

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К КУРСОВОМУ ПРОЕКТИРОВАНИЮ по дисциплине «Управление данными»

Изучение дисциплины Управление данными предполагает занятия в лекционной и практической форме, а также курсовое проектирование.

Курсовой проект по дисциплине посвящен разработке реляционных баз данных в среде MS Office Access.

В техническом задании на разработку курсового проекта предложена структура пояснительной записки к нему:

1. титульный лист,
2. лист технического задания,
3. содержание,
4. введение,
5. раздел, раскрывающий инфологическое проектирование (краткое описание предметной области, инфологическая модель),
6. раздел, раскрывающий даталогическое проектирование (модель данных, краткая характеристика СУБД, схема базы данных, таблицы и их характеристика, формы, запросы, отчеты),
7. выводы,
8. литература.

Рассмотрим их последовательно.

Титульный лист включает размещенные в верхней части листа название университета и название факультета, размещенные в средней части листа (по центру) название дисциплины, слова «Курсовой проект на тему» и тему курсового проекта. Ниже справа необходимо указать кто, выполнил курсовой проект (фамилия, имя, отчество, группа) и кто проверил - фамилия, имя, отчество преподавателя. В нижней части листа указывают название города, где размещается университет, и год написания курсового.

Лист технического задания берется из приложения. В него вписывается фамилия, имя, отчество студента, который выполняет данный курсовой проект, и дату получения задания.

Содержание – это оглавление курсового проекта, которое должно «собираться» автоматически путем использования стилей и специального инструмента текстового процессора MS Office Word.

Введение – это структурный элемент курсового проекта, в котором должна быть отражена актуальность темы. Актуальность темы - это степень ее важности в данный момент и в данной ситуации для решения данных проблемы, вопроса или задачи.

При обосновании актуальности темы курсового проекта необходимо объяснить, причину, почему тема исследования назрела именно в данный момент, что являлось препятствием правильному раскрытию её ранее, выявить, насколько обращение к данной теме обусловлено развитием науки, накоплением новой информации и новыми методами исследования, необходимостью изучения проблемы исследования в новой экономической ситуации и т. д.

При раскрытии актуальности темы исследования можно

– указать на наличие некой проблемы, решение которой пока не найдено, а данное исследование (курсовой проект) направлено на поиск решения этой проблемы;

– отразить тот факт, что тема является элементом некой сложной системной проблемы, которую невозможно решить, пока не выяснена сущность каждого из её элементов, то есть определенные аспекты темы изучены не в полной мере и курсовая работа направлена на преодоление этого пробела;

– связать актуальность с возможностью решения определенной практической задачи на основе полученных в исследовании данных.

Освещение актуальности не должно быть многословным, для курсового проекта достаточно полстраницы или страницы машинописного текста, необходимо показать главное.

Раздел, раскрывающий инфологическое проектирование, должен включать анализ предметной области базы данных, разработку и описание инфологической модели, ее ER-диаграмму.

Первоочередной задачей является анализ предметной области разрабатываемой базы данных и определение ее границ. Предметная область – это некоторая часть реального мира, полнота описания которой определяется объемами собранной информации. Эту информацию разработчик берет как из технического задания, так и из литературных и интернет источников. За соответствие базы данных предметной области отвечает проектировщик, который должен адекватно представлять себе все нюансы изучаемой области и уметь отобразить их в базе данных.

Пусть предметной областью разрабатываемой базы данных является работа приемной комиссии учебного заведения. Необходимо определить тип учебного заведения – общее, среднее или высшее, рассмотреть и описать процедуру организации приема.

Это прежде всего прием от абитуриентов необходимых для поступления документов, их проверка, регистрация в системе учета в соответствии с выбранными специальностями, распределение всех зарегистрированных абитуриентов по потокам и группам. Для каждой группы должны быть назначены даты проведения консультаций, даты вступительных экзаменов, аудитории, где данные мероприятия будут проводиться. Условия выделения аудиторий могут быть различными, к примеру, в аудитории могут быть выделены для нескольких групп одновременно в зависимости от предмета, по которому проводятся консультации и экзамены. В системе учета может быть (или должен быть) предусмотрен ввод результатов экзаменов каждого абитуриента с возможностью изменения их при подаче апелляции, обобщение результатов, определение списка поступающих в ВУЗ в соответствии с принятыми критериями и др.

Проанализировав эту информацию, можно приступить к разработке инфологической модели.

Инфологическое проектирование позволяет организовать и упорядочить информацию о предметной области базы данных. Для нормального функционирования создаваемой системы необходимо, чтобы инфологическая (концептуальная, семантическая) модель адекватно, полно и корректно отображала реалии той предметной области, для которой она разрабатывается.

Инфологическое проектирование можно определить, как построение инфологической модели предметной области, без ориентации на какую-либо конкретную СУБД.

Наиболее популярна инфологическая модель – «сущность-связь», компоненты которой сущности, атрибуты и связи.

Сущность – это абстракция объекта предметной области, о котором должна накапливаться информация и что может быть однозначно определено (идентифицировано). Множество однородных объектов реального мира с одинаковыми или схожими свойствами определяют тип сущности.

Следующий элемент модели – атрибут, поименованная характеристика сущности, с помощью которой моделируется свойство объекта. Атрибут имеет имя (название) и принимает значения из некоторого множества, включающего все потенциальные значения, которые могут быть присвоены.

Сущности могут быть связаны между собой. Подобная связь осуществляется через связь экземпляров одной сущности с экземплярами другой сущности. Таким образом, связь является средством представления отношений между сущностями. В большинстве случаев, при проектировании базы данных, используются бинарные связи, когда в отношении участвуют две сущности. При этом, бинарные связи можно разделить на несколько видов:

- «один к одному» (1:1) – каждому элементу из первого множества соответствует только один элемент из второго множества, и наоборот;
- «один ко многим» (1:M) – одному элементу из первого множества может соответствовать несколько элементов из второго множества, но в обратном направлении, элементу из второго множества соответствует лишь одно значение из первого множества
- «многие ко многим» (M:M) – нескольким элементам из первого множества может соответствовать несколько элементов из второго множества, и наоборот.

Как правило, семантическая модель предметной области изображается в виде диаграммы, на которой сущность отображается в виде прямоугольника, атрибуты в виде овалов, а связи – ромбов.

Рассмотрим пример. Анализ предметной области разрабатываемой базы данных для распространителя газетной продукции позволил выделить объекты – газеты, типография, почтовые отделения, которым были поставлены сущности:

- Газеты;
- Типографии;
- Почтовые отделения.

Сущность «Газеты» имеет следующие атрибуты (характеристики): шифр, название, количество газет данного наименования, ФИО редактора, номер типографии, где печатается.

Сущность «Типографии» имеет такие атрибуты: номер, адрес, количество газет данного наименования, печатающихся в данной типографии и ее статус: закрыта она или открыта.

Сущность «Почтовые отделения» характеризуется номером, адресом, названием поступающих на каждое отделение газет и их количеством.

Атрибуты «Шифр» сущности «Газеты», «Номер» сущности «Типография» и «Номер_отд» сущности «Почтовые отделения» являются ключевыми, то есть их

значения уникальны, не повторяются, что определяет уникальность (неповторимость) экземпляров сущности.

Анализ значений атрибутов показывает, что в данной модели имеется два типа значений атрибутов - текстовые (Название, Адрес, и т.д.) и числовые (Количество экземпляров, и т.д.).

Отношения определяют связи между сущностями. Типография «печатает» разные газеты, а каждая газета «печатается» только в одной типографии. Имеем бинарное отношение «Типография»-«Газета» - «один-ко-многим» (1:M). «Газета» и «Почтовое отделение» связаны другим отношением: газета «отправляется» в разные «почтовые отделения», а почтовые отделения «получают» разные газеты. Это бинарное отношение «многие-ко-многим» (M:M).

ER-диаграмма представлена на рисунке 1.

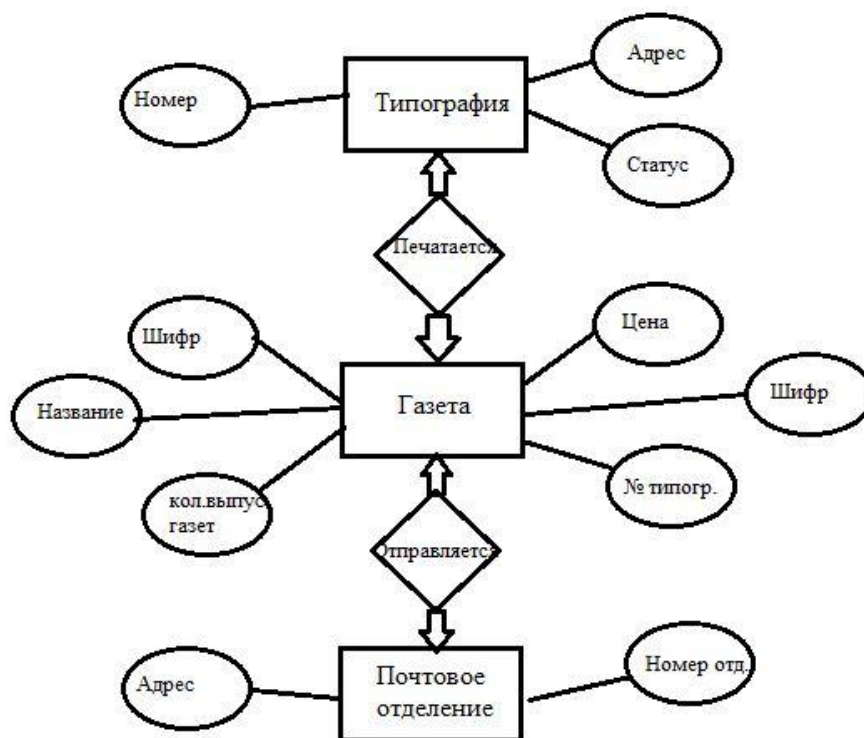


Рисунок 1 – ER-диаграмма модели базы данных

Раздел, раскрывающий даталогическое проектирование, включает краткую характеристику СУБД, описание модели (схемы) данных, характеристику основных объектов СУБД - таблиц, форм, запросов, отчетов,

Инфологическое моделирование проводится без ориентации на конкретную СУБД, поэтому на следующем этапе создания базы данных – даталогическом проектировании, необходимо определиться с выбором конкретных программных средств для реализации поставленной задачи и преобразовать инфологическую модель в даталогическую.

Даталогическая модель включает в себя допустимые форматы данных, команд и состав операций, выполняемых над ними.

На этом этапе выбирается какая-либо модель данных – иерархическая, сетевая, реляционная или гибридная. Каждая модель имеет свои свойства и особенности,

поэтому к ее выбору следует подойти основательно, потому что от этого зависит дальнейшая реализация базы данных.

Иерархическая модель представляет данные в иерархическом порядке, она направлена на описание объектов, находящихся между собой в отношении подчинения.

Сетевая модель данных позволяет описать более сложные отношения, в которых присутствуют не только иерархические, но и одноуровневые зависимости. Она сложна в реализации и работе с данными.

Наконец, реляционная модель данных является самой распространенной, универсальной и простой в моделировании. Данные представлены в виде таблиц, которые, в свою очередь, состоят из строк и столбцов. Соответственно, каждая строка таблицы – это информация о конкретном объекте, каждый столбец представляет свойства этого объекта.

Реляционную модель чаще всего и выбирают для проектирования баз данных. Для работы с реляционной моделью базы данных можно использовать программное средство Microsoft Office Access – реляционную СУБД, имеющую широкий спектр функций, включающий связанные запросы, связь с внешними таблицами и базами данных.

Благодаря пользовательскому интерфейсу Microsoft Office Access и интерактивным возможностям, не требующим глубоких знаний базы данных, приложение позволяет с легкостью отслеживать данные и составлять отчеты. Кроме того, приложение поддерживает интеграцию с другими программными средствами пакета Microsoft Office, что также является большим преимуществом.

Основными объектами любой создаваемой базы данных являются: таблицы, запросы, формы и отчеты. Создание таблиц, определение их полей, типов данных и их свойств, а также связей между таблицами определяет структуру разрабатываемой базы данных, ее схему или, как говорят, даталогическую (логическую) модель данных.

При переходе от инфологической модели к даталогической, необходимо уделить особое внимание бинарным связям «многие ко многим». В СУБД Access невозможно напрямую реализовать такую связь, поэтому необходимо ввести дополнительную таблицу, которая будет выражать требуемое отношение.

Так в рассмотренном выше примере по разработке инфологической модели необходимо решить проблему отношения «многие ко многим», поскольку в одном почтовом отделении может быть несколько наименований газет, и в тоже время одна и та же газета может находиться в нескольких отделениях сразу. Решением станет введение дополнительной таблицы, и связь «многие-ко-многим» заменится на связи «один-ко-многим» и «многие-к-одному». Даталогическая модель представлена на рисунке 2.

Далее необходимо описать разработанные объекты базы данных и привести их «скриншоты».

Например, на рисунках 3 и 4, соответственно, представлена таблица «Газета» в режиме Конструктора и в режиме Таблица.

Можно не представлять таблицу «Газета» в режиме Конструктора (рисунок 3), а описать свойства отдельных полей этой таблицы иначе, указав в отдельной таблице значения наиболее употребимых свойств таблицы «Газета» (рисунок 5). Но

необходимо в курсовом представить все заполненные таблицы (рисунок 4), причем содержание таблиц (значения атрибутов) должно отвечать теме курсового проекта.

Для ввода данных в таблицы используют такой объект СУБД как формы, хотя формы сегодня используют и для представления содержимого базы данных на экране компьютера.

Существуют разные виды форм: в столбец, ленточная, табличная. Если форма в столбец позволяет отобразить на экране только одну запись (одну строку) таблицы, то ленточная и табличная – группу записей.

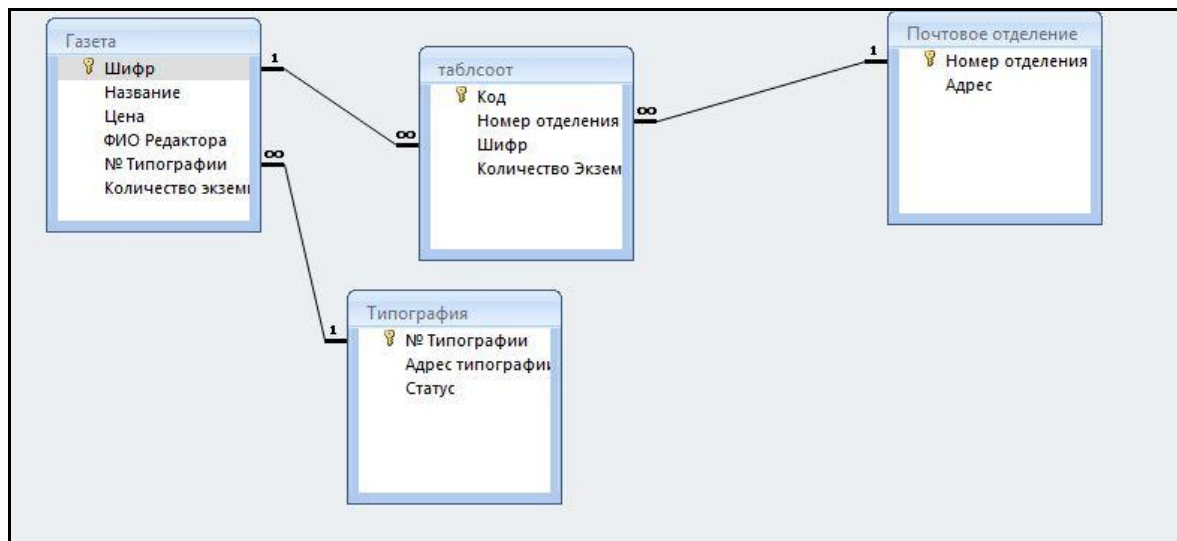


Рисунок 2 – Схема данных

| Имя поля | Тип данных |
|------------------------|------------|
| Шифр | Числовой |
| Название | Текстовый |
| Цена | Числовой |
| ФИО Редактора | Текстовый |
| № Типографии | Текстовый |
| Количество экземпляров | Числовой |

| Общие | | Подстановка | |
|-------------------------|--------------------------------|---------------|--|
| Размер поля | | Длинное целое | |
| Формат поля | | | |
| Число десятичных знаков | Авто | | |
| Маска ввода | | | |
| Подпись | | | |
| Значение по умолчанию | | | |
| Условие на значение | | | |
| Сообщение об ошибке | | | |
| Обязательное поле | Нет | | |
| Индексированное поле | Да (Совпадения не допускаются) | | |
| Смарт-теги | | | |
| Выравнивание текста | Общее | | |

Рисунок 3 – Таблица «Газета» в режиме Конструктора

| | Шифр | Название | Цена | ФИО Редактора | № Типографии | Количество |
|---|------|-----------|------|--------------------|--------------|------------|
| + | 100 | Известия | | 17 Белов И.В. | 2-2 | 1000 |
| + | 200 | Жизнь | | 18 Иванов С.Ю. | 1-1 | 50000 |
| + | 300 | ЗОЖ | | 12 Ивлев А.А. | 2-2 | 10000 |
| + | 400 | Власть | | 15 Рубинштейн Л.А. | 3-3 | 7500 |
| + | 500 | Экономика | | 20 Рубинштейн Л.А. | 3-3 | 9200 |
| + | 600 | Спорт | | 22 Рабинер И.С. | 1-1 | 13000 |
| * | | | | | | |

Рисунок 4 – Таблица «Газета» в режиме Таблица

| Название поля | Характеристика поля | Тип данных | Размер данных | Индексирование поля |
|------------------------|--|------------|---------------|---------------------|
| Шифр | Ключевое поле | Числовой | Длинное целое | Да |
| Название | Название газеты | Текстовый | 70 | Нет |
| Цена | Цена 1 экземпляра | Числовой | Байт | Нет |
| ФИО Редактора | ФИО редактора газеты | Текстовый | 40 | Нет |
| № Типографии | Номер типографии | Текстовый | Байт | Нет |
| Количество экземпляров | Количество выпущенных экземпляров газеты | Числовой | Целое | Нет |

Рисунок 5 – Характеристика полей таблицы «Газета»

Разделенная форма позволяет вводить данные в таблицу и отслеживать вносимые изменения в реальном времени. Пример такой формы приведен на рисунке 6.

| | |
|-------------------------|--------|
| Шифр: | 900 |
| Название: | Восток |
| Цена: | 43 |
| ФИО Редактора: | Цынь |
| № Типографии: | 2-2 |
| Количество экземпляров: | 1445 |

| Шифр | Название | Цена | ФИО Редактора | № Типографии | Количество |
|------|-----------|------|-----------------|--------------|------------|
| 200 | Жизнь | 18 | Иванов С.Ю. | 1-1 | 50000 |
| 300 | ЗОЖ | 12 | Ивлев А.А. | 2-2 | 10000 |
| 400 | Власть | 15 | Рубинштейн Л.А. | 3-3 | 7500 |
| 500 | Экономика | 20 | Рубинштейн Л.А. | 3-3 | 9200 |
| 600 | Спорт | 22 | Рабинер И.С. | 1-1 | 13000 |
| 900 | Восток | 43 | Цынь | 2-2 | |

Рисунок 6 – Форма для ввода данных в таблицу Газета

Для получения данных из базы необходимо использовать специальные объекты СУБД – запросы. Запрос — это набор инструкций, который можно использовать для обработки данных. Чтобы эти инструкции были выполнены, запрос следует запустить.

Запрос не только возвращает результаты, которые можно сортировать, группировать и фильтровать. С помощью запроса можно также создавать, копировать, удалять и изменять данные.

Наиболее часто используют запрос на выборку, который возвращает данные из одной или нескольких таблиц, а также результаты, которые при желании пользователь может изменить. При помощи такого запроса можно также передавать данные в другие объекты базы данных. После создания запроса на выборку его можно использовать по мере необходимости.

Можно использовать запрос на выборку, чтобы сгруппировать записи для вычисления сумм, средних значений, пересчета и других действий.

Запрос может получать данные из одной или нескольких таблиц, из существующих запросов или из комбинаций таблиц и запросов. Таблицы или запросы, используемые для получения данных, называются источниками записей.

Чаще всего запрос создается в режиме Конструктора, в котором открывается специальный бланк запроса, состоящий из 2-х областей:

- в верхней размещается структура таблиц, к которым адресован запрос,
- в нижней – столбцы результирующей таблицы, которая будет сформирована в результате выполнения запроса; один столбец – одно поле результирующей таблицы

Строки Поле и Имя таблицы содержат списки, которые позволяют определить нужное поле. Их элементы соответствуют размещенной сверху схеме, поэтому пользователь добавляет в запрос новые поля.

Контекстное меню, связанное с бланком запроса позволяет скрывать или показывать строки “Имя таблицы” и “Групповые операции”, а также добавлять в верхнюю часть бланка таблицы, необходимые для построения запроса.

Строки Сортировка, Вывод на экран, Условие отбора служат для формирования структуры запроса и его оформления.

Сначала с помощью контекстного меню в верхней половине бланка запроса открываются те таблицы, к которым обращен запрос. Если пользователь составляет запрос на основе нескольких таблиц, то между ними необходимо объявить отношение, связывая поля таблиц друг с другом, иначе результат запроса может быть некорректным. Связи между таблицами должны соответствовать схеме данных.

Затем поля, которые должны войти в результирующую таблицу перемещают в нижнюю часть бланка и формируют структуру запроса - определяют условие отбора, порядок сортировки, создают вычисляемые поля для вычисления суммы, среднего значения, пересчета и других действий с данными, либо изменяют структуру запроса.

Так в рассмотренном ранее примере запрос на выдачу ФИО редактора газеты, которая печатается в заданной типографии самым большим тиражом, формируется в режиме Конструктора как запрос на выборку с параметром, когда критерии в условие запроса вводятся в процессе диалога пользователя с базой данных (рисунки 7 и 8).

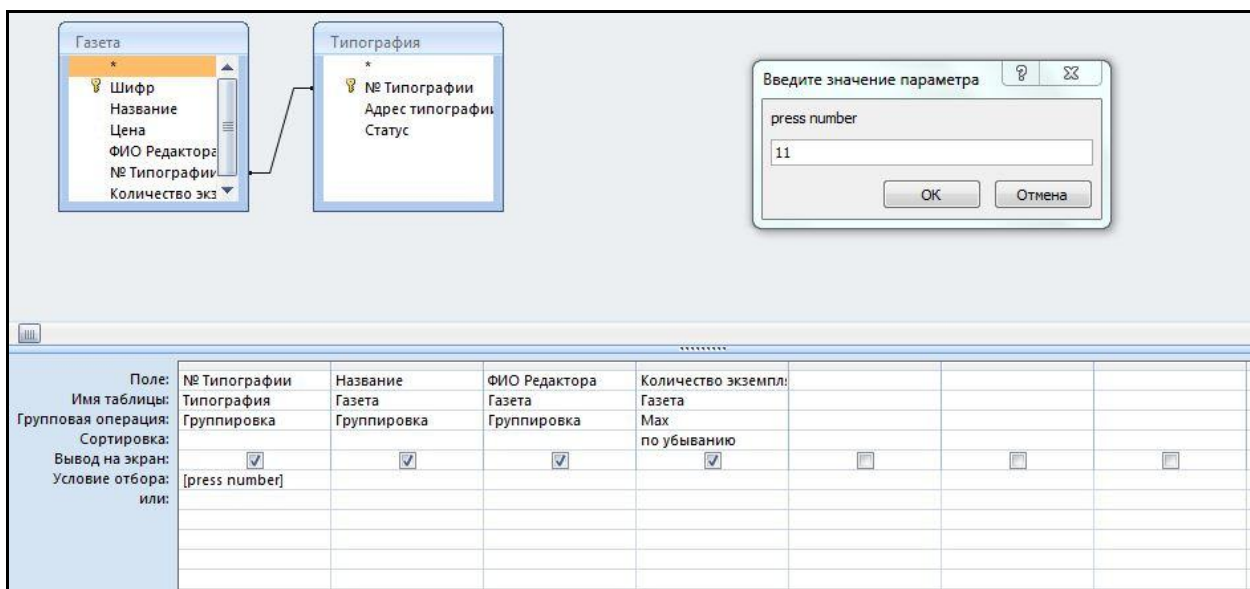


Рисунок 7- Запрос на выдачу фамилии редактора газеты с наибольшим тиражом

| № Типограф | Название | ФИО Редактора | Max-Колич |
|------------|----------|---------------|-----------|
| 11 | Жизнь | Иванов С.Ю. | 50000 |

Рисунок 8 - Результат выполнения запроса

Есть еще один объект базы данных – отчеты. Отчеты похожи на формы, но имеют другое функциональное назначение – обеспечивают форматированный вывод на печатающее устройство.

При их создании можно использовать Мастер отчетов, по которому выбирают базовые таблицы и запросы, на которых базируется отчет, выбирают поля, используемые в отчете, выбирают поля группировки, поля и методы сортировки, форму печатного макета и стиль оформления.

Структура отчета отличается от структуры формы количеством разделов – используют еще нижний и верхний колонтитулы.

Редактирование структуры отчета выполняется в режиме Конструктора. Приемы редактирования те же, что и для форм. Элементы управления выполняют здесь функции элементов оформления.

Особенности отчетов – вставка в область верхнего и нижнего колонтитулов текущего номера страницы и полного количества страниц.

Отчеты позволяют вывести на экран или на печать наиболее общую информацию из базы данных. На рисунке 9 представлен отчет о работе типографий.

| Типография | | | | | | |
|------------------|--------|--------------|----------|------------|---------------|--|
| Адрес типографии | Статус | № Типографии | Название | кземпляров | ФИО Редактора | |
| ул. Мира 70 | Отк | 11 | Жизнь | 50000 | Иванов С.Ю. | |
| | | | Спорт | 13000 | Рабинер И.С. | |
| | | | | | | |
| ул. Правды 15 | Отк | 22 | Восток | 1445 | Цынь | |
| | | | ЗОЖ | 10000 | Ивлев А.А. | |
| | | | Известия | 1000 | Белов И.В. | |
| | | | | | | |

Рисунок 9 - Отчет о работе типографий

Выводы курсовой работы должны отражать процесс работы студента над исследованием. В выводах необходимо перечислить, что сделано и какие результаты получены.

Литература должна быть не только правильно оформлена, но и на нее в курсовой работе обязательно должны быть ссылки.

Требования к оформлению курсовой работы.

Курсовая работа должна быть оформлена согласно нормативным документам:

- ГОСТ 7.32-2017 «Отчет о научно-исследовательской работе. Структура и правила оформления»;
- ГОСТ Р 7.0.5-2008 «Библиографическая ссылка. Общие требования и правила составления»;
- ГОСТ 7.1-2003 «Библиографическая запись. Библиографическое описание. Общие требования и правила составления»;
- ГОСТ 7.80-2000. «Библиографическая запись. Заголовок. Общие требования и правила составления»;

Литература

2. Базы данных - <http://www.intuit.ru/department/database/dbmdi/7/>.
3. Проектирование реляционных баз данных - <http://rema44.ru/resurs/study/dbprj/dbprj.html>.
4. Инфологическое проектирование баз данных - <http://wiki.mvtom.ru/>.
5. База данных Access - <http://www.tepka.ru/msoffice/1601.html>.
6. Кузнецов С.Д. Введение в реляционные базы данных. –М: НОУ “Интуит”, 2016, 48 с.
- 7 Карпова Т.С. БД, модели, разработки, реализация – М.: НОУ "Интуит", 2016, 403с .

- 8 Маклаков С.В. Создание информационных систем с ALLFusion Model Suite. – М: Диалог, МИФИ, 2013, 396 с.
- 9 Пржиялковский В.В. Введение в ORACLE SQL – М: НОУ “Интуит”, 2016, 358 с.
- 10 Мишра С, Бьюли А. Секреты Oracle SQL – СПб: Символ-Плюс, 2016, 368 с.