



Филиал национального исследовательского
технологического Университета «МИСиС» в
городе Душанбе

***Кафедра «Информационные технологии и
автоматизации»***

Составитель Худжов И.И

МЕТОДИЧЕСКИЕ УКАЗАНИЯ
по выполнению курсовой работы для студентов направления
09.03.01 «Информатика и ВТ» по дисциплине «Базы данных»

Душанбе-2017

Методические указания по выполнению курсовой работы по дисциплине «Базы данных» для студентов направления подготовки 09.03.01 "Информатика и вычислительная техника" рассмотрена и утверждена на заседании кафедры «Информационные технологии и автоматизация»

ОГЛАВЛЕНИЕ

Введение	4
1. Рекомендуемая структура отчета курсовой работы	5
2. Описание содержания разделов отчета курсовой работы	6
2.1. Функциональные возможности системы	6
2.2. Проектирование базы данных	7
2.2.1. <i>Концептуальное проектирование</i>	7
2.2.2. <i>Логическое проектирование</i>	8
2.2.3. <i>Описание информационных объектов базы данных</i>	9
2.2.4. <i>Схема данных</i>	13
2.3. Проектирование и реализация алгоритмов обработки данных	13
2.3.1. <i>Моделирование алгоритмического поведения системы</i>	13
2.3.2. <i>Обработка данных средствами запросов</i>	18
2.4. Проектирование и реализация приложения пользователя	20
2.4.1. <i>Структура приложения пользователя</i>	20
2.4.2. <i>Проектирование форм</i>	21
2.4.3. <i>Проектирование отчетов</i>	24
2.4.4. <i>Разработка средств управления и навигации</i>	27
2.5. Краткое руководство пользователя	31
2.6. Заключение	34
2.7. Литература	35
2.8. Приложение	35
Критерии оценки	36
Список литературы	41
Приложения	42

Введение

Целью курсовой работы является формирование умений по проектированию и реализации базы данных для выбранной предметной области на основе полученных теоретических знаний.

Студент в процессе выполнения курсовой работы должен продемонстрировать владение методикой моделирования выбранной предметной области средствами баз данных, умение реализовывать базу данных в определенной системе управления базами данных и создать приложение пользователя.

Выбор темы курсовой работы осуществляется из списка рекомендованных кафедрой тем (приведены в приложении А). Студент может предложить свою тему, при этом она должна быть согласована с руководителем.

В процессе выполнения курсовой работы студент изучает современную и классическую литературу по теме “Базы данных”. На любом этапе он может получить консультацию у руководителя. Для своевременного выполнения курсовой работы рекомендуется соблюдать график (приложение Б), согласованный с руководителем.

Каждый этап выполнения работы проверяется и оценивается руководителем. При наличии недостатков по содержанию или оформлению каждого этапа курсовая работа направляется на доработку. В этом случае студент должен устранить указанные недостатки в кратчайшие сроки и сдать работу на повторную проверку.

При положительной оценке курсовая работа допускается к защите. Решение о допуске к защите принимается руководителем не менее, чем за один день до защиты курсовой работы.

На защите курсовой работы студент делает доклад о содержании своей работы, демонстрирует работу приложения, отвечает на задаваемые вопросы. Длительность доклада регламентируется и обычно составляет не более 10 минут. После защиты курсовой работы студент получает окончательную оценку, которая проставляется в зачетной книжке и в экзаменационной ведомости.

Студент, не защитивший курсовую работу, не допускается к сдаче экзамена по данной дисциплине.

1. РЕКОМЕНДУЕМАЯ СТРУКТУРА ОТЧЕТА КУРСОВОЙ РАБОТЫ

Отчет по курсовой работе должен быть оформлен в соответствии с требованиями к оформлению письменных работ, выполняемых студентами и слушателями ДВФУ и должен включать перечисленные ниже разделы.

Титульный лист.

Оглавление.

Введение.

1. Функциональные возможности системы.

2. Проектирование базы данных.

2.1. Концептуальное проектирование.

2.2. Логическое проектирование.

2.3. Описание информационных объектов базы данных.

2.4. Схема данных.

3. Проектирование и реализация алгоритмов обработки данных.

3.1. Моделирование алгоритмического поведения системы.

3.2. Обработка данных средствами запросов.

4. Проектирование приложения пользователя.

4.1. Структура приложения пользователя.

4.2. Проектирование форм.

4.3. Проектирование отчетов.

4.4. Разработка средств управления и навигации.

5. Краткое руководство пользователя.

Заключение.

Литература.

Приложение.

2. ОПИСАНИЕ СОДЕРЖАНИЯ РАЗДЕЛОВ ОТЧЕТА КУРСОВОЙ РАБОТЫ

Описание содержания разделов будет осуществляться на примере курсовой работы “Разработка и проектирование базы данных **Библиотека**”. Текст в начале каждого раздела, выделенный курсивом, является пояснением и не должен включаться в отчет.

*Во **введении** приводится полное описание предметной области, для которой проектируется и реализуется база данных. Ниже описана предметная область для нашего примера.*

В курсовой работе осуществляется проектирование и разработка приложения на основе баз данных для следующей предметной области.

В некоторой библиотеке хранятся книги. Каждая книга имеет название, относится к одному жанру, имеет одного или нескольких авторов, издана в определенном издательстве. В библиотеке может быть несколько экземпляров одной книги. Каждый экземпляр имеет уникальный шифр.

Пользователями библиотеки являются читатели, о которых в карточках хранится информация о фамилии, адресе и телефоне. Каждый читатель может взять несколько экземпляров книг, которые имеются в наличии. При этом в личной карточке читателя сохраняются сведения о взятой книге и дате ее получения. При возврате книги в карточке читателя указывается дата возврата, и данный экземпляр книги становится доступным для других пользователей.

2.1. Функциональные возможности системы

В данном разделе подробно описываются функциональные возможности проектируемой системы.

1. Занесение, просмотр и корректировка информации о книгах, представленных в библиотеке.
2. Занесение, просмотр и корректировка информации о читателях библиотеки.
3. Поиск книг по различным критериям.
4. Поиск читателей и данных о перечне взятых и сданных ими книг.
5. Организация выдачи и возврата книг пользователем.
6. Поддержка фонда книг в актуальном состоянии.
7. Формирование статистической информации за некоторый период:
 - списка должников;
 - списка книг по их популярности;
 - списка авторов по их популярности;

- ежедневной активности читателей;
- списка невозвращенных книг.

2.2. Проектирование базы данных

Процесс проектирования базы данных представляет собой последовательность переходов от словесного описания информационной структуры предметной области к формализованному описанию объектов предметной области в терминах некоторой модели. В общем случае выделяют следующие этапы проектирования:

- Анализ и описание предметной области информационной системы.
- Концептуальное моделирование.
- Построение логической модели, схемы базы данных.

2.2.1. Концептуальное проектирование

В этом разделе необходимо дать частично формализованное описание объектов предметной области в терминах некоторой семантической модели, например, в терминах ER-модели (модели “сущность-связь”).

Данная концептуальная модель обеспечивает интегральное представление о предметной области и имеет слабо формализованный характер, отображает информационные объекты, свойства и связи между ними.

*Согласно ER-модели, **информационный объект** изображается на диаграмме в виде прямоугольника, содержащего имя объекта. **Связь** изображается линией, которая связывает два информационных объекта, участвующих в отношении. Степень конца связи указывается графически, множественность связи изображается в виде «стрелки» на конце связи. Таким образом, для связи “один-к-одному” используется линия без стрелки, для связи “один-ко-многим” – линия со стрелкой на стороне второго объекта, для связи “многие-ко-многим” – линия со стрелками на сторонах обоих объектов. При необходимости **Атрибуты** информационного объекта записываются внутри прямоугольника, его изображающего.*

Для построения ER-диаграммы можно использовать online средство www.draw.io, находящееся в открытом доступе.

На основе анализа предметной области “Библиотека”, были выделены следующие информационные объекты, которые необходимо хранить в базе данных: КНИГА, ЖАНР, ИЗДАТЕЛЬСТВО, АВТОР, ЭКЗЕМПЛЯР, ЧИТАТЕЛЬ, УЛИЦА.

Каждый из выделенных информационных объектов имеет следующие атрибуты:

- ЖАНР – название жанра;
- ИЗДАТЕЛЬСТВО – название издательства;
- АВТОР – фамилия, имя, отчество автора;

ЧИТАТЕЛЬ – фамилия, имя, отчество, телефон, адрес;

УЛИЦА – название улицы;

КНИГА – название книги, год издания;

ЭКЗЕМПЛЯР – шифр экземпляра, его наличие в библиотеке.

На рис. 1 приведена концептуальная модель предметной области “Библиотека” в виде ER-диаграммы.

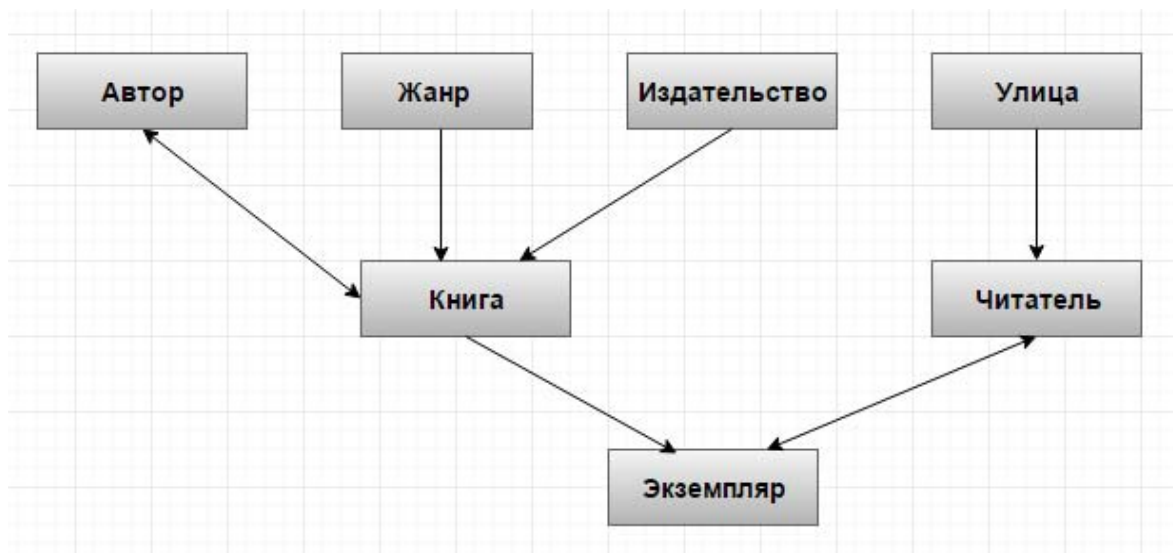
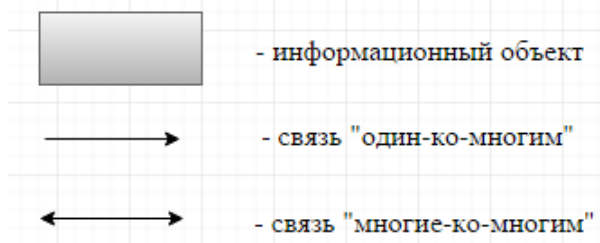


Рис. 1. Концептуальная схема базы данных “Библиотека”

Для этой диаграммы приняты следующие обозначения:



2.2.2. Логическое проектирование

Логическое проектирование базы данных – это описание базы данных в терминах принятой логической модели данных, в нашем случае – реляционной. Результат – схема реляционной базы данных, в которой информационные объекты и связи, полученные на этапе концептуального проектирования, представляются в виде, допустимом для реализации средствами реляционной модели. Полученные информационные объекты должны отвечать правилам нормализации.

На рис. 2 приведена логическая модель базы данных “Библиотека”. На этой диаграмме жирным шрифтом выделены ключи информационных объектов.

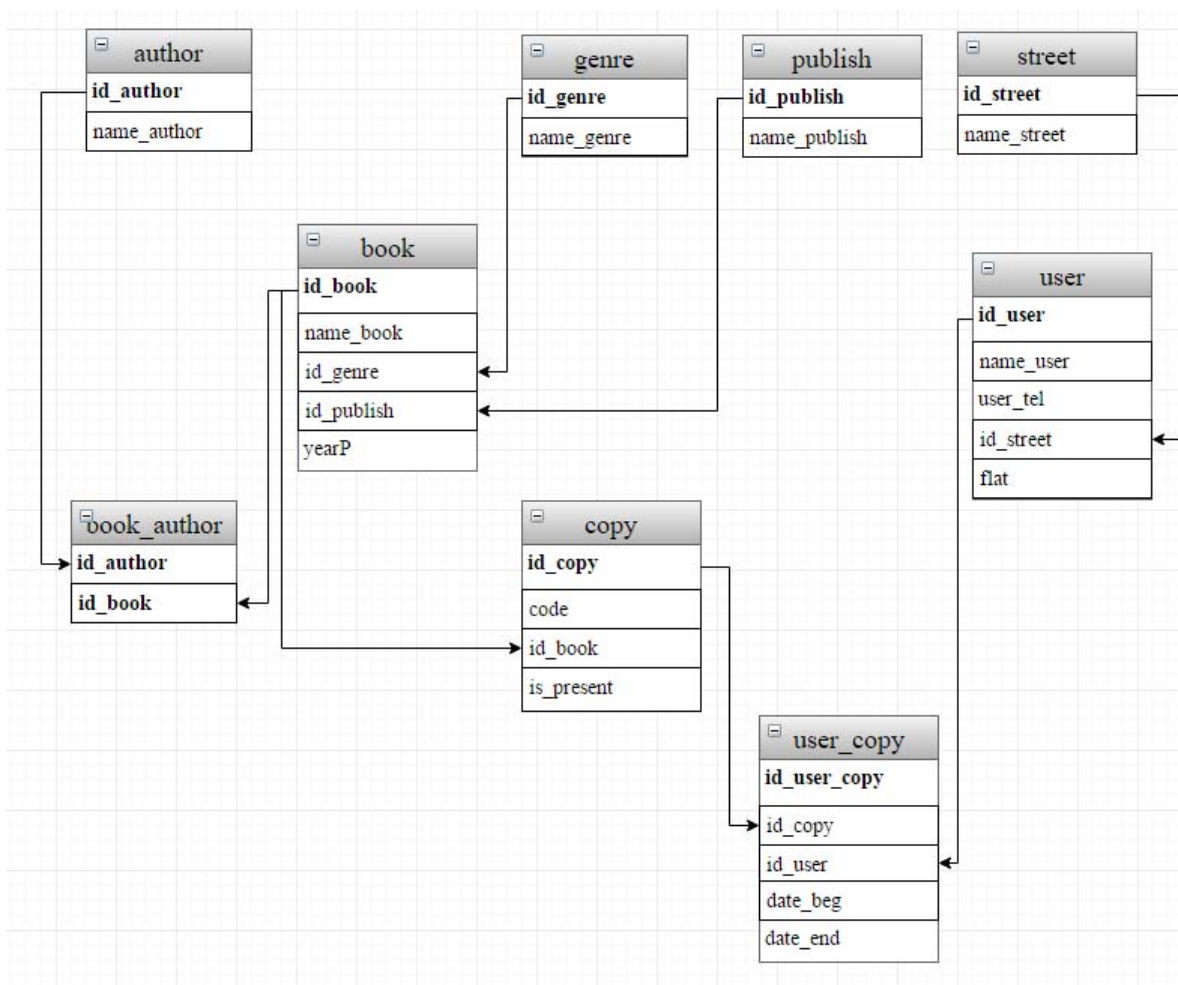


Рис. 2. Логическая схема базы данных “Библиотека”

2.2.3. Описание информационных объектов базы данных

В данном разделе приводится подробное описание каждого информационного объекта: атрибутов и их типов, а также указываются ключевые атрибуты.

В результате проектирования базы данных были получены информационные объекты, структура и характеристики которых приведены в табл. 1-9.

Таблица 1

Список жанров (genre)

Название поля	Тип данных	Свойства поля	Ключ
id_genre	Счетчик		<input checked="" type="checkbox"/>
name_genre	Текстовый	Размер поля: 50 Подпись: Жанр	

Таблица 2

Список авторов (author)

Название поля	Тип данных	Свойства поля	Ключ
id_author	Счетчик		<input checked="" type="checkbox"/>
name_author	Текстовый	Размер поля: 50 Подпись: Автор	

Таблица 3

Список издательств (publish)

Название поля	Тип данных	Свойства поля	Ключ
id_publish	Счетчик		<input checked="" type="checkbox"/>
name_publish	Текстовый	Размер поля: 50 Подпись: Издательство	

Таблица 4

Список улиц (street)

Название поля	Тип данных	Свойства поля	Ключ
id_street	Счетчик		<input checked="" type="checkbox"/>
name_street	Текстовый	Размер поля: 50 Подпись: Улица	

Таблица 5

Список книг (book)

Название поля	Тип данных	Свойства поля / Подстановка	Ключ
id_book	Счетчик		<input checked="" type="checkbox"/>
name_book	Текстовый	Размер поля: 50 Подпись: Книга	
id_genre	Числовой	Тип элемента управления: поле со списком Источник строк: genre Присоединенный столбец: 1	

		Число столбцов: 2 Ширина столбцов: 0 см; 4 см	
id_publish	Числовой	Тип элемента управления: поле со списком Источник строк: publish Присоединенный столбец: 1 Число столбцов: 2 Ширина столбцов: 0 см; 4 см	
yearP	Числовой	Подпись: Год издания	

Таблица 6

Список книг и их авторов (book_author)

Название поля	Тип данных	Свойства поля / Подстановка	Ключ
id_book	Числовой	Тип элемента управления: поле со списком Источник строк: book Присоединенный столбец: 1 Число столбцов: 2 Ширина столбцов: 0см; 4см	<input checked="" type="checkbox"/>
id_author	Числовой	Тип элемента управления: поле со списком Источник строк: author Присоединенный столбец: 1 Число столбцов: 2 Ширина столбцов: 0 см; 4 см	<input checked="" type="checkbox"/>

Таблица 7

Список экземпляров книг (copy)

Название поля	Тип данных	Свойства поля / Подстановка	Ключ
id_copy	Счетчик		<input checked="" type="checkbox"/>
code	Текстовый	Размер поля: 15 Подпись: Экземпляр	
id_book	Числовой	Тип элемента управления: поле со списком Источник строк: book Присоединенный столбец: 1 Число столбцов: 2 Ширина столбцов: 0 см; 4 см	
is_present	Логический	Значение по умолчанию: Да	

Таблица 8

Список читателей (user)

Название поля	Тип данных	Свойства поля / Подстановка	Ключ
id_user	Счетчик		<input checked="" type="checkbox"/>
name_user	Текстовый	Размер поля: 50 Подпись: Читатель	
user_tel	Текстовый	Размер поля: 10 Подпись: Телефон Маска ввода: ###\-\##\-\##	
id_street	Числовой	Тип элемента управления: поле со списком Источник строк: street Присоединенный столбец: 1 Число столбцов: 2 Ширина столбцов: 0 см; 4 см	
flat	Текстовый	Размер поля: 12 Подпись: Дом, квартира	

Таблица 9

Карточка читателя (user_copy)

Название поля	Тип данных	Свойства поля/ Подстановка	Ключ
id_user_copy	Счетчик		<input checked="" type="checkbox"/>
id_user	Числовой	Тип элемента управления: поле со списком Источник строк: user Присоединенный столбец: 1 Число столбцов: 2 Ширина столбцов: 0см; 4см	
id_copy	Текстовый	Тип элемента управления: поле со списком Источник строк: q_lookUp_user_copy Присоединенный столбец: 1 Число столбцов: 2 Ширина столбцов: 0 см; 4 см	
date_beg	Дата/время	Подпись: Дата выдачи	
date_end	Дата/время	Подпись: Дата возврата	

Примечание. Запрос *q_lookUp_user_copy* используется для реализации сложной подстановки на основе нескольких таблиц. С его помощью в поле *id_copy* одновременно будет показано и название книги, и шифр экземпляра. Синтаксис запроса на языке SQL:

```
SELECT id_copy, [name_book] & " (экземпляр " & [code] & ") "
FROM book INNER JOIN copy ON book.id_book = copy.id_book;
```

2.2.4. Схема данных

В данном разделе приводится информационно-логическая схема базы. Информационно-логическую модель желательно изображать в каноническом виде, то есть размещать информационные объекты по уровням. На нулевом уровне размещаются объекты, не подчиненные никаким другим объектам. Уровень остальных объектов определяется наиболее длинным путем к объекту от нулевого уровня. Такое размещение информационных объектов дает представление об их иерархической подчиненности, делает модель более наглядной и облегчает понимание связей между объектами.

На рис. 3 приведен снимок экрана схемы данных для базы данных “Библиотека”, реализованная в СУБД Access.

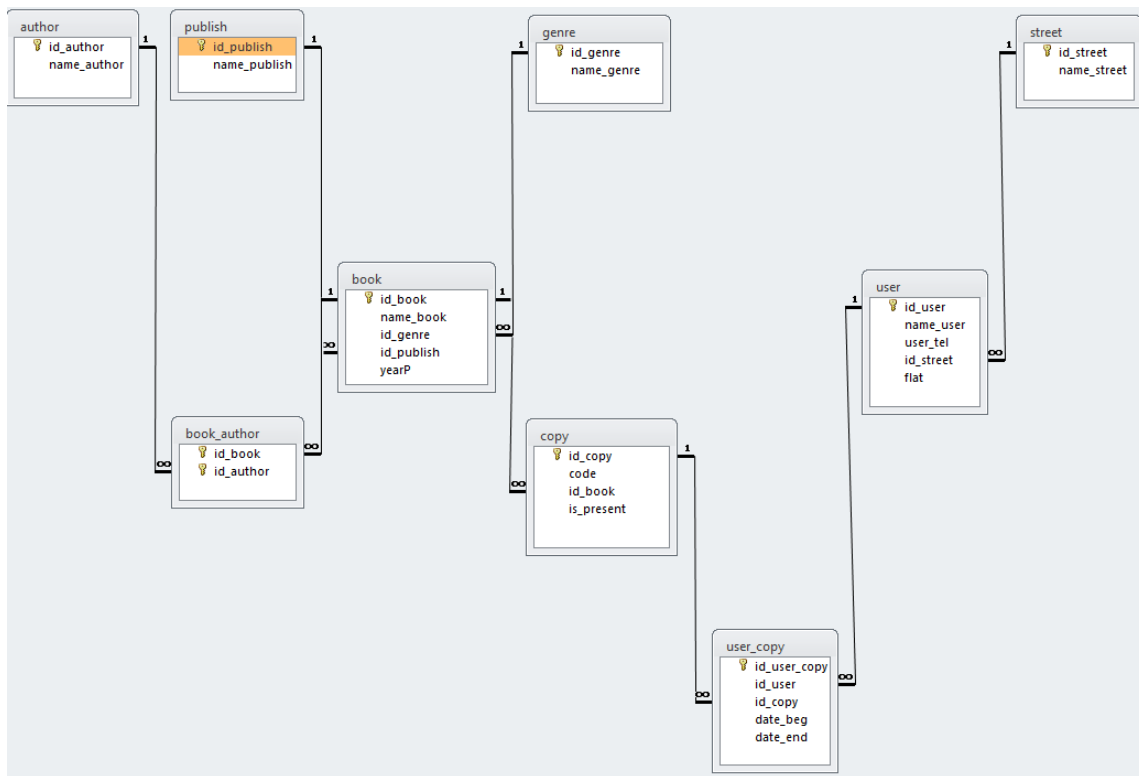


Рис. 3. Схема данных

2.3. Проектирование и реализация алгоритмов обработки данных

2.3.1. Моделирование алгоритмического поведения системы

В данном разделе проводится моделирование и описание алгоритмического поведения проектируемой системы. При этом необходимо графически изобразить последовательность действий, выполняемых системой. Для этой цели, как

правило, используются блок-схемы или UML-диаграммы деятельности. Каждая такая диаграмма акцентирует внимание на последовательности выполнения определенных процедур или элементарных операций, которые в совокупности приводят к получению желаемого результата. Для описания процесса работы разрабатываемой в рамках курсовой работы системы рекомендуется использовать UML-диаграммы.

Применяемая в них нотация представляет собой граф, вершинами которого являются состояния действия или деятельности, а дугами – переходы от одного состояния действия к другому. Каждое состояние на диаграмме соответствует выполнению некоторой операции, а переход в следующее состояние происходит только после завершения выполнения текущей операции.

Нотация описания диаграмм действия включает:

1. Прямоугольник с закруглениями краями – состояние действия.
2. Тёмный прямоугольник с закругленными краями – сложное состояние, которое будет описано на отдельной диаграмме.

3. Ромб – ветвление. В этот ромб может входить только одна стрелка от того состояния действия, после выполнения которого поток управления должен быть продолжен по одной из взаимно исключающих ветвей. Принято входящую стрелку присоединять к верхней или левой вершине символа “ромб”. Выходящих стрелок может быть две или более, но для каждой из них явно указывается соответствующее “сторожевое” условие в форме булевского выражения в квадратных скобках.

4. Широкие полосы – специальный символ для разделения и слияния параллельных вычислений или потоков управления. На диаграммах деятельности такая полоса изображается отрезком горизонтальной, реже – вертикальной, линии. При этом разделение имеет один входящий переход и несколько выходящих, Слияние, наоборот, имеет несколько входящих переходов и один выходящий. С помощью этой фигуры выделяют состояния действия, которые могут выполняться параллельно.

5. Чёрный круг – начало процесса (начальное состояние).

6. Чёрный круг с обводкой – окончание процесса (конечное состояние).

Для построения диаграмм деятельности также может использоваться online средство www.draw.io.

Анализ предметной области “Библиотека” позволил выделить основные операции, которые выполняют работники библиотеки.

1. Работа с читателями, которая включает в себя занесение информации о читателях, ведение учета взятых и сданных ими книг и пр.

2. Работа с фондом библиотеки: занесение информации о книгах, ее корректировка и актуализация.

3. Сбор и обработка статистической информации, составление отчетов о деятельности библиотеки.

Все эти виды работ могут выполняться параллельно различными людьми. На рис. 4 приведена диаграмма деятельности, описывающая данный процесс.

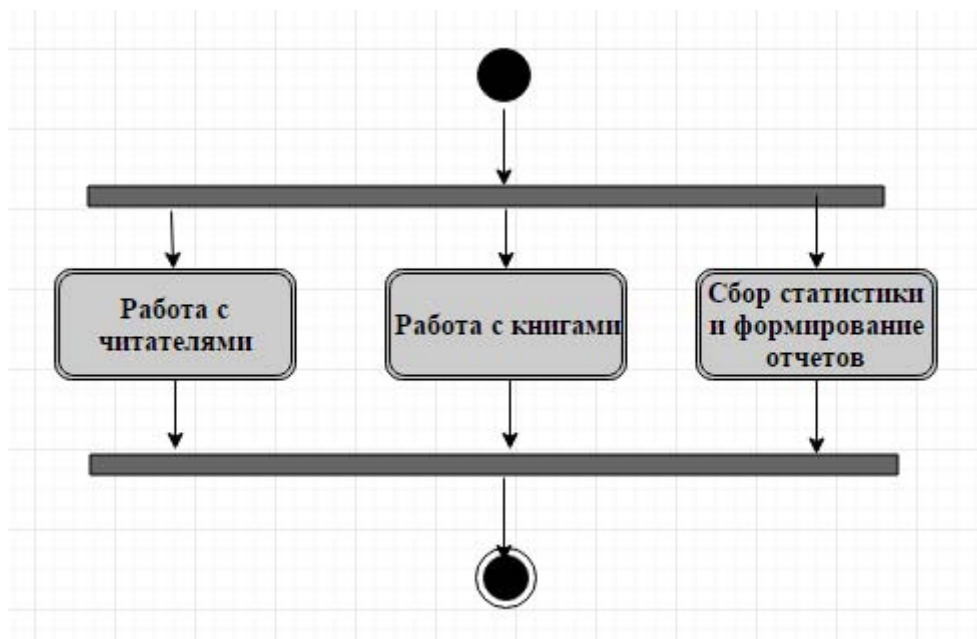


Рис. 4. Диаграмма деятельности для системы “Библиотека”

На рис. 5 приведена диаграмма деятельности для сложного состояния “Работа с читателем”. Когда читатель приходит в библиотеку в первый раз, в базу данных заносится его личная информация. Если же человек уже был в библиотеке – осуществляется поиск информации о нем в базе данных. Личную информацию о человеке можно скорректировать. Далее определяется цель посещения библиотеки: пользователь пришел взять какие-то книги или сдать те, которые брал раньше. В первом случае осуществляется подбор нужных книг по критериям, заданных пользователем. Одну или несколько выбранных книг читатель может взять из библиотеки, при этом в его карточку заносится взятая книга и дата ее выдачи. Во втором случае, когда человек принес книги, отмечается дата сдачи этой книги. Пока книга находится на руках – она не доступна для других читателей.

На рис. 6-7 приведены диаграммы деятельности для сложного состояния “Работа с книгами”. Эти диаграммы описывают последовательность действий, которые осуществляют работники библиотеки при поступлении новых книг и актуализации информации по уже имеющейся в фонде библиотеке. Книги в библиотеке можно найти по их названию, затем просмотреть информацию о них, а также при необходимости ее скорректировать, в том числе и занести новые экземпляры книг.

При поступлении новой книги (рис. 7) в базу данных заносится информация о ней, при необходимости добавляются значения новых характеристик книг (фамилии авторов, издательства, жанры). Также осуществляется присвоение библиотечных шифров каждому экземпляру книги.

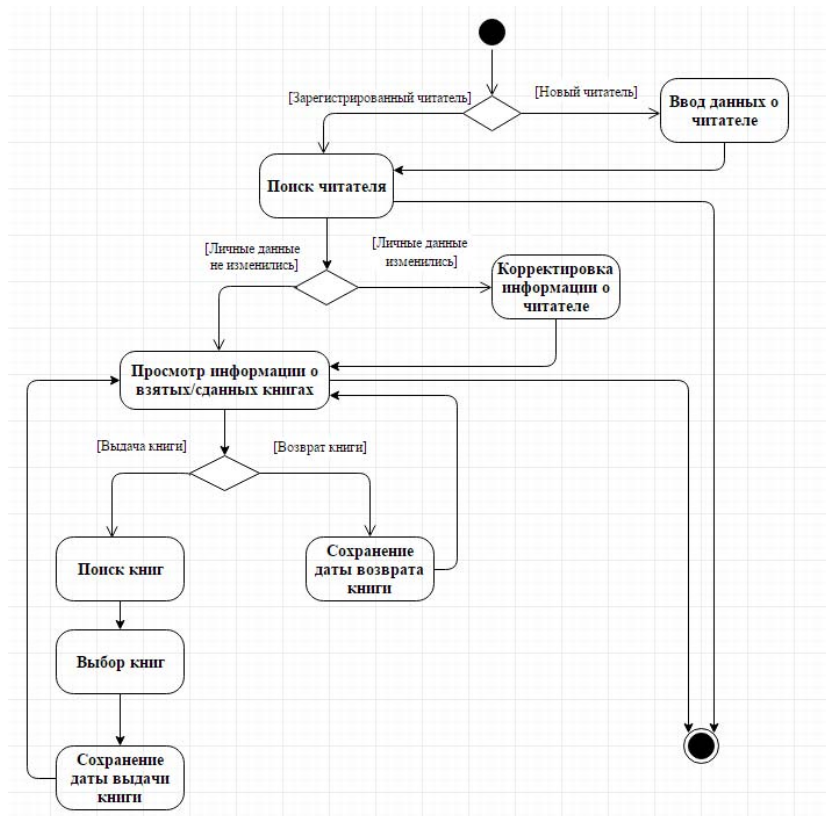


Рис. 5. Диаграмма деятельности “Работа с читателем в библиотеке”

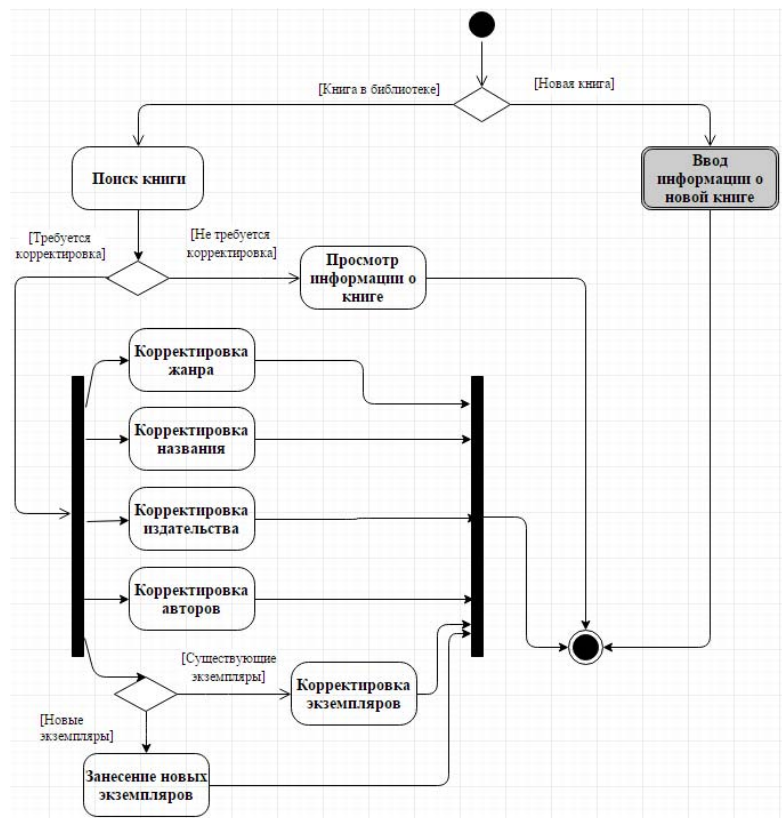


Рис. 6. Диаграмма деятельности “Работа с книгами”

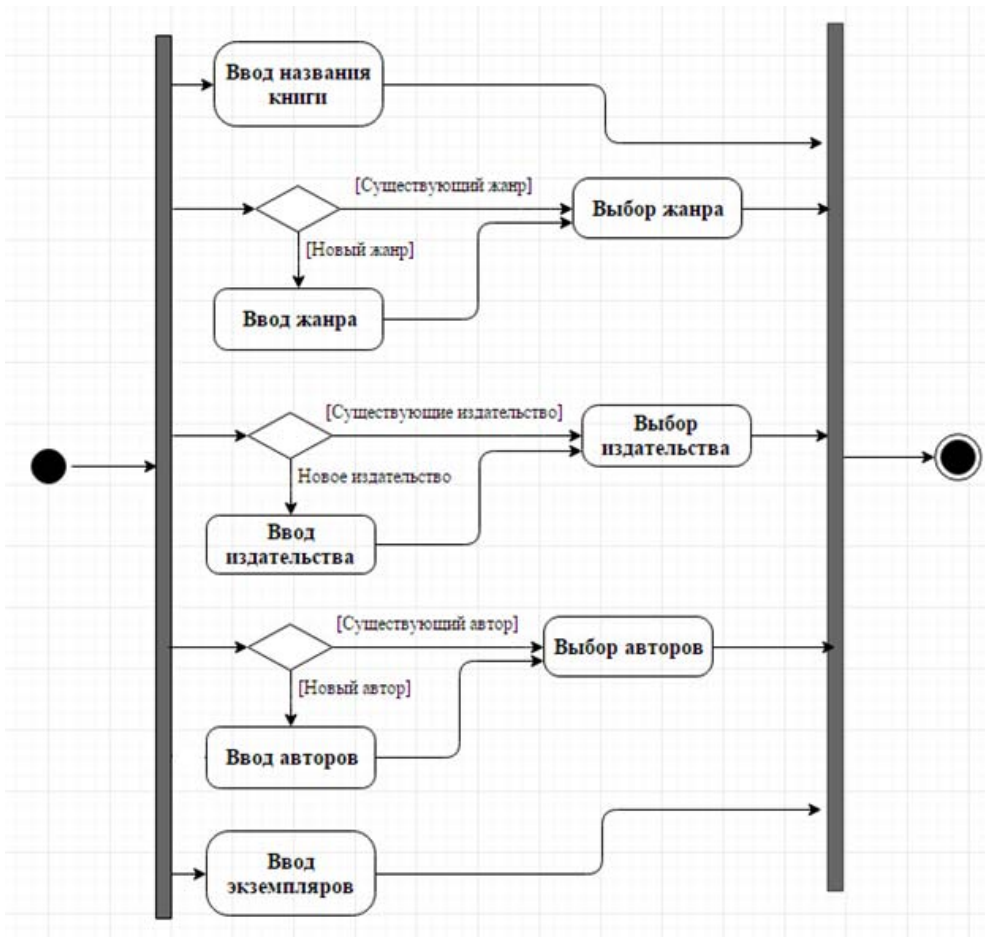


Рис. 7. Диаграмма деятельности “Ввод информации о новой книге”

На рис. 8 приведена диаграмма, описывающая процесс сбора статистики и формирования отчетов по работе библиотеки.

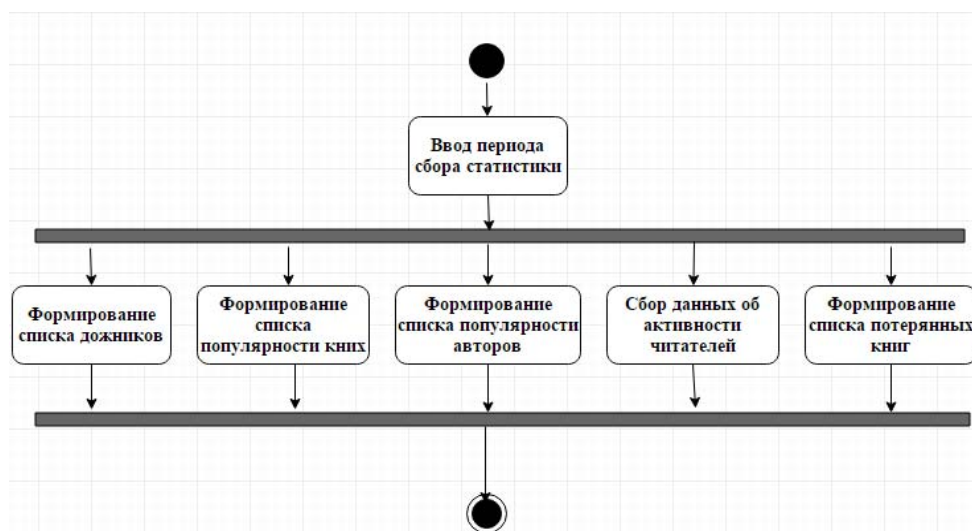


Рис. 8. Диаграмма деятельности “Сбор статистики и формирование отчетов”

2.3.2. Обработка данных средствами запросов

В конечном итоге, все алгоритмы, описанные диаграммами деятельности в предыдущем разделе, должны быть реализованы тем или иным средством СУБД. Это могут быть запросы, формы, отчеты, макросы или программы. В данном разделе описывается реализация некоторых фрагментов диаграмм деятельности средствами SQL запросов.

Для описания алгоритма, реализуемого одним запросом, достаточно его словесного описания. Также можно использовать функциональную схему, на которой указываются входные и выходные данные, описывается назначение запроса. К входным данным запроса относятся имена таблиц, используемые в запросе, а также параметры, задаваемые пользователем. К выходным данным относится таблица с результатами запроса, а также, если используются запросы на корректировку данных, таблица, сформированная или обновленная запросом. После схемы желательно привести описание запроса на языке SQL.

Более сложные задачи требуют последовательного выполнения нескольких запросов. Тогда рекомендуется описать последовательность выполнения запросов по этапам, описав каждый этап как один запрос.

Поскольку при реализации курсовой работы, средствами SQL может быть реализовано достаточно много алгоритмов, необходимо подробно описать один из них, а описание остальных привести в конце раздела.

На диаграмме деятельности, приведенной на рис. 5, показан процесс выдачи книги читателю, который может быть реализован средствами запросов корректировки данных SQL.

1 этап. Создание таблицы *find*, куда заносится информация по всем, имеющимся в наличии экземплярам книг, если у книги несколько авторов – выводится первый из них. На рис. 9 показана схема этого запроса.

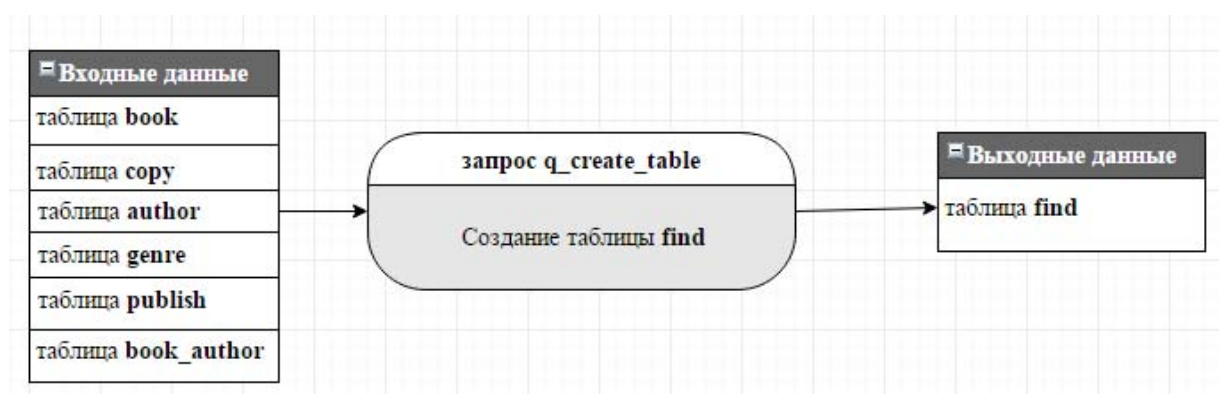


Рис. 9. Схема запроса `q_create_table`

Запрос `q_create_table` на языке SQL:

```
SELECT name_genre, name_book, First(name_author) AS  
[First-name_author], name_publish, yearP, code, Choose,  
copy.id_copy INTO find
```

```

FROM genre INNER JOIN ((publish INNER JOIN book ON
publish.[id_publish] = book.id_publish) INNER JOIN copy ON
book.id_book = copy.id_book) INNER JOIN (author INNER JOIN
book_author ON author.id_author = book_author.id_author)
ON book.id_book = book_author.id_book) ON genre.id_genre =
book.id_genre

```

```

GROUP BY name_genre, name_book, name_publish, yearP,
code, Choose, id_copy, is_present
HAVING copy.is_present=True;

```

2 этап. После того, как пользователь отметил в таблице *find* выбранные им экземпляры (*choose=true*), необходимо занести информацию об этих экземплярах и дату их выдачи в таблицу *user_copy* для определенного пользователя. На рисунке 10 приведена схема запроса.

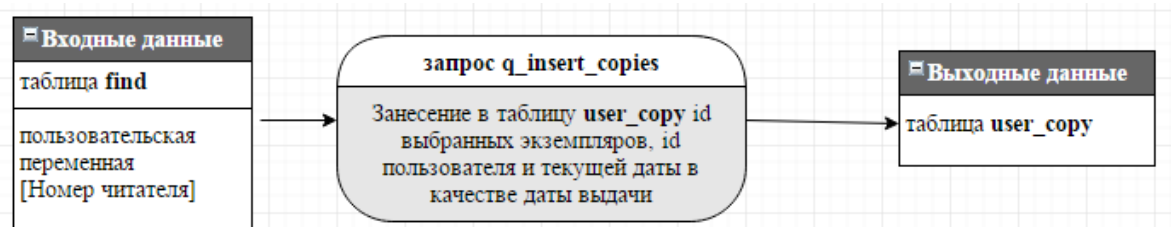


Рис. 10. Схема запроса *q_insert_copies*

Запрос *q_insert_copies* на языке SQL:

```

INSERT INTO user_copy ( id_user, id_copy, date_beg )
SELECT [Номер читателя], find.id_copy, Date()
FROM find
WHERE choose=True;

```

3 этап. Поскольку экземпляр книги уже выдан читателю, необходимо установить соответствующий признак в таблице *copy*. На рисунке 11 приведена соответствующая схема.

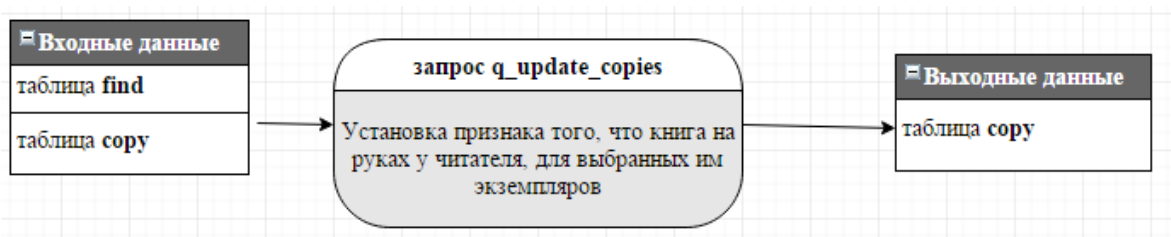


Рис. 11. Схема запроса *q_update_copies*

Запрос *q_update_copies* на языке SQL:

```

UPDATE copy INNER JOIN find ON copy.id_copy =
find.id_copy
SET copy.is_present = No

```

```
WHERE find.Choose=True;
```

Кроме описанного выше алгоритма выдачи книги, средствами запросов на языке SQL реализованы следующие алгоритмы:

- реализация сдачи экземпляра книги читателем в соответствии с диаграммой деятельности, приведенной на рис. 5 (запрос *q_edit_copy_true*);
- формирование статистической и отчетной информации в соответствии с диаграммой, приведенной на рис. 8 (запросы *q_debtor*, *q_popular_book*, *q_popular_author*, *q_activity*, *q_book_lost*).

2.4. Проектирование и реализация приложения пользователя

2.4.1. Структура приложения пользователя

В данном разделе описывается структура приложения пользователя, состоящая из совокупности форм и переходов между ними. Приложение должно быть спроектировано таким образом, чтобы реализовать все описанные в предыдущей главе алгоритмы, а также, по возможности, моделировать привычную для пользователя среду работы.

Структура приложения описывается на “верхнем” уровне, то есть без детализации по каждой форме. Например, подчиненные формы на схеме со структурой не указываются.

В соответствии с видами деятельности библиотеки, работа пользователя с приложением будет осуществляться через три основные формы: Читатели, Книги и Статистика (рис. 12). Также на данной схеме показаны остальные формы приложения и их взаимные переходы.

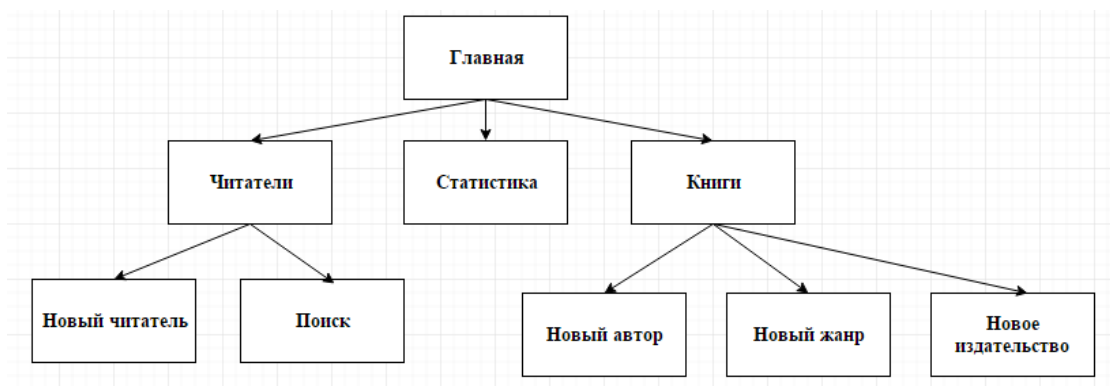


Рис. 12. Структура приложения

2.4.2. Проектирование форм

В разделе описывается проектирование одной формы, остальные просто перечисляются, с указанием их назначения.

Этапы проектирования формы:

- 1) определить подструктуру данных, на основе которой строится форма;
- 2) определить несколько вариантов построения формы, для каждого из вариантов указать:
 - тип формы (простая, составная или вложенная),
 - источник записей – таблицы или запрос, данные из которых должны отображаться в форме,
 - таблицу, на которой строится форма,
 - связанные поля для вложенных форм,
 - особенности данного варианта построения формы;
- 3) выбрать вариант реализации формы;
- 4) определить все поля базы данных, которые должны быть отображены в форме;
- 5) определить управляющие и вычисляемые поля, указать имя каждого поля, а также, при необходимости, формулу вычисления;
- 6) разработать макет формы.

Одной из форм разрабатываемого приложения является форма “Книги”, предназначенная для работы с книжным фондом библиотеки. С помощью этой формы можно добавить и скорректировать информацию о книге, а также создать и скорректировать ее экземпляры.

Проектирование формы “Книги”

1. Реализация формы осуществляется на основе следующего фрагмента предметной области (рис. 13).

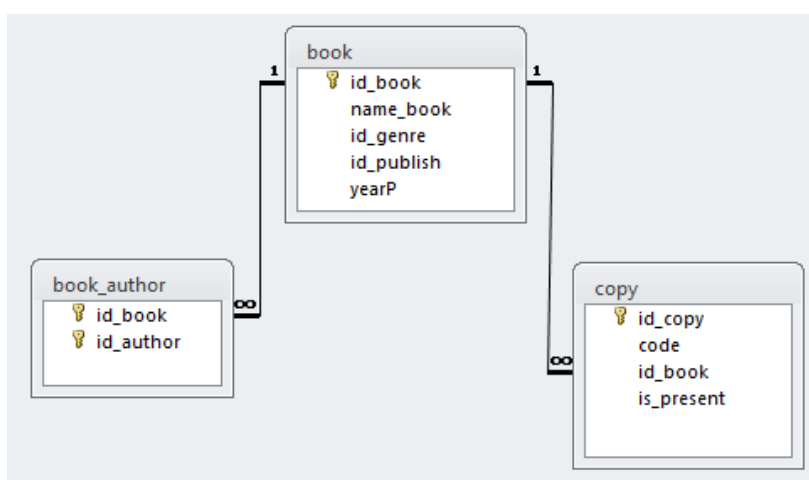


Рис. 13. Подструктура данных формы “Книги”

2. На основе данного фрагмента предметной области можно построить следующие варианты форм:

2.1. Форма с одной вложенной, следующей структуры:

Главная форма

Тип формы: составная

Источник записи: *book, book_author*

Таблица, на которой строится форма: *book_author*

Вложенная форма

Тип формы: простая

Источник записи: *сору*

Таблица, на которой строится форма: *сору*

Связанные поля

Главная форма: *book.id_book*

Подчиненная форма: *сору.id_book*

Особенности формы:

- с помощью этой формы нельзя занести новую книгу;
- форма поддерживает занесение новых экземпляров книги;
- поддерживается иерархический просмотр экземпляров;
- в главной форме дублируются названия книг, если книга имеет нескольких авторов;

– с помощью этой формы нельзя ввести нового автора, жанр и издательства.

2.2. Форма с двумя вложенными формами:

Главная форма

Тип формы: простая

Источник записи: *book*

Таблица, на которой строится форма: *book*

Вложенная форма 1

Тип формы: простая

Источник записи: *сору*

Таблица, на которой строится форма: *сору*

Связанные поля

Главная форма: *book.id_book*

Подчиненная форма: *сору.id_book*

Вложенная форма 2

Тип формы: простая

Источник записи: *book_author*

Таблица, на которой строится форма: *book_author*

Связанные поля

Главная форма: *book.id_book*

Подчиненная форма: *book_author.id_book*

Особенности формы:

- поддерживает ввод новых книг;

- позволяет вводить новые экземпляры;
- осуществляется иерархический просмотр экземпляров и авторов;
- с помощью этой формы нельзя ввести нового автора, жанр и издательство.

3. Поскольку данная форма предназначена для корректировки и занесения книг и их экземпляров, реализация формы будет осуществляться на основе второго варианта построения. При этом данная форма будет поддерживать ввод новых авторов, издательств и жанров. Эти действия необходимо реализовать отдельно.

4. В разрабатываемую форму должны быть включены следующие элементы:

Главная форма

- поле *book.name_book* (подпись: Название);
- поле со списком *book.id_genre* (подпись: Жанр);
- поле *book.yearP* (подпись: Год издания);

Подчиненная форма 1 (подпись: Авторы)

- поле со списком *book_author.id_author* (Подпись: Фамилия И.О.);

Подчиненная форма 2 (подпись: Экземпляры)

- поле *copy.code* (подпись: Шифр);
- поле *copy.is_present* (подпись: Наличие).

5. В заголовок формы необходимо включить поле со списком для реализации поиска уже существующей книги *NBook*, со следующими свойствами этого поля:

Подпись: Поиск книги

Источник: book

Присоединенный столбец: 2

Число столбцов: 2

Ширина столбцов: 0 см; 3 см

6. Макет формы приведен на рис. 14.

Для реализации приложения “Библиотека” разработаны следующие формы:

f_book – с помощью этой формы осуществляется просмотр, корректировка и занесение новых книг и экземпляров (ее проектирование описано выше);

f_reader – используется для просмотра и корректировки личных данных пользователя и информации о взятых и сданных книгах;

f_stat – используется для формирования и просмотра различных статистических показателей работы библиотеки;

f_search – с помощью этой формы осуществляется поиск книг и выбор необходимых пользователю экземпляров;

f_new_reader, f_new_genre, f_new_publish, f_new_author – используются для занесения новых пользователей, жанров, издательств и авторов.

Поиск книги

Название

Жанр

Издательство

Год издания

Авторы

Фамилия И.О.

*

Экземпляры

Шифр	Наличие
s12d123	<input checked="" type="checkbox"/>
s12d124	<input checked="" type="checkbox"/>
f09p456	<input checked="" type="checkbox"/>
*	<input checked="" type="checkbox"/>

Запись: 1 из 10

Рис. 14. Макет формы

2.4.3. Проектирование отчетов

В данном разделе описывается проектирование одного отчета, остальные просто перечисляются с указанием, для чего они используются.

Этапы проектирования отчета:

- 1) определить подструктуру данных, на основе которой строится отчет;
- 2) определить несколько вариантов построения отчета, для каждого из вариантов указать:
 - тип отчета (простой, составной или вложенный),
 - источник записей – таблицы или запрос, данные из которых должны отображаться в отчете,
 - таблицу, на которой строится отчет,
 - уровни группировки отчета, при необходимости,
 - связанные поля для вложенных отчетов,
 - особенности данного варианта построения отчета;
- 3) выбрать вариант реализации отчета;
- 4) определить все поля базы данных, которые должны быть отображены в отчете;
- 5) определить управляющие и вычисляемые поля, при необходимости указать имя каждого поля и формулу вычисления, указать в каком разделе они размещаются;
- 6) разработать макет отчета.

Для формирования списка читателей, которые имеют на руках книги, используется отчет “Должники”, в котором выводятся личные данные читателей, а также несданные ими книги.

Проектирование отчета “Должники”

1. Реализация формы осуществляется на основе следующего фрагмента предметной области (рис. 15).

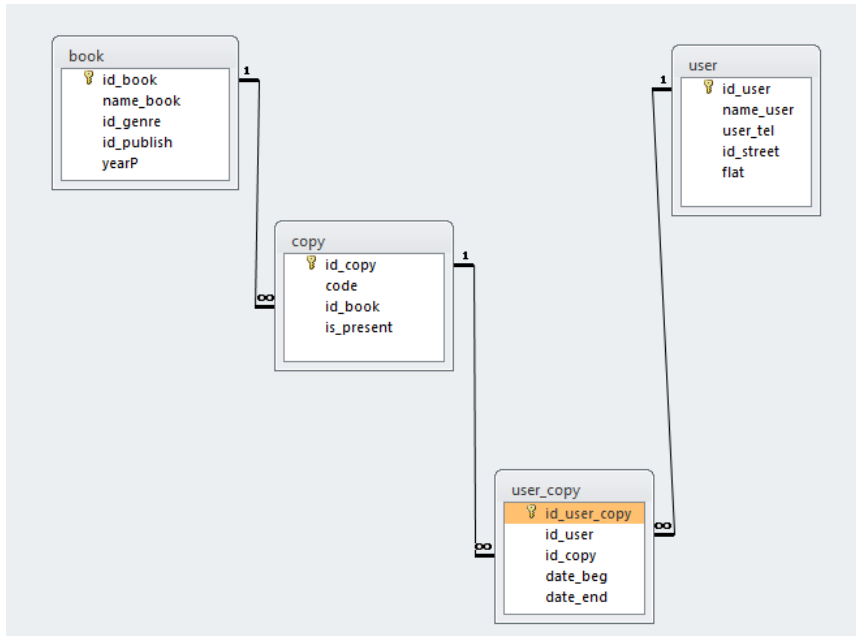


Рис. 15. Подсхема данных отчета “Должники”

2. На основе данного фрагмента предметной области можно построить следующие варианты отчетов:

2.1. Составной отчет, следующей структуры:

Тип отчета: составной

Источник записи: *запрос*

```
SELECT name_user, user_tel, name_book, code, date_beg  
FROM (book INNER JOIN copy ON book.id_book =  
copy.id_book) INNER JOIN (user INNER JOIN user_copy ON  
user.id_user = user_copy.id_user) ON copy.id_copy =  
user_copy.id_copy  
WHERE date_end Is Null;
```

Таблица, на которой строится отчет: *user_copy*

Уровни группировки: 1 уровень: по полю *name_user*

Вложенный отчет:

Главный отчет

Тип отчета: простой

Источник записи: *user*

Таблица, на которой строится отчет: *user*

Вложенный отчет

Тип отчета: составной

Источник записи: *запрос*

```

SELECT name_book, code, date_beg
FROM (book INNER JOIN copy ON book.id_book =
copy.id_book) INNER JOIN user_copy ON copy.id_copy =
user_copy.id_copy
WHERE user_copy.date_end Is Null;

```

Таблица, на которой строится отчет: *user_copy*

Связанные поля

Главная форма: *user.id_user*

Подчиненная форма: *user_copy.id_user*

Особенности отчета:

– в данном отчете будут отображаться все читатели, даже если у них нет несданных книг.

3. Поскольку в отчет необходимо вывести только должников, для реализации выбираем первый вариант.

4. В разрабатываемый отчет должны быть включены следующие элементы:

Заголовок группы:

– поле *user.name_user* (подпись: Фамилия И.О.);

– поле *user.user_tel* (подпись Телефон)

Область данных:

– поле *book.name_book* (подпись: Название книги);

– поле *copy.code* (подпись: Шифр);

– поле *data_beg* (подпись: Дата выдачи).

5. В данном отчете используются следующие вычисляемые поля:

Заголовок отчета

– поле для текущей даты.

Данные: = Now()

Подпись: Список должников

Область данных

– поле для нумерации взятых экземпляров книг.

Данные: = +1

Подпись: № пп

Сумма с накоплением: для группы

Примечание группы (по name_group)

– поле для вычисления общего количества книг, которые на руках у читателя.

Данные: = count([code])

Подпись: Всего книг на руках

6. Макет отчета приведен на рис. 16

Список должников на 7 апреля 2016 г.				
Читатель Федоров		Телефон 367-15-12		
№ пп	Название книги	Шифр	Дата выдачи	Количество дней на руках
1	Золотой теленок	f09p001	29 марта 2016 г.	9
2	Пуаро ведет следствие	ц4556	7 апреля 2016 г.	0
Всего книг на руках				2
Читатель Туполев		Телефон 235-23-47		
№ пп	Название книги	Шифр	Дата выдачи	Количество дней на руках
1	Таинственный остров	к56к5767	29 марта 2016 г.	9
2	Двенадцать стульев	s12d124	29 марта 2016 г.	9
3	Двенадцать стульев	f09p456	29 марта 2016 г.	9
Всего книг на руках				3

Рис. 16. Макет отчета

Приложение “Библиотека” позволяет создавать и печатать следующие отчеты:

r_debtors – для печати списка должников библиотеки (проектирование этого отчета описано выше);

r_top_book – для печати списка книг по их популярности среди читателей в заданный период времени;

r_top_author – для печати списка авторов по их популярности среди читателей в заданный период времени;

r_user_activity – отчет о ежедневной работе библиотеки в заданный период времени.

2.4.4. Разработка средств управления и навигации

В данном разделе описываются способы организации работы пользователя с приложением, а также реализация основных операций, выполняемых системой.

Подробно описывается наиболее активно используемая последовательность действий (в соответствии с диаграммами деятельности) и способы навигации по формам.

В соответствии с диаграммой деятельности, приведённой на рис. 4, работа с читателями, работа с книгами и сбор и обработка информации могут выполняться независимо друг от друга. Каждый вид деятельности будет реализован в отдельной форме, а для переключения между ними используется механизм “вкладок” (рис. 17).

Читатели			Книги	Статистика
Читатель	Федоров	▼	Новый	
адрес	Алеутская	▼	12-34	
телефон	3671512		Поиск книг	
Название книги	Выдана	Сдана		
Золотой теленок (экземпляр f09p001)	29.03.2016		Сдать	
Золотой теленок (экземпляр k34p705)	29.03.2016	29.03.2016	Сдать	
Таинственный остров (экземпляр k09p890)	29.03.2016	07.04.2016	Сдать	
Пуаро ведет следствие (экземпляр ц4556)	07.04.2016		Сдать	
			Сдать	

Рис. 17. Главная форма приложения *main*

На рис. 17 показана форма *f_reader* для работы с читателями в соответствии с диаграммой деятельности, приведенной на рис. 5. В этой форме пользователь может выполнить следующие действия.

1. *Поиск читателя.* Поиск осуществляется по шаблону, выбранному пользователем из поля со списком *Читатель*. Для его реализации:

– в заголовок формы Читатели включено поле со списком со следующими свойствами:

Имя: *NUser*

Подпись: Читатель

Источник: user

Присоединенный столбец: 2

Число столбцов: 2

Ширина столбцов: 0 см; 3 см

– в области данных размещено поле *user.id_user*, скрытое под подчиненной формой;

– с событием *После Обновления* поля со списком *NUser* связан макрос *find_user*:

КЭлементуУправления ([id_user])

НайтиЗапись (=[NUser])

КЭлементуУправления ([NUser])

2. *Занесение нового читателя.* Выполняется через дополнительную форму *f_new_reader* (“Новый читатель”), вызов которой осуществляется по щелчку по кнопке *Новый* (рис. 18).

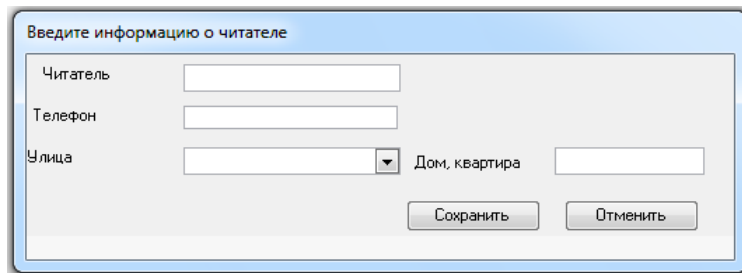


Рис. 18. Форма для занесения нового читателя

Реализация операции “Занесение нового читателя” включает:

– вызов макроса для открытия формы “Новый читатель” в режиме добавления по событию *Нажатие Кнопки* для кнопки главной формы *Новый*:

Открыть форму (*f_new_reader*, , , *Добавление*)

– вызов процедуры по событию *Нажатие Кнопки* для кнопки *Сохранить* формы *f_new_user*, которая закрывает текущую форму, обновляет главную и настраивает поле поиска *NUser* в главной форме:

```
Private Sub ButtonSave_Click()
    'запоминаем id нового читателя
    n = Me.id_user
    'закрываем форму Новый
    DoCmd.Close acForm, "f_new_reader1"
    'обновляем форму "Читатели"
    Forms![main]![f_reader1]![NUser].Requery
    'устанавливаем id нового пользователя в поле поиска
    Forms![main]![f_reader1]![NUser] = n
    'обновляем форму Читатели
    Forms![main]![f_reader1].Requery
    'устанавливаем на форму фокус
    Forms![main]![f_reader1]![NUser].SetFocus
    'запускаем макрос поиска
    DoCmd.RunMacro "find_user"
End Sub
```

– вызов процедуры по событию *Нажатие Кнопки* для кнопки *Отменить* формы *f_new_user* для закрытия текущей формы без сохранения данных:

```
Private Sub ButtonCancel_Click()
    DoCmd.Close acForm, "f_new_reader", acSaveNo
End Sub
```

3. *Поиск*. Выполняется через дополнительную форму *f_search* (“Поиск”), вызов которой осуществляется по щелчку по кнопке *Поиск книг* (рис. 19). В этой форме осуществляется поиск книг по различным критериям и выбор нужных пользователю экземпляров книг.

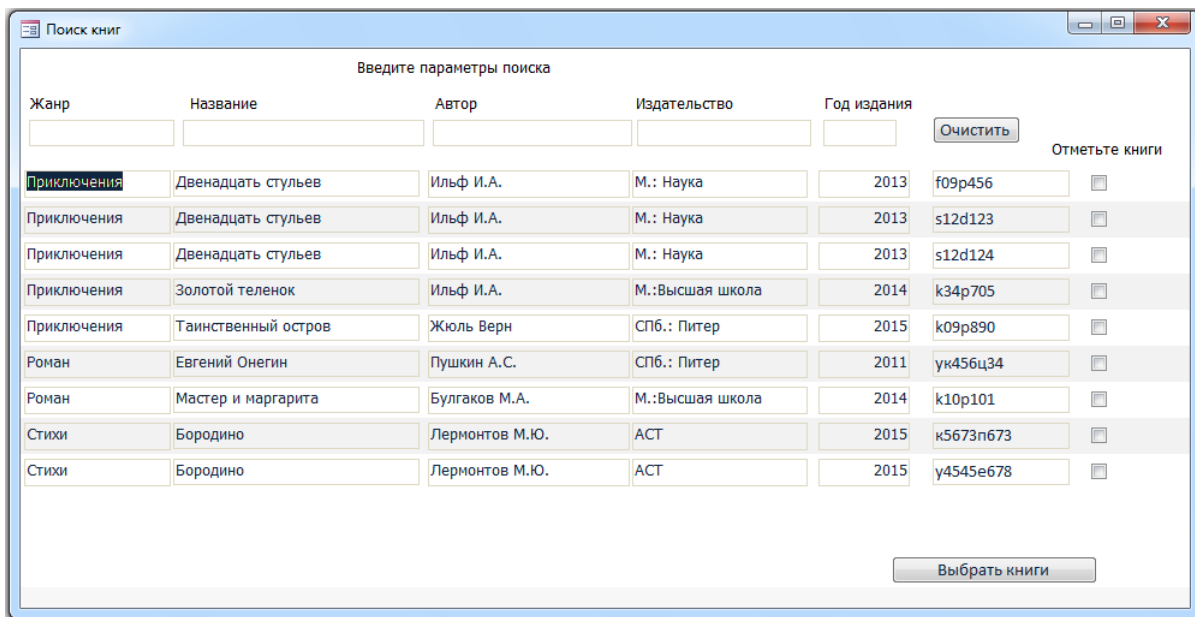


Рис. 19. Форма поиска и выбора книг

Реализация поиска и выбора книг включает:

– вызов процедуры по событию *Нажатие Кнопки* для кнопки *Поиск книг* формы *f_reader*, которая открывает запрос на создание таблицы *find* и запускает форму поиска *f_search*:

```
Private Sub ButtonSearch_Click()
    'предварительное удаление таблицы
    DoCmd.DeleteObject acTable, "find"
    'создание таблицы
    DoCmd.OpenQuery "Create_table"
    'открытие формы
    DoCmd.OpenForm "f_search"
End Sub
```

– поиск книг через расширенный фильтр, который реализуется следующим образом:

а) создание запроса *q_filtr_book*:

```
SELECT * FROM find
WHERE (find.name_genre) Like [Формы]![f_search]![f1] &
"*) AND (find.name_book Like [Формы]![f_search]![f2] &
"*) AND (find.[name_author] Like [Формы]![f_search]![f3] &
"*) AND (find.name_publish Like [Формы]![f_search]![f4] &
"*) AND (find.yearP Like [Формы]![f_search]![f5] &
"*) ;
```

б) вызов макроса *m_filtr* по событию *После обновления* полей *f1, f2, f3, f4, f5*, используемых для задания образца поиска для *Жанра, Названия, Автора, Издательства* и *Года издания*:

```
Применить фильтр (q_filtr_book)
```

в) вызов процедуры по событию *Нажатие Кнопки* для кнопки *Очистить* формы *f_search*, которая очищает поля поиска:

```
Private Sub buttonClear_Click()  
    [f1] = ""  
    [f2] = ""  
    [f3] = ""  
    [f4] = ""  
    [f5] = ""  
    DoCmd.ApplyFilter ("filtr_book")  
End Sub
```

– вызов процедуры по событию *Нажатие Кнопки* для кнопки *Выбрать книги* формы *f_search*, которая реализует выбор отмеченных читателем книг (осуществляется в соответствии с алгоритмом, описанным ранее – заносится соответствующая запись в таблицу *user_copy*, и данный экземпляр помечается как отсутствующий в таблице *copy*):

```
Private Sub buttonChoose_Click()  
    'обновление текущей формы  
    Me.Requery  
    'запуск соответствующих запросов  
    DoCmd.OpenQuery "q_add_r_b"  
    DoCmd.OpenQuery "q_edit_copy"  
    'обновление подчиненной формы с экземплярами  
    Forms![main]![f_reader]![fp_user_copy].Requery  
    'закрытие формы поиска  
    DoCmd.Close acForm, "f_search"  
End Sub
```

Сдача книг читателем. Выполняется по щелчку по кнопке *Сдать* (рис. 16), в результате чего в поле *Дата сдачи* таблица *user_copy* заносится текущая дата, а также обновляется статус данного экземпляра книги в таблице *copy*. Соответствующая процедура представлена ниже:

```
Private Sub buttonPas_Click()  
    Me.date_end = Date  
    DoCmd.OpenQuery "q_edit_copy_true"  
End Sub
```

Остальные виды деятельности: работа с книгами и статистика, – реализованы в виде отдельных форм, размещенных на главной форме, в соответствии с диаграммами деятельности, приведенными на рис. 6-8. В них используются аналогичные приемы навигации и способы управления.

2.5. Краткое руководство пользователя

В разделе описывается работа пользователя с приложением для тех видов деятельности, которые не рассмотрены в предыдущем разделе.

На рис. 20 показана главная форма приложения “Библиотека”.

Название книги	Выдана	Сдана	Сдать
Золотой теленок (экземпляр f09p001)	29.03.2016		Сдать
Золотой теленок (экземпляр k34p705)	29.03.2016	29.03.2016	Сдать
Таинственный остров (экземпляр k09p890)	29.03.2016	07.04.2016	Сдать
Пуаро ведет следствие (экземпляр ц4556)	07.04.2016		Сдать
			Сдать

Рис. 20. Главная форма приложения “Библиотека”

Через вкладку “Читатели” пользователь может найти уже существующего читателя, скорректировать его личную информацию, занести нового читателя, найти нужную книгу, оформить ее и сдать экземпляры книг, которые были у него на руках (подробно описано в предыдущем разделе).

На рис. 21 приведена форма для работы с фондом книг в библиотеке. Через данную форму пользователь может найти уже существующую книгу (поле со списком *Поиск Книги*), скорректировать информацию о ней, внести новые экземпляры книги.

Новую книгу можно занести в эту форму, щелкнув по кнопке *Новая книга* (рис. 22). В результате в форме появятся кнопки для вызова форм, через которые можно задать новый жанр, издательство и автора, а также пустые поля для занесения характеристик новой книги. Сохранение книги осуществляется щелчком по кнопке *Сохранить*.

По щелчку по кнопкам *Новая книга*, *Новый жанр*, *Новое издательство* будут вызваны формы, приведенные на рис. 23. Работа с ними осуществляется как с формой *Новый читатель*, описанной в предыдущем разделе.

Для просмотра и анализа деятельности библиотеки используется форма *f_stat*, расположенная на вкладке “*Статистика*” (рис. 24)

Читатели	Книги	Статистика
Поиск книги	Мастер и маргарита	Новая книга
Название	Мастер и маргарита	
Жанр	Роман	
Издательство	М.:Высшая школа	
Год издания	2014	
Авторы	Фамилия И.О. Булгаков М.А. *	
Экземпляры	Шифр k10p101 *	Наличие <input checked="" type="checkbox"/> <input checked="" type="checkbox"/>

Рис. 21. Форма для работы с фондом библиотеки

Читатели	Книги	Статистика
Поиск книги		Новая книга
Название		
Жанр		Новый жанр
Издательство		Новое издательство
Год издания		
Авторы	Фамилия И.О. *	Новый автор
Экземпляры	Шифр *	Наличие <input checked="" type="checkbox"/>
		Сохранить

Рис. 22. Форма для занесения новой книги

The image shows three separate forms for data entry:

- Введите новый жанр**: A form with a text input field labeled "Жанр" and two buttons: "Сохранить" (Save) and "Отменить" (Cancel).
- Введите нового автора**: A form with a text input field labeled "Автор" and two buttons: "Сохранить" (Save) and "Отменить" (Cancel).
- Введите новое издательство**: A form with a text input field labeled "Издательство:" and two buttons: "Сохранить" (Save) and "Отменить" (Cancel).

Рис. 23. Формы для занесения нового жанра, издательства, автора

The screenshot shows a web interface for library activity analysis. It includes a navigation menu with "Читатели", "Книги", and "Статистика". The "Статистика" section is active, showing a "Запрос" (Query) dropdown menu with options like "Популярность книг", "Список должников", and "Популярность авторов". The "Период" (Period) is set from "01-мар-2016" to "12-апр-2016". A "Напечатать" (Print) button is visible. Below the form is a table with the following data:

Запрос	Count-id_cc
Двенадцать	3
Таинственный	2
Золотой теленок	2
Пуаро ведет следствие	1

Рис. 24. Форма для анализа деятельности библиотеки

Все запросы, которые можно выполнить, заданы в виде поле со списком *Запрос*. В поля *Период* вводится интервал времени, для которого выполняются запросы. При необходимости, выбранные данные можно напечатать в виде соответствующих отчетов, щелкнув по кнопке *Напечатать*.

2.6. Заключение

В заключении описываются основные результаты курсовой работы. Указываются количественные характеристики разработанного приложения. Данный раздел в отчете не нумеруется.

В рамках курсовой работы выполнено проектирование базы данных для предметной области “Библиотека”, разработано соответствующее приложение. Реализация выполнена в СУБД ACCESS и включает:

- 9 таблиц;
- 12 запросов, из них 4 запроса корректировки данных;
- 13 форм, включая подчиненные;

- 7 макросов;
- 15 процедур обработки событий (используемый язык программирования – Microsoft Visual Basic for Applications).

В отчете показаны основные этапы проектирования базы данных и приложения пользователя.

2.7. Литература

В данном разделе указывается список используемой литературы, включающий как минимум список основной литературы по дисциплине “Базы данных”. Также перечисляются другие используемые литературные или интернет-источники. Литература перечисляется подряд, без деления на подразделы. Раздел в отчете не нумеруется. Список основной литературы по дисциплине приведен в соответствующем разделе данного методического пособия.

2.8. Приложение

В приложении приводятся отчеты, формы, макросы и процедуры, не вошедшие в основной текст. Отчеты и формы отображаются в виде снимков экрана, макросы и процедуры в виде текстов программ. Для каждого элемента выполняется описание его назначения. В отчете нумерация данного раздела отсутствует.

Критерии оценки

Оценка за курсовую работу формируется в соответствии со следующими критериями:

- степень самостоятельности;
- содержание работы;
- оформление работы;
- стили изложения;
- защита.

Оценка “отлично”

Степень самостоятельности

Работа выполнена студентом самостоятельно.

Содержание

В работе присутствуют все структурные элементы (оглавление, введение, основная часть из пяти разделов, заключение, список использованных источников).

Все этапы проектирования базы данных грамотно, логично изложены.

Количество выделенных сущностей в предметной области: не менее 7.

Представлены модели проектирования базы данных: концептуальная, логическая, физическая (структуры таблиц и схема базы данных).

Создание логической модели выполнено с использованием программного CASE-средства.

База данных нормализована.

Построены диаграммы деятельности для всех операций проектируемого приложения.

В базе данных реализованы и описаны все запросы корректировки данных, выделенные в процессе проектирования.

Реализованы запросы на выборку (в том числе запросы с группировкой и вложенные запросы).

Выполнено описание структуры приложения.

Показаны этапы проектирования форм на примере одной формы.

Показаны этапы проектирования отчетов на примере одного отчета.

Реализован удобный интерфейс для пользователя базы данных.

Реализовано как минимум два дополнительных способа навигации по формам.

Реализованы все алгоритмы обработки данных, выявленные на этапе проектирования, средствами макросов и программ.

Приложение реализовано в полном объеме, работает без ошибок.

Оформление

Работа оформлена в соответствии с требованиями, изложенными в документе “Требования к оформлению письменных работ, выполняемых студентами и слушателями ДВФУ”.

Стиль изложения

Продемонстрирован высокий уровень владения стилем письменной речи, работа полностью соответствует нормам лексики, использованы грамотные речевые обороты.

Защита работы

Слайды презентационных материалов содержат необходимую и достаточную информацию для визуальной поддержки доклада, не содержат мелкий текст, не перегружены эффектами, анимацией.

В доклад включена демонстрация работы созданного приложения. Программа работает без ошибок.

Студент полностью владеет теорией и практикой по теме выступления, отсутствуют ошибки в использовании терминов и понятий.

Время выступления полностью соответствует нормативному времени (10 минут).

Студент грамотно отвечает на все заданные вопросы, держится уверенно, ответы аргументированы, логически выстроены.

Оценка “хорошо”

Степень самостоятельности

Работа выполнена студентом самостоятельно.

Содержание

В работе присутствуют все структурные элементы (оглавление, введение, основная часть из пяти разделов, заключение, список использованных источников).

Все этапы проектирования базы данных грамотно, логично изложены. *Есть несущественные замечания по описанию проектирования базы данных.*

Количество выделенных сущностей в предметной области: не менее 6.

Представлены модели проектирования базы данных: концептуальная, логическая, физическая (структуры таблиц и схема базы данных).

Создание логической модели выполнено с использованием программного CASE-средства.

База данных нормализована.

Построены диаграммы деятельности для всех операций проектируемого приложения. *Есть несущественные замечания по описанию диаграмм деятельности.*

В базе данных реализованы и описаны все запросы корректировки данных, выделенные в процессе проектирования.

Реализованы запросы на выборку (в том числе запросы с группировкой и вложенные запросы).

Выполнено описание структуры приложения.

Показаны этапы проектирования форм на примере одной формы.

Показаны этапы проектирования отчетов на примере одного отчета.

Реализован удобный интерфейс для пользователя базы данных.

Реализовано *один* дополнительный способ навигации по формам.

Реализованы все алгоритмы обработки данных, выявленные на этапе проектирования, средствами макросов и программ.

Есть незначительные замечания по реализации приложения.

Оформление

Работа *не полностью* соответствует требованиям, изложенными в документе “Требования к оформлению письменных работ, выполняемых студентами и слушателями ДВФУ”.

Стиль изложения

Продемонстрирован *достаточно высокий* уровень владения стилем письменной речи, работа *в основном* соответствует нормам лексики, использованы грамотные речевые обороты.

Защита работы

Слайды презентационных материалов содержат необходимую и достаточную информацию для визуальной поддержки доклада, не содержат мелкий текст, не перегружены эффектами, анимацией.

В доклад включена демонстрация работы созданного приложения. Программа работает без ошибок.

Студент *недостаточно уверенно* владеет теорией и практикой по теме выступления, отсутствуют ошибки в использовании терминов и понятий.

Время выступления *незначительно превышает* нормативное время (10 минут).

Студент грамотно отвечает на все заданные вопросы, держится уверенно, ответы на некоторые вопросы *недостаточно* аргументированы, логически выстроены.

Оценка “удовлетворительно”

Степень самостоятельности

Работа выполнена студентом самостоятельно.

Содержание

В работе присутствуют все структурные элементы (оглавление, введение, основная часть из пяти разделов, заключение, список использованных источников).

В работе присутствуют все этапы проектирования базы. *Есть существенные замечания по описанию проектирования базы данных. Работа по проектированию недостаточно аргументирована.*

Количество выделенных сущностей в предметной области: не менее 5.

Представлены модели проектирования базы данных: концептуальная, логическая, физическая (структуры таблиц и схема базы данных).

Создание логической модели выполнено с использованием программного CASE-средства.

База данных нормализована.

Построены диаграммы деятельности для всех операций проектируемого приложения. *Есть существенные замечания по описанию диаграмм деятельности или отсутствует описание некоторых операций.*

В базе данных реализованы и описаны *не все* запросы корректировки данных, выделенные в процессе проектирования.

Реализованы запросы на выборку.

Описание структуры приложения *выполнено не полностью*.

Не показаны этапы проектирования форм на примере одной формы.

Не показаны этапы проектирования отчетов на примере одного отчета.

Не реализовано ни одного дополнительного способа навигации по формам.

Реализованы *не все* алгоритмы обработки данных, выявленные на этапе проектирования, средствами макросов и программ.

Есть существенные замечания по реализации приложения.

Оформление

Работа *не полностью* соответствует требованиям, изложенными в документе “Требования к оформлению письменных работ, выполняемых студентами и слушателями ДВФУ”.

Стиль изложения

Продемонстрирован *средний уровень* владения стилем письменной речи, работа *не полностью* соответствует нормам лексики, *не всегда* используются грамотные речевые обороты.

Защита работы

Слайды презентационных материалов содержат *избыточную* информацию для визуальной поддержки доклада, *содержат* мелкий текст, *перегружены* эффектами, анимацией.

В доклад включена демонстрация работы созданного приложения, *программа работает не всегда верно*.

Студент *недостаточно уверенно* владеет теорией и практикой по теме выступления, так как допускает значительные ошибки в использовании части терминов и понятий.

Время выступления *превышает* нормативное время (10 минут).

Студент отвечает *не на все* заданные вопросы.

Оценка “неудовлетворительно”

Степень самостоятельности

Работа выполнена студентом *несамостоятельно или самостоятельно, но с большим количеством замечаний по содержанию и оформлению*.

Содержание

В работе присутствуют *не все* структурные элементы.

В работе изложены *не все* этапы проектирования базы данных. *Есть существенные замечания по логике изложения этапов проектирования базы данных. Работа по проектированию не аргументирована.*

Количество выделенных сущностей в предметной области: *менее 5*.

Представлены модели проектирования базы данных: концептуальная, логическая, физическая (структуры таблиц и схема базы данных).

Не построены диаграммы деятельности для всех операций проектируемого приложения.

В базе данных *не реализованы* запросы корректировки данных, выделенные в процессе проектирования.

Реализованы запросы на выборку.

Описание структуры приложения *не выполнено*.

Не показаны этапы проектирования форм на примере одной формы.

Не показаны этапы проектирования отчетов на примере одного отчета.

Не реализовано ни одного дополнительного способа навигации по формам.

Не реализованы алгоритмы обработки данных, выявленные на этапе проектирования, средствами макросов и программ.

Приложение не реализовано.

Оформление

Работа *не соответствует* требованиям, изложенными в документе “Требования к оформлению письменных работ, выполняемых студентами и слушателями ДВФУ”.

Стиль изложения

Продемонстрирован *низкий уровень* владения стилем письменной речи, работа *не полностью* соответствует нормам лексики, *не всегда* используются грамотные речевые обороты.

Защита работы

Слайды презентационных материалов содержат *избыточную* информацию для визуальной поддержки доклада, *содержат* мелкий текст, *перегружены* эффектами, анимацией, *либо слайды вообще отсутствуют*.

Студент *слабо владеет или не владеет* теорией и практикой по теме выступления, допускает значительные ошибки в использовании части терминов и понятий.

Время выступления *значительно меньше или больше* нормативного времени (10 минут).

Список литературы

Основная литература

- Волошина В.Н. Организация баз данных: учеб. пособие. – Владивосток: Изд-во ДВФУ, 2011. – 503 с.
- Дубровская А.О. Информатика: Базы данных: учеб. пособие для вузов. – Владивосток: Изд-во ТГЭУ, 2009. – 104 с.
- Кабанов В.А. Практикум Access [Электронный ресурс] / В.А. Кабанов. – М.: Инфра-М.
- Кузин А.В. Базы данных: учеб. пособие. – М.: Академия, 2008. – 315 с.
- Шустова Л.И. Базы данных: учебник / Л.И. Шустова, О.В. Тараканов. – М.: НИЦ ИНФРА-М, 2016. – 336 с.

Дополнительная литература

- Брешенков А.В. Проектирование объектов баз данных в среде Access [Электронный ресурс]: учеб. пособие / А.В. Брешенков, А.М. Губарь. – Электрон. дан. – М.: МГТУ им. Н.Э. Баумана (Московский государственный технический университет имени Н.Э. Баумана), 2006. – 183 с.
- Голенищев Э.П. Информационное обеспечение систем управления. Уч. пособие для вузов. – Ростов-н/Д.: Феникс, 2010. – 315 с.
- Дейт К.Дж. Введение в системы баз данных / К.Дж. Дейт. – М.: Вильямс, 2001. – 1072 с.
- Муравьев А.И. Базы данных [Электронный ресурс]: учеб. пособие. – Электрон. дан. – М.: ТУСУР (Томский государственный университет систем управления и радиоэлектроники), 2006. – 137 с.
- Озерова Г.П. Проектирование и реализация реляционной базы данных в среде Access: метод. указания для выполнения курсовых и лабораторных работ / Дальневосточный государственный технический университет; [сост. Г.П. Озерова]. – Владивосток: Изд-во Дальневосточного технического университета, 2004. – 48 с.

Перечень ресурсов информационно-телекоммуникационной сети “Интернет”

- Справочник с примерами по языку SQL. – Режим доступа: <http://sql.itsoft.ru>.
- Самоучитель по СУБД ACCESS. – Режим доступа: <http://accesshelp.ru/samouchitel-ms-access>.
- Введение в базы данных. – Режим доступа: <http://www.mstu.edu.ru/study/materials/zelenkov/toc.html>.

Приложения

Приложение А

Список и содержание тем курсовой работы

1. Проектирование и реализация базы данных “Абитуриент”

Университет состоит из совокупности институтов. В институтах имеются определенные наборы специальностей. Для каждой специальности определены свои вступительные экзамены, а также количество абитуриентов, которые могут быть зачислены по данной специальности.

В приемную комиссию абитуриентами подаются заявления на специальность, абитуриент может выбрать несколько специальностей. В базе данных хранятся характеристики абитуриента (фамилия, имя, отчество, средний балл аттестата и пр.).

В базу данных заносятся баллы за каждый экзамен (результаты ЕГЭ). На основе этих баллов осуществляется расчет проходного балла по каждой специальности, а затем зачисление студентов в соответствии с набранными баллами.

2. Проектирование и реализация базы данных “Библиотека”

В некоторой библиотеке хранятся книги. Книга имеет название, относится к одному жанру, имеет одного или нескольких авторов, издана в определенном издательстве. В библиотеке может быть несколько экземпляров одной книги. Каждый экземпляр имеет уникальный шифр.

Пользователями библиотеки являются читатели, о которых хранится информация об их фамилии, адресе и телефоне. Читатель может взять несколько экземпляров книг из имеющихся в наличии, при этом в личной карточке читателя сохраняются сведения о взятой книге и дате ее получения. При возврате книги в карточке читателя указывается дата возврата, и данный экземпляр книги становится доступным для других пользователей.

3. Проектирование и реализация базы данных “Компьютерное тестирование”

Существует несколько предметов, по которым нужно осуществлять компьютерное тестирование студентов. Каждый предмет включает некоторое количество вопросов. Ответы на вопрос представлены в виде вариантов ответов, один из них правильный.

Студент, который приходит сдавать тестирование, регистрируется в системе, указав свое имя, фамилию, отчество и адрес. После этого он может приступить к тестированию по одному или нескольким предметам. Каждому студенту случайным образом выбирается набор вопросов по предмету, на которые он должен ответить. Тестируемый отвечает на вопросы, его ответы заносятся в базу данных. По окончании ответов на вопросы студенту выставляется оценка – результат тестирования.

4. Проектирование и реализация базы данных “Коммунальные платежи”

В ЖЭУ начисляются коммунальные платежи владельцам городских квартир. Каждая квартира имеет адрес (улица, номер дома и квартира), у нее может быть только один владелец, квартира характеризуется площадью и количеством

жильцов, в ней проживающих. Некоторые владельцы квартир имеют льготы по оплате жилья.

Ежемесячно каждому владельцу начисляется плата за квартиру, которая включает: оплату за жилую площадь, за воду, отопление, горячую воду и т.д. Причем часть начислений зависит от площади квартиры, а часть от числа проживающих в ней. Тарифы могут меняться ежемесячно. Каждому владельцу выписывается счет с расшифровкой всех платежей и итоговой суммой платежа.

После того, как владелец оплатит счет, информация об этом заносится в базу данных и хранится там.

5. Проектирование и реализация базы данных “Расчет заработной платы”

Предприятие состоит из нескольких отделов. В каждом отделе работают сотрудники. Их оклады устанавливаются согласно штатному расписанию, действующему на данном предприятии. У каждого сотрудника могут быть различные надбавки, выраженные в процентах от оклада, либо непосредственно в рублях (премия, надбавка за стаж и пр.). Также на предприятии устанавливаются коэффициенты к заработной плате: ДВ коэффициент за выслугу лет и районный коэффициент в размере 30%. Эти надбавки вычисляются от суммарных начислений сотруднику.

Ежемесячно сотруднику вносятся изменения по надбавкам и удержаниям, а затем начисляется заработная плата в зависимости от количества отработанных им дней. При необходимости рассчитываются больничные и отпускные.

При начислении заработной платы от нее производятся отчисления единого подоходного налога, вычисление которого зависит от количества иждивенцев, имеющих у каждого сотрудника. Также некоторые сотрудники могут иметь дополнительные удержания.

Результатом расчета заработной платы является сводная ведомость с указанием всех начислений и удержаний, расчетный листок для каждого сотрудника и ведомость на оплату с указанием итоговой суммы выдачи денег каждому сотруднику.

6. Проектирование и реализация базы данных “Поликлиника”

Поликлиника обслуживает определенный район города. В поликлинике происходит деление данного района на участки, каждый участок – это одна или несколько улиц. Больные обращаются в поликлинику по месту жительства и записываются на прием к участковому врачу, согласно участку, или на прием к врачу-специалисту.

Каждый врач имеет график работы. Запись к врачам осуществляется на свободное время, на каждое посещение отводится некоторое количество времени (например, 30 мин). Регистратура поликлиники выдает направления на посещение врачей с указанием времени, а также ведет учет приема врачом больных.

7. Проектирование и реализация базы данных “Складской учет”

Некоторый склад имеет свой ассортимент товаров. Товары подразделяются по типу. Каждый тип имеет свои характеристики (наименование, производитель, упаковка и пр.). На складе хранится некоторое количество каждого наименования товара. При исчерпании определенного вида товара необходимо оформить заказ на него.

У склада есть свои поставщики, которые поставляют товар по определенным ценам. Продажа товаров со склада осуществляется оптом по цене, которая формируется как покупная стоимость и наценка. У склада есть заказчики, которые покупают тот или иной товар в нужном количестве. Для них формируется счет с полной расшифровкой количества купленного товара, его цены и стоимости.

В системе ведется учет состояния склада, обновление склада при поступлении новых товаров, а также при их продаже.

8. Проектирование и реализация базы данных “Гостиница”

В некотором отеле есть несколько номеров (комнат). Номера различаются по категориям, по ценам. В каждый конкретный момент времени номер может быть занят, свободен, забронирован.

В отеле есть ряд структурных подразделений, обслуживающих клиентов: ресторан, прачечная, телефонная служба.

В отеле проживает некоторое количество клиентов (каждый в определенном номере). Клиенты имеют ряд характеристик: фамилия, имя, отчество дата рождения, паспортные данные, срок проживания, форма оплаты и т.д. В период проживания собираются данные о пользовании клиентом различными услугами. В момент выезда выдается счет-итог, в котором указывается сумма к оплате.

9. Проектирование и реализация базы данных “Видеопрокат”

В прокате есть набор видеокассет. На каждой кассете – один фильм. Кассет с одинаковыми фильмами может быть несколько. Каждый фильм относится к некоторому жанру (боевик, комедия и т.д.). Фильмы могут также группироваться по другому признаку, например, ведущие актеры или год выпуска.

В прокате ведется картотека пользователей: его фамилия, адрес, а также информация о взятой кассете, дате выдачи, оплате и т.д.

Как только кассета выдается, ее количество в прокате уменьшается, в картотеку пользователя заносится дата выдачи кассеты, срок и сумма оплаты. При возврате кассеты проверяется, вовремя ли она сдана, при необходимости начисляется доплата. Количество кассет с этим фильмом автоматически увеличивается.

10. Проектирование и реализация базы данных “Автомобильный рынок”

Существует автомобильный рынок, представленный совокупностью магазинов и фирм по продаже автомашин и запасных частей к ним. В информационно-справочной системе хранится и постоянно обновляется информация об ассортименте товаров в каждом из магазинов с указанием количества и цены. Информацию предоставляют магазины, каждый магазин ежемесячно оплачивает услуги информационно-справочной системы.

Пользователь системы должен иметь возможность получить полную информацию о нужном ему товаре. А затем, при желании, заказать тот или иной товар. Заявка должна отправляться в магазин, а количество товара автоматически уменьшаться на заказанное количество. Также пользователь должен иметь возможность просмотреть полный ассортимент товара того или иного магазина.

11. Проектирование и реализация базы данных “Информационно-справочная система по развлекательным предприятиям города”

В городе есть совокупность развлекательных предприятий (фирм): театры, кинотеатры и т.д. Каждая организация имеет свой адрес и телефон. Фирма представляет информацию о своем репертуаре и ценах на билеты.

Информационно-справочная система должна выдавать информацию о репертуаре того или иного предприятия, его месторасположении, ценах на билеты и пр. Размещение информации в системе для фирм, представляющих рекламу, платно. Объявления имеют разную цену и срок действия. В системе должен вестись учет оплаты для каждой фирмы по каждому объявлению.

12. Проектирование и реализация базы данных “Городской телефонный справочник”

Все предприятия города подразделяются на группы: медицина, образование и пр. Часть телефонов являются личными, то есть принадлежащими конкретному владельцу. Каждое предприятие города имеет название, возможно, несколько адресов и несколько телефонов, установленных в разных отделах.

Реализовать возможность поиска информации по произвольным критериям (адрес, телефон и т.д.), а также корректировку и занесение информации в базу данных.

13. Проектирование и реализация базы данных “Учет дорожно-транспортных происшествий”

В базе данных хранится информация об автомобилях (марка автомобиля; серия и номер технического паспорта, государственный номер, номер двигателя, номер кузова). Каждый автомобиль зарегистрирован на определенного владельца. Информация о владельце включает следующие данные: фамилия имя и отчество водителя, адрес, серия и номер водительского удостоверения. Также имеется справочник, содержащий список нарушений правил дорожного движения и суммы штрафов за каждое нарушение.

Необходимо вести учет о совершенных нарушениях (ДТП) с указанием даты, когда оно было совершено, и кем нарушение зафиксировано (фамилия имя отчество инспектора).

14. Проектирование и реализация базы данных “Ремонт компьютеров”

Фирма выполняет ремонт компьютеров. Требуется разработать базу данных для хранения информации о выполнении ремонтных работ сотрудниками фирмы.

Фирма имеет прайс-лист, в котором описываются выполняемые работы и их стоимость. Для некоторых работ требуются запасные части, стоимость которых определена. В фирме имеется некоторое количество запчастей.

В фирме работают мастера, которые имеет категорию, с категорией связывается фиксированный процент от выполнения работ.

При оформлении заказа фиксируется дата выполнения заказа, вид выполненной работы, используемые запчасти, исполнитель работы, а также информация о заказчике.

В конце месяца каждому сотруднику начисляется зарплата в зависимости от его категории и объема выполненных работ.

15. Проектирование и реализация базы данных “Туристическое агентство”

Фирма предоставляет клиентам услуги по организации зарубежных туров. Цели туров могут быть различными: отдых, туризм, лечение и т.д. Клиенту представляется описание тура: страна и город назначения, список возможных отелей, используемый транспорт. Для транспорта известна его цена, а для каждого отеля – стоимость проживания в день.

При оформлении поездки указывается дата начала, количество дней тура, паспортные данные клиента, а также менеджер, оформивший заказ, и дату оформления.

Менеджеры получают фиксированный процент от проданного тура, процент устанавливается для каждого менеджера индивидуально.

График выполнения курсовой работы

№ этапа	Сроки (недели семестра)	Содержание	Результат
I	1-4	Проектирование и реализация информационно-логической схемы базы данных.	Совокупность таблиц и схема данных, реализованные в среде Access
II	5-7	Проектирование основных алгоритмов обработки данных. Реализация запросов на выборку	Диаграммы деятельности. Совокупность запросов, реализованных на языке SQL в среде Access
III	8-11	Проектирование структуры приложения. Проектирование форм. Проектирование отчетов	Совокупность форм и отчетов, реализованных в среде Access
IV	12-15	Проектирование и реализация приложения пользователя	Приложение пользователя, реализованное в среде Access
V	16-17	Написание отчета	Отчет по курсовой работе, подготовленный в среде Word. Подготовка презентации в Powerpoint
VI	18	Защита курсовой работы	