после обработки всех записей.

3. Задания для самостоятельной работы

3.1 Лабораторно-практическое задание

Тема: *Работа с системой управления базами данных Access 2010.*
Создание таблиц.

Цель занятия: Ознакомиться с формированием информационных баз данных и основными командами СУБД Access.

Содержание занятия:

1. Изучить теоретическую часть построения базы данных.
2. Ознакомиться с примером базы данных MS Access.
3. Научиться создавать таблицы, формы, запросы и отчеты в MS Access.

Описание: Создать упрощенную базу регистрации приема больных животных в ветеринарной клинике, позволяющую формировать отчеты о заболеваемости отдельных видов животных по районам г. Новосибирска, видам животных, выполненной работе ветеринарными работниками клиники, о загруженности отдельных должностей работников в приеме животных.

Порядок выполнения задания:

3.1.1. Определить перечень информации, подлежащей регистрации

1. Номер первичного учета.
2. Дата приема.
3. Сведения о владельце животного.
4. Сведения о месте проживания владельца животного.
5. Информация о животном.
6. Сведения о враче, осуществляющим лечебные мероприятия.
7. Сведения о диагнозе.

3.1.2. Разработка структуры базы и ее нормализация (оптимизация)

Необходимо сформировать таблицы базы данных, позволяющие исключить дублирование информации.

На этом этапе, исходя из поставленных задач, следует определить, какая информация потребуется для составления отчетов и будет дублироваться.

2.1. В базе необходимо сохранить контактную информацию о владельце животного и месте проживания. При этом в задаче стоит формировать отчеты о распространенности заболеваний животных по районам города. Таким образом, из адреса проживания владельца животного следует отдельно выделить район. Процесс выделения дублирующейся информации и формирование новых (справочных) таблиц называется **нормализацией базы**.

Таблицу со сведениями о районах города мы будем называть **СПРАВОЧНИКОМ**. В названии таблицы, относящейся к справочникам будем добавлять префикс «сп». Справочник районов получит название «спРайон».

Создадим новую таблицу и перейдем в режим «Конструктора». На запрос о присвоении названия таблицы введем имя «спРайон».

Создадим поля таблицы.

Имя поля	Тип данных	Описание	Размер поля
Район	Текстовый	Код района	3
Название	Текстовый	Наименование района	15

Сохраним структуру таблицы и перейдем к ее заполнению по команде «Открыть».

Справочник будет содержать следующую информацию:

Район	Название
001	Дзержинский
002	Железнодорожный
003	Заельцовский
004	Калинский
005	Кировский
006	Ленинский
007	Октябрьский
008	Первомайский
009	Советский
010	Центральный
011	Новосибирский
012	Другие

2.2. Поставленной задачей определена необходимость формирования различных отчетов по видам животных, следовательно в амбулаторном журнале должна фиксироваться информация о виде животного. Для исключения ошибок и ее дублирования следует создать справочник видов животных, который мы назовем «спВидЖивотного».

Создадим новую таблицу и перейдем в режим «Конструктора». На запрос о присвоении названия таблицы введем имя «спВидЖивотного».

Создадим поля таблицы.

Имя поля	Тип данных	Описание	Размер поля
ВидЖивотного	Текстовый	Код вида животного	3
Название	Текстовый	Наименование вида животного	25

Сохраним структуру таблицы и перейдем к ее заполнению по команде «Открыть».

Справочник будет содержать следующую информацию:

ВидЖивотного	Название
001	Гуси
002	Кошки
003	Кролики

Вид Животного	Название
004	Крупный рогатый скот
005	Крысы
006	Куры
007	Лисы
008	Лошади
009	Мелкий рогатый скот
010	Морские свинки
011	Мыши
012	Норки
013	Песцы
014	Попугаи
015	Рыбки
016	Собаки
017	Утки
018	Шиншиллы
019	Другие

2.3. Исходя из поставленных задач потребуется формировать отчеты о работе ветеринарных врачей клиники. Следовательно эта информация должна быть занесена в каждой записи. Поэтому сформируем еще один справочник с названием «спВрач».

Создадим новую таблицу и перейдем в режим «Конструктора». На запрос о присвоении названия таблицы введем имя «спВрач».

Создадим поля таблицы.

Имя поля	Тип данных	Описание	Размер поля или формат
Врач	Текстовый	Код врача	3
Вр_ФИО	Текстовый	ФИО врача	50
Вр_Должн	Текстовый	Должность врача	50
Вр_Дата_Пр	Дата/время	Дата приема врача на работу	Краткий формат даты

Сохраним структуру таблицы и перейдем к ее заполнению по команде «Открыть».

Справочник будет содержать следующую информацию:

Врач	Вр_ФИО	Вр_Должн	Вр_Дата_Пр
001	Глущенко Василий Вячеславович	Главный ветврач	01.01.2001
002	Глущенко Екатерина Евгеньевна	Хирург	01.01.2001
003	Лазарева Марина Викторовна	Терапевт	01.01.2012
004	Грязин Валерий Николаевич	Дерматолог	01.01.1988
005			
006			
007			

2.4. Сформируем структуру основной таблицы амбулаторного журнала.

Создадим новую таблицу и перейдем в режим «Конструктора». На запрос о присвоении названия таблицы введем имя «Прием».

Следует помнить рекомендацию, что названия полей, значения которых будут браться из справочников следует давать по аналогии с названием поля в самом справочнике.

Создадим поля таблицы.

Имя поля	Тип данных	Описание	Размер поля или формат
Код	Счетчик	Код первичного учета	Длинное целое
Дата	Дата/время	Дата приема животного	Краткий формат даты
Владелец	Текстовый	ФИО владельца животного	50
Район	Текстовый	Код района, из которого поступило животное	3
Вл_Адрес	Текстовый	Место жительства владельца	100
Вл_телефон	Текстовый	Номер телефона владельца (мобильного)	12 (###) ###-##-## ¹
ВидЖивотного	Текстовый	Код вида животного	3
Кличка	Текстовый	Кличка животного	15
Возраст	Числовой	Возраст животного, полных лет	Длинное целое
Врач	Текстовый	Код врача, осуществляющего прием	3
Диагноз	Текстовый	Название диагноза	50

Примечание: 1 – формат номера телефона вносится в строку «Маска ввода»

Поля «Район», «ВидЖивотного», «Врач» должны содержать значения из соответствующих справочников. Получение фиксированных значений из справочника – называется подстановкой. Поэтому, для данных полей, необходимо настроить правила подстановки.

Делаем активным поле «Район» и открываем закладку «Подстановка». Содержание этой вкладки приведено на рис. 10.

Общие	Подстановка
Тип элемента управления	Поле со списком
Тип источника строк	Таблица или запрос
Источник строк	спРайон
Присоединенный столбец	1
Число столбцов	2
Заглавия столбцов	Нет
Ширина столбцов	
Число строк списка	16
Ширина списка	Авто
Ограничиться списком	Нет
Разрешение нескольких знач	Нет
Разрешить изменение списка	Да
Форма изменения элементов	фПрием
Только значения источника с	Да

Рис. 10. Настройка полей вкладки «Подстановка»

3.1.3. Определение связей таблицы и справочников

Вспомним, что различают следующие типы связей: «Один – к одному» (в одной группе может быть только один староста), «Один – ко многим» (на одном факультете обучается много студентов), «Многие – к одному» (много студентов зачислены на одну специальность или направление подготовки), «Многие – ко многим» (многие студенты могут обучаться по многим специальностям, но в СУБД MS Access, как и во многих других реляционных СУБД, данный вид связи бесполезен, поэтому не поддерживается). Чаще всего используется тип связи «Один – ко многим». Например, в нашей базе это значит, что одной записи из справочника «спРаон» может соответствовать много записей в таблице «Прием».

Чтобы установить связи между таблицами надо перейти в режим «Схемы данных» (на ленте «Работа с базами данных»).

Добавляем все созданные нами базы данных на рабочий стол. Таблицу «Прием» располагаем в центре.

Для всех справочников установим с таблицей «Прием» связь «Один – ко многим», через соответствующие поля. В окне связей устанавливаем параметры, как указано на рис. 11.

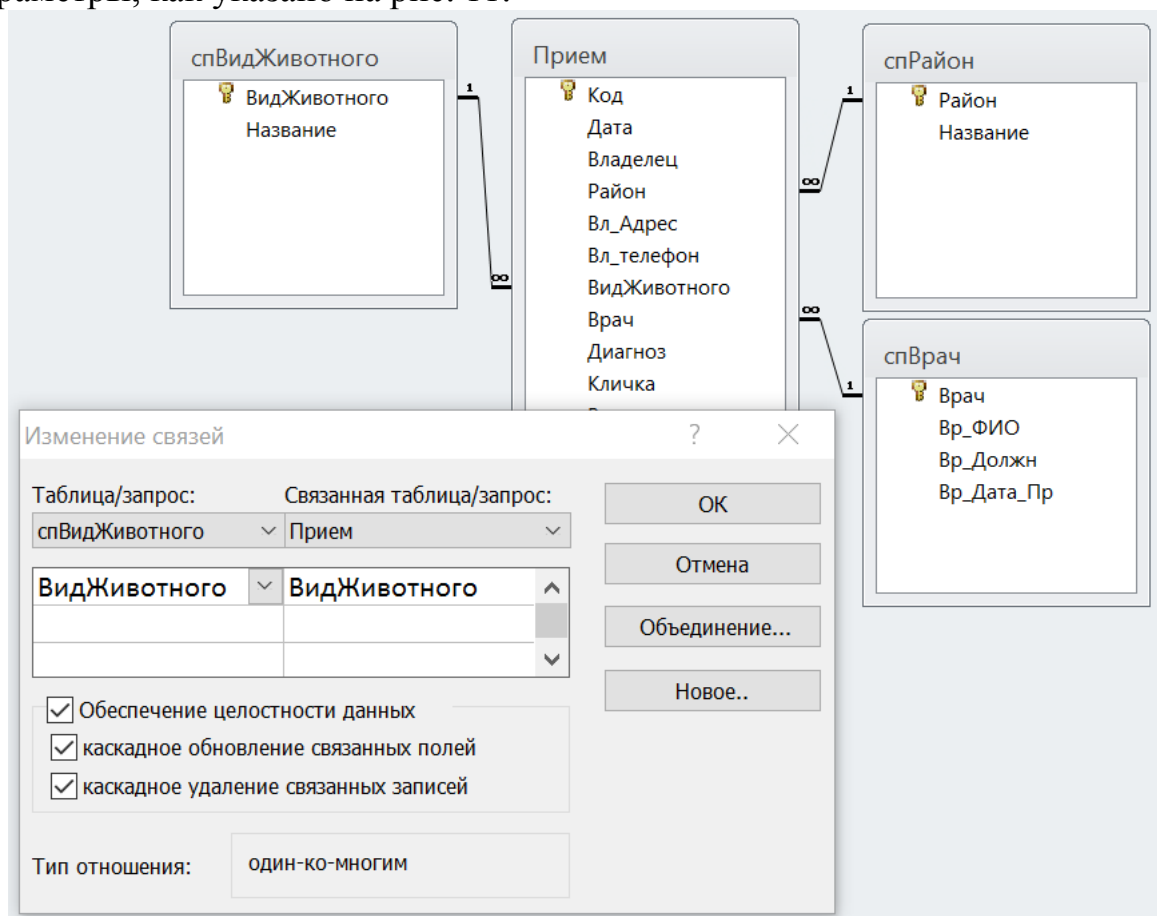


Рис. 11. Установка параметров связей между таблицами

3.1.4. Создание основной формы для заполнения базы «Прием»

Для создания формы ввода переходим на ленту «Создание» и выбираем команду «Конструктор форм».

На экране появляется пуста форма.

Из выпадающего меню (по нажатию правой кнопки мышки) выбираем команду «Свойства формы». В закладке данные в качестве «Источника записей» выбираем таблицу «Прием».

С правой части отображен перечень таблиц и доступных полей. Если на экране нет окна с перечнем таблиц и доступных полей, то на ленте «Конструктор» выбираем команду «Добавить поля».

1. Справа в таблице «Прием» находим поле дата и с помощью мышки переносим ее на форму. На экране отображаются два элемента. «Подпись» - это текстовый комментарий к полю ввода и «Поле» для ввода значений.

2. Следующим шагом на форме создаем блок для поля «Владелец».

3. Формируем поле «Вл_телефон», в строке подпись меняем комментарий на «Номер телефона» или «Номер телефона владельца».

4. Создание поля для работы со справочником имеет свои особенности.

а) на ленте «Конструктор» находим команду добавления поля со списком (рис. 12);

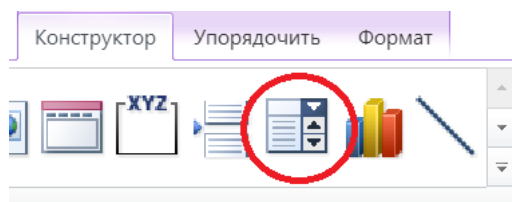


Рис. 12. Выбор команды добавления поля со списком

б) с помощью мышки указываем на форме место расположения поля со списком и его размер;

в) выбираем каким способом поле со списком будет получать эти значения: «Будет введен фиксированный набор значений»; «Объект «поле со списком» получит значения из другой таблицы или другого запроса». Выбираем режим «Объект «поле со списком» получит значения из другой таблицы или другого запроса» и нажимаем кнопку «Далее»;

г) выбираем режим «показать Таблицы» и выбираем из списка таблицу «спРайон»;

д) выбираем поле с именем «Название», которое станет столбцом в «поле со списком» и нажимаем кнопку «Далее»;

е) выбираем поле, по которому будет производится сортировка списка и режим сортировки. В нашем случае список сортируется по коду района – поле «Район», режим сортировки – «по возрастанию»;

ж) по умолчанию включен режим «Скрыть ключевой столбец (рекомендуется)»;

з) указать где сохраняется результат выбора из поля со списком: «Запомнить значение», «Сохранить в поле». Выбираем режим «Сохранить в поле», в качестве поля для сохранения выбираем поле «Район» в таблице «Прием», нажимаем кнопку «Далее»;

и) задаем подпись, которую содержит поле со списком – «Район проживания». Жмем кнопку «Готово».

Внимание! Если программа Access установлена не полностью и автоматически окно настройки подстановки не открывается, то выполняем следующие действия:

а) выполняем действия описанные выше в пунктах 4а и 4б;

б) открываем окно свойств поля со списком;

в) на закладке «Данные» в строке «Источник строк» выбираем справочник «спРайон», если его нет в перечне доступных таблиц, то нажимаем в этой строке на кнопку «...». В окне построителя запросов выбираем таблицу «спРайон» и оба поля «Район» и «Название» переносим в нижнее окно в отдельные столбцы. Закрываем окно построителя запросов;

г) в строке «Тип источника строк» выбираем «Таблица или запрос»;

д) строке «Присоединенный столбец» ставим число 1;

е) на закладке «Все» в строке «Число столбцов» ставим цифру 2;

ж) в строке «Ширина столбцов» вносим следующий текст: «0см; 10 см», где «0см;» говорит о том, что содержимое первого столбца не будет отображаться на экране.

Добавляем остальные поля таблицы «Прием». Не забываем про особенности создания полей со списками для заполнения значений из справочников «спВидЖивотного» и «спВрач».

После добавления всех полей форма будет иметь вид, указанный на рис. 13.

Дата	Дата																		
Владелец	Владелец							Номер телефона	Вл_телефон										
Район	Район																		▼
Адрес:	Вл_Адрес																		
Вид животного	ВидЖивотного																		▼
Кличка	Кличка							Возраст	Возраст										
ФИО врача	Врач																		▼
Диагноз:	Диагноз																		

Рис. 13. Вид формы

Присвоим созданной форме название «фПрием». В выпадающем меню выбираем команду «Открыть».

Последовательность перехода при заполнении полей формы можно указать через «Свойства поля» на закладке «Другие» в параметрах «Индекс перехода по Tab». Первое поле имеет индекс «0».

3.1.5. Создание запросов

Запрос (*query*) – это средство СУБД или соответствующим образом сформированный вопрос, позволяющие производить выбор необходимой информации из базы данных.

Так как наша база нормализована и необходимые нам данные хранятся в разных таблицах, запрос обеспечит отбор информации из разных таблиц и объединение ее в виде формы или отчета. Чаще всего на наши вопросы дает ответ «запрос на выборку».

Попробуем сформировать запрос, который даст ответ на вопрос «Из каких районов чаще всего обращались в нашу клинику?».

Основная информация о приеме животных хранится в базе «Прием», названия районов – «спРайон».

В названии запросов будем добавлять букву «з». Создадим простой запрос.

1. Открываем ленту «Создание».
2. Выбираем команду «Конструктор запросов».
3. Из появившегося диалогового окна (рис. 14) выбираем таблицы, которые будут использованы при формировании запроса. Это таблица «Прием» и «спРайон». После закрытия этого окна мы увидим добавленные таблицы с линиями связи.

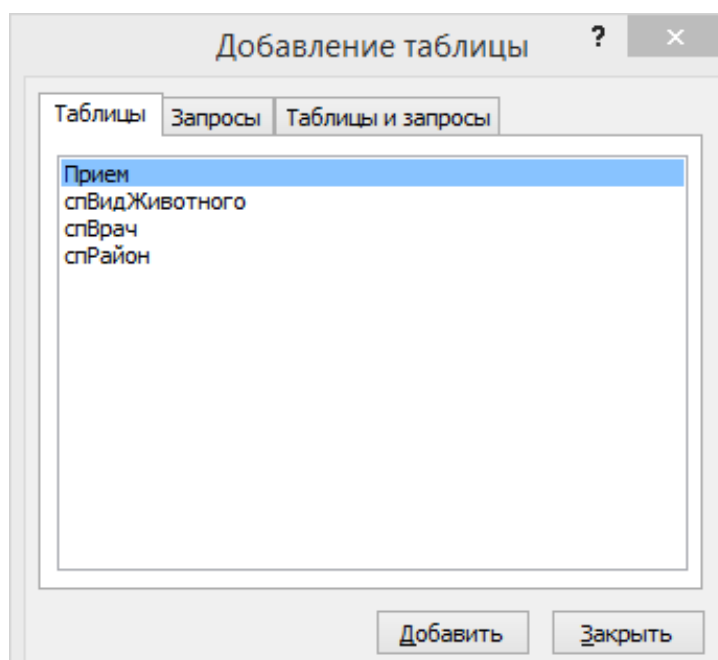


Рис. 14. Окно добавления таблиц при создании запроса

4. В нижней части экрана имеется форма с графами, в которых мы должны выбирать поля и указать дополнительные условия (значения для поиска, группировка данных, выполнение действий: подсчет количества, суммы, мин. значения, средней арифметической и т.д.).

а) в первой графе в *строке* «Имя таблицы:» выберем таблицу «спРайон»;

б) в *строке* «Поле:» выберем поле «Название»;

в) указатель мышки размещаем в первой графе и нажимаем на правую кнопку мышки. В появившемся выпадающем меню выбираем первый пункт – «Итоги»;

г) в перечне строк (слева) появится строка с названием «Групповая операция:». Нам необходимо выбрать пункт «Группировка»;

д) во второй графе в *строке* «Имя таблицы:» выберем таблицу «Прием»;

е) в *строке* «Поле:» выберем поле «Код»;

ж) в строке «Групповая операция:» выбирает команду «Count»;

з) сохраняем запрос с присвоением имени «зРайон».

Внешний вид окна конструктора запросов имеет вид, представленный на рис. 15.

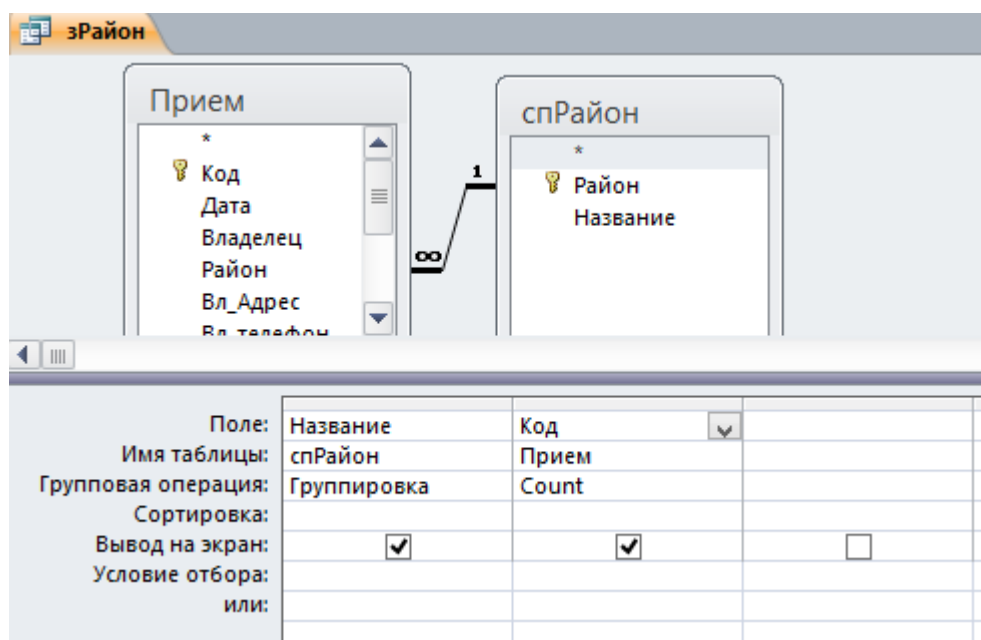


Рис. 15. Окончательный вид окна построения простого запроса

Открываем созданный нами запрос. В новой таблице будет два столбца. В первом «Название» указаны названия районов, по которым есть информация в базе «Прием», во втором – «Count-Код» указывается количество записей в базе «Прием», свидетельствующих о приеме животных из этого района.

5. Усложним задание и сформируем перекрестный запрос, в котором будут указаны не только названия районов, но и дана расшифровка о количестве и видах животных, которых приводили на прием в нашу клинику.

5.1. Открываем ленту «Создание».

5.2. Выбираем команду «Мастер запросов».

5.3. Указываем вид нового запроса «Перекрестный запрос», «Ок».

5.4. Выбираем таблицу, на основе которой будет создан запрос «Прием» двойным нажатием на левую кнопку мыши.

5.5. Из списка доступных полей выбираем «Район». Название этого поля переносится в окно «Выбранные поля», а в образце новой таблицы в названиях строк появляются названия «Район1», «Район2», «Район3» и т.д.

5.6. Следующим шагом выбираем поле «ВидЖивотного», которое будет использовано в качестве заголовков столбцов.

5.7. На следующем шаге (рис. 16) выбираем поле «Код» по которому будут производиться вычисления и задаем функцию «Число» или «Count» (подсчет количества записей).

Создание перекрестных таблиц

Какие вычисления необходимо провести для каждой ячейки на пересечении строк и столбцов?

Например, можно вычислить сумму заказов для каждого сотрудника (столбец) по странам и регионам (строка).

Вычислить итоговое значение для каждой строки?

☒ Да.

Поля:

- Код
- Дата
- Владелец
- Вл_Адрес
- Вл_телефон
- Врач
- Диагноз
- Кличка
- Возраст

Функции:

- Min
- StDev
- Var
- Максимум
- Первый
- Последний
- Среднее
- Сумма
- Число

Образец:

Район	ВидЖивотно	ВидЖивотно	ВидЖивотно
Район1	Число(Код)		
Район2			
Район3			
Район4			

Отмена < Назад **Далее >** Готово

Рис. 16. Окно создания перекрестного запроса

5.8. Последним действием будет присвоение имени новому запросу «зРайонВид». Но так как мы еще не завершили работу над запросом, то выбираем команду «Изменить структуру запроса». После чего мы перейдем в режим «Конструктора».

5.9. Внесем изменения в наш запрос.

а) добавим с помощью команды «Отобразить таблицу» справочники «спРайон» и «спВидЖивотного». Окно конструктора запроса имеет вид, представленный на рис. 17.

зРайонВид

спРайон

- * Район
- Название

1

∞

Прием

- * Код
- Дата
- Владелец
- Район
- Вл_Адрес
- Вл_телефон

1

∞

спВидЖивотного

- * ВидЖивотного
- Название

Поле:	[Район]	[ВидЖивотного]	[Код]	Итоговое значение
Имя таблицы:	Прием	Прием	Прием	Прием
Групповая операция:	Группировка	Группировка	Count	Count
Перекрестная таблица:	Заголовки строк	Заголовки столбцов	Значение	Заголовки строк
Сортировка:				
Условие отбора:				

Рис. 17. Окно конструктора перекрестного запроса

- б) в первой графе в строке «Имя таблицы:» выберем таблицу «спРайон»;
- в) в строке «Поле:» выберем «Название»;
- г) во второй графе в строке «Имя таблицы:» выберем таблицу «спВидЖивотного»;
- д) в строке «Поле» выберем «Название:»;
- е) сохраняем полученный запрос. После выполнения всех настроек внешний вид окна представлен на рис. 18.

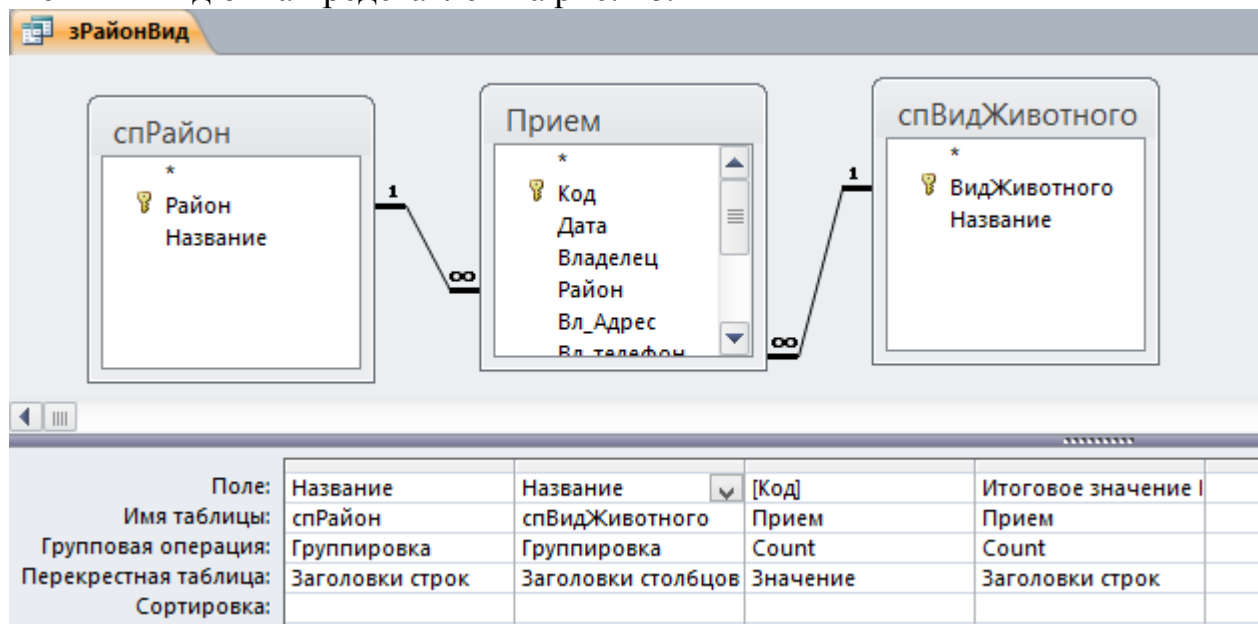


Рис. 18. Окончательный вид окна конструктора перекрестного запроса

Открываем построенный запрос и получаем таблицу, в которой в строках указаны названия районов, а в заголовках граф – «Итоговое значение Код» и виды животных. Значениями являются сведения о количестве животных, в разрезе их видов, поступавших в нашу клинику.

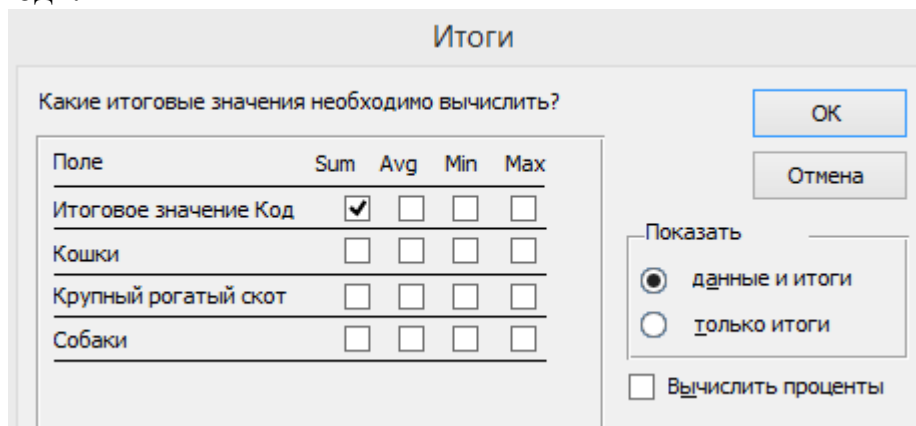
3.1.6. Создание отчета

Отчет – это форматированное представление данных, которое выводится на экран, на печать или файл в виде, удобном для восприятия.

Отчет может строиться по данным таблицы или запроса.

1. Сформируем отчет по запросу «зРайонВид». Выделим его в списке.
2. Перейдем на ленту «Создание» и выберем команду «Мастер отчетов».
3. Перенесем все «Доступные поля» в окно «Выбранные поля». «Далее».
4. На вопрос «Добавить уровни группировки» выберем поле «Название». «Далее».
5. На этом этапе произведем настройку вычисления итоговых значений через кнопку «Итоги...».

6. В диалоговом окне «Итоги» (см. рис. 19) укажем на необходимость суммирования (функция «Sum») значений по полю запроса «Итоговое значение Код».



Поле	Sum	Avg	Min	Max
Итоговое значение Код	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Кошки	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Крупный рогатый скот	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Собаки	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Показать

☒ данные и итоги

☐ только итоги

☐ Вычислить проценты

Рис. 19. Настройка вычисляемых полей

7. Дополнительных параметров сортировки задавать не будем и перейдем к следующему шагу («Далее»).

8. Из предложенного списка видов макета отчета выберем «блок» и ориентацию бумаги «книжная». «Далее».

9. Присвоим название «oРайонВид» нашему первому отчету и посмотрим на результат нажав на кнопку «Готово».

Библиографический список

1. Бекаревич Ю.Б. Самоучитель Access 2010 / Ю.Б. Бекаревич, Н.В. Пушкина.– СПб.: БХВ-Петербург, 2011.– 426 с.
2. Гвоздева В.А. Базовые и прикладные информационные технологии: Учебник / В.А. Гвоздева.– М.: ИД ФОРУМ, НИЦ ИНФРА-М, 2015.– 384 с.
3. Гурвиц Г.А. Microsoft Access 2010. Разработка приложений на реальном примере / Г.А. Гурвиц.– СПб: БХВ-Петербург, 2012.– 497 с.
4. Карпузова В.И. Информационные технологии в менеджменте: Учебное пособие / В.И. Карпузова, Э.Н. Скрипченко, К.В. Чернышева, Н.В. Карпузова.– М.: Вузовский учебник: НИЦ ИНФРА-М, 2014.– 301 с.
5. Кузин А.В. Разработка баз данных в системе Microsoft Access / А.В. Кузин, В.М. Демин.– М.: Форум, 2007.– 223 с.
6. Обухов Н.П. Разработка баз данных в Microsoft Access / Н.П. Обухов.– СПб.: ИВЭСЭП, Знание, 2008.– 92 с.
7. Одиночкина С.В. Разработка баз данных Microsoft Access 2010: Учебно-методическое пособие / С.В. Одиночкина.– СПб.: НИУ ИТМО, 2012.– 83 с.
8. Хомоненко А.Д. Базы данных / А.Д. Хомоненко, В.М. Цыганков, М.Г. Мальцев.– СПб.: КОРОНА принт, 2004.– 736с.

Составители:

**Наумкин Игорь Викторович
Лазарева Марина Викторовна**

Основные понятия и принципы работы Access 2010

**Методические указания
к лабораторно-практическим занятиям**

Редактор	Т.К. Крупина
Компьютерная верстка	И.В. Наумкин

Подписано в печать

Формат: 60x84 ¹/₁₆

Объем: 2,0 уч. изд. л.

Тираж 100 экз.

Изд. №

Заказ №

630039, г. Новосибирск, ул. Добролюбова, 160, Издательский центр НГАУ