# Автоматизация коммерческой деятельности ТЦ Гипермаркет

БАЛТИЙСКАЯ ГОСУДАРСТВЕННАЯ АКАДЕМИЯ

РЫБОПРОМЫСЛОВОГО ФЛОТА

**Радиотехнический факультет**

**Кафедра Информационной безопасности**

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

К КУРСОВОЙ РАБОТЕ

По дисциплине: «Безопасность систем баз данных»

Тема: «Автоматизация коммерческой деятельности ТЦ Гипермаркет»

Студент: уч.гр ИБ-4

Станкевич И.Ю.

Руководитель:

доцент, к.т.н. Капустин В.В.

Калининград 2009

**ЗАДАНИЕ НА КУРСОВУЮ РАБОТУ**

по дисциплине «Безопасность систем баз данных»

Создать базу данных для «Автоматизация коммерческой деятельности ТЦ Гипермаркет» Исследовать заданную предметную область, выбрать существенные атрибуты. Построить концептуальную инфологическую модель предметной области.

На основе инфологической модели построить реляционную модель, установить связи между объектами. Задать первичные и внешние ключи. Провести нормализацию. Объяснить цель выполненных преобразований.

Рассмотреть вопросы соблюдения целостности и иных аспектов обеспечения информационной безопасности проектируемой БД. Проверить соответствие предлагаемых решений правилам целостности, предложить необходимые организационные и общетехнические меры обеспечения информационной безопасности с учетом возможной специфики БД.

Провести исследование полученной модели, создав несколько сложных запросов к полученной модели на языке SQL. Объяснить синтаксис использованных языковых конструкций.

### Перечень обязательных разделов пояснительной записки

Введение

1.   Построение инфологической модели

2.   Построение реляционной модели

3.   Нормализация

4.   Проектирование базы данных в ACCESS

5.   Проблемы обеспечения информационной безопасности в БД

6.   Создание SQL запросов

Заключение

Список использованных источников

**Содержание**

ЗАДАНИЕ НА КУРСОВУЮ РАБОТУ

Введение

Глава 1. Построение инфологической модели

Глава 2. Построение реляционной модели

Глава 3. Нормализация

Глава 4. Проектирование базы данных в ACCESS

Глава 6. Проблемы информационной безопасности

Глава 7. Создание SQL запросов

Заключение

Литература

**Введение**

База данных – это информационная система, в которой данные могут совместно использоваться многими приложениями. Это означает, что данные организованны таким образом, чтобы поддерживать не только одно конкретное приложение. Основное отличие БД от традиционной файловой системы – это многократное и разнообразное использование одних и тех же данных. Данные не привязаны к какому-либо конкретному приложению и не контролируется им. Отдельные приложения больше не отвечают за создание и ведение данных. Эти обязанности возлагаются на нижележащий уровень программного обеспечения – систему управления базой данных (СУБД). СУБД выполняет роль интерфейса между пользователями приложений и данными.

Кроме того, СУБД должна обеспечивать гарантии безопасности и целостности базы данных. Пользователи компьютера должны иметь возможность защитить свои данные от несанкционированного доступа, а также восстановить их в случае системных сбоев.

Централизованное обеспечение безопасности данных – важная особенность СУБД. Наиболее значительное преимущество систем с базами данных – это централизованное обеспечение целостности данных.

база данных автоматизация коммерция деятельность

**Глава 1. Построение инфологической модели**

Построение инфологической модели представляет собой процесс моделирования смыслового наполнения базы данных. Инфологическая модель включает следующие три основные компонента.

1. Сущности. Это элементы реального мира, которые могут существовать независимо. В данном случае сущностями являются: Накладная, Магазины, Продавец, Главный офис, Товары магазина, Данные магазина, Данные продавца, Описание продавца, Данные накладной, Данные главного офиса, Главный офис. Сущность представляется в инфологической модели прямоугольником, в котором указано её имя.

2. Атрибуты. Они описывают сущность. Атрибуты представляются овалами с указанием имен, которые прикреплены к сущности. В рассматриваемом случае сущности *Накладная* соответствуют: **id\_N** Сумма, Кем выдана, Дата выдачи, Налог, Вид товара, В какой магазин, id\_C; сущности *Магазины* соответствуют: **id\_M**, id\_P, Телефон Адрес; сущности *Продавец* соответствуют: **id\_P**, ФИО, Магазин, Адрес; сущности *Главный офис* соответствуют: **id\_C**,Телефон, Адрес; сущности *Товары магазина* соответствуют id\_M, Товары, Количество на складе; сущности *Данные магазина* соответствуют id\_M, Название; сущности *Данные продавца* соответствуют: id\_P, №; сущности *Описание продавца* соответствуют: **Адрес**, Телефон; сущности *Данные накладной* соответствуют: id\_N, №; сущности *Данные главного офиса* соответствуют: id\_C, Название; сущности *Главный офис* соответствуют: **id\_C**, Телефон, Адрес.

3. Связи. Связь представляет взаимодействие между сущностями. На диаграмме она изображается ромбом, который соединяет сущности, участвующие в связи. Например, связь между *Главным офисом* и *Данные главного офиса* будет один ко многим.

На Рис. 1 представлена инфологическая модель заданной базы данных:


# Глава 2. Построение реляционной модели

В настоящее время преобладает технология реляционных баз данных. Она обеспечивает относительно простые средства представления данных и манипулирования ими. В реляционной базе данных все данные хранятся в таблицах. Названия сущностей станут заголовками таблиц, а атрибуты станут столбцами. Целостность данных в реляционной базе данных основывается на концепции ключей. Первичный ключ (PK) – это атрибут который можно использовать для уникальной идентификации таблицы. Так у таблицы “1” первичным ключом станет “id\_M”, у таблицы “2” – “id-P”, у таблицы “3” – “id\_N”, таблица “4” будет идентифицироваться атрибутом “id\_C”. Внешний ключ (FK) – это атрибут, который существует в нескольких таблицах и является первичным ключом одной из этих таблиц. Связь проводим от первичного ключа одой таблицы до внешнего ключа другой таблицы. Реляционная модель представлена на Рис 2.

Рис. 2 - Реляционная модель

Глава 3. Нормализация

Нормализация – это процесс, позволяющий гарантировать эффективность структур данных в реляционной базе данных.

Первая нормальная форма требует, чтобы все значения полей были атомарными и все записи уникальными. Реляционная модель, представленная на Рис. 3.1, находится в первой нормальной форме.

Модель находится во второй нормальной форме, если она, во-первых, находится в первой нормальной форме; и, во-вторых, не содержит неключевых атрибутов, находящихся в частичной функциональной зависимости от первичного ключа. Исходя из определения, разбиваем таблицу “6” на две таблицы, вторую образовавшеюся таблицу назовем “8”. В таблице “6” у нас остался только один идентификатор “id\_P”, значит, неключевые атрибуты зависят от всего первичного ключа. В таблице “22” нет неключевых атрибутов, значит, частичной зависимости быть не может. Таким же образом разбиваем таблицы “14”, “55” и “23”. Реляционная модель во второй нормальной форме представлена на Рис. 3.

Модель находится в третьей нормальной форме, если она находится во второй нормальной форме и не имеет транзитивных зависимостей. Транзитивная зависимость – это зависимость между неключевыми атрибутами. Таким образом, выделяем из таблицы “75” неключевые атрибуты “26” и “63”, которые находятся в зависимости, в отдельную таблицу “72”. Получаем модель в третьей нормальной форме, которая представлена на Рис. 4.

Рис. 3 – Вторая нормальная форма

Рис. 4 – Третья нормальная форма

Глава 4. Проектирование базы данных в ACCESS

Microsoft Access – это СУБД предназначенная для хранения и поиска информации, её представления в удобном виде и автоматизации часто повторяющихся операций (рис. 5). Чтобы реализовать базу данных в access, надо ввести через режим конструктора свою модель. Для начала надо ввести название таблиц и всех их атрибутов. Здесь же задается тип данных и первичный ключ (рис. 6).

Рис. 5 – Таблицы в access

Рис. 6 – атрибуты таблицы “Накладная”

Затем представляем свою реляционную модель третей нормальной формы в схеме данных (рис. 7).

Рис. 7 – Схема данных

После этого вводим в таблицы данные и делаем запросы.

Глава 6. Проблемы информационной безопасности

Решение проблем информационной безопасности применительно к БД предусматривает использование как обычных – организационных и общетехнических мер, так и специфичных, обусловленных особенностями создания и эксплуатации баз данных. В частности, достоверность данных в любой момент времени представляется понятием целостности. В соответствии со спецификой структуры БД различают следующие виды целостности:

-      по сущностям;

-      по ссылкам;

-      определяемые пользователем.

Для любых реляционных БД должно обеспечиваться безусловное следующих правил целостности:

Не допускается, чтобы какой-нибудь атрибут, участвующий в первичном ключе, принимал неопределенное значение.

Значение внешнего ключа должно быть либо равно значению первичного ключа цели, либо быть полностью неопределенным, то есть каждое значение атрибута, участвующего во внешнем ключе, должно быть неопределенным.

Для любой конкретной БД разработчиком должны быть дополнительно заданы правила, обеспечивающие:

-      уникальность тех или иных атрибутов;

-      ограничение диапазона изменения параметров;

-      принадлежность переменных разрешенному набору значений.

В рамках выполнения курсовой работы следует оценить соответствие инфологической модели, построенных на ее основе таблиц и реализации БД требованиям целостности и предложить комплекс общетехнических и организационных мер, способствующих повышению ее информационной безопасности.

# Глава 7. Создание SQL запросов

SQL – это язык реляционных баз данных, позволяющий пользователю создавать и удалять данные, содержащиеся в базе, управлять ими и налагать правила, обеспечивающие целостность реляционных данных.

Посредством механизма SQL становится возможным как обращаться к содержимому баз Microsoft Access из других приложений (в том числе и приложений от других производителей), так и получать данные для Microsoft Access из внешних приложений, поддерживающих SQL. Кроме того, язык запросов удобно использовать еще в тех случаях, когда нужно получать типовые данные, расположенные в одном и том же месте в строго определенном порядке, но разрабатывать для этого целый стандартный запрос (с таблицами и прочими атрибутами) совершенно излишне. Это также позволяет на примерах и относительно быстро разобраться в языке запросов, за счет того, что внутри любого запроса или таблицы Microsoft Access на самом деле "лежит" запись на SQL.

Работа начинается с создания таблицы или запроса в обычной среде визуального программирования Microsoft Access. Затем проверяется работоспособность полученной конструкции. Если все нормально, то далее следует переключить представление таблицы или запроса в режим языка запросов и внимательно изучить полученную последовательность команд. При желании, с ними можно поэкспериментировать для лучшего понимания.

Переключение "в SQL" делается следующим образом. Необходимо активизировать интересующий объект, например, в поле конструктора запроса (см. рис. 7.1), навести маркер мыши в любое место окна запроса или таблицы и нажатием правой клавиши мыши включить контекстно-зависимое меню. Выбрав РЕЖИМ SQL, преобразуем запрос в набор предложений языка запросов. Теперь можно, в зависимости от желания, либо просто просмотреть запись, либо что-нибудь в ней подредактировать.

Потом, когда нужда в SQL отпадет, можно все вернуть в исходное состояние. Это делается опять же через контекстно-зависимое меню мыши, которое, правда, имеет несколько другой вид. В нем существует РЕЖИМ ТАБЛИЦЫ, который превращает набор предложений языка запросов в привычный графический модуль.

В сочетании с достаточно мощным встроенным в Microsoft Access учебником по SQL такой прием позволяет довольно быстро наработать необходимые навыки, если не для полноценного программирования на SQL, то, во всяком случае, для того, чтобы понимать его особенности и чувствовать, когда имеет смысл использовать именно конструкции на SQL.

Примеры запросов.

1.         Надо отсортировать номера телефонов продавцов по возрастанию (рис. 8); пишем следующий запрос:

SELECT [описание продавца].адрес, [описание продавца].телефон

FROM [описание продавца]

ORDER BY [описание продавца].телефон;

Оператор SELECT создает требуемую таблицу, с помощью FROM перечисляются таблицы, содержащие необходимые для выполнения запроса данные, FROM – оператор сортировки по возрастанию. Этот запрос можно описать так: создать таблицу со столбцами “адрес” и “телефон”, сортируя таблицу “телефон” по возрастанию.

В итоге появится таблица с телефонами расположенными в порядке возрастания (рис. 8).

Рис. 8 – Запрос SQL в access

Рис. 9 – Результат запроса

2.   Запрос выводит поля: «налог», «в какой магазин» – из таблицы «Накладная» и выбирает те магазины, у которых налог составит <500 и название имеет окончание и:

SELECT [Накладная].[Налог], [Накладная].[Вид товара]

FROM Накладная

WHERE ((([Накладная].[Налог])<"500") And (([Накладная].[Вид товара]) Like "\*и"));

3.   Следующий запрос выводит поля: “id\_N” и “№” из таблицы “Данные накладной” и отбирает номера накладных больше 40000000000.

SELECT [данные накладной].[id\_N], [данные накладной].[№]

FROM [данные накладной]

WHERE ((([данные накладной].[№])>"40000000000"));

**Заключение**

В отчете представлены результаты работы по исследованию и проектированию базы данных для < > В полученной базе можно хранить данные о < >. и использовать ее совместно с другими приложениями, например, в целях< >..

Проектирование заключалось в построении инфологической модели, разработке на её основе реляционной модели и реализации базы в Microsoft Access.

В ходе работы были исследованы вопросы обеспечения информационной безопасности проектируемой БД, изучены команды языка SQL и реализованы примеры запросов.

**Литература**

1. Смирнов С.Н. Безопасность систем баз данных. - Гелиос АРВ, 2007

2. Смирнов С.Н. Работаем с Oracle. Учебное пособие. - Гелиос АРВ, 2002

3. Бен Чанг, Марк Скардина, Стефан Киритцов. Использование Oracle9i XML. Разработка приложений – М., Лори, 2003

4. Хансен Г., Хансен Д. Базы данных: разработка и управление. – М.: Бином, 2004

5. Артеменко Ю.Н., Волкова Я.П., Мухин Н.А. MySQL Справочник по языку – М.: 2005

6. Дейт Дж. - Введение в системы баз данных. - Вильямс. 2005 г. 1072 с.

7. Дэн Тоу Настройка SQL. Для профессионалов. Oracle, DB2, SQL Server. - Питер, BHV, 2002

8. Андон Ф., Резниченко В. - Язык запросов SQL. Учебный курс. – M., Питер, 2006