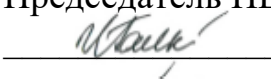


ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ФИНАНСОВЫЙ УНИВЕРСИТЕТ ПРИ ПРАВИТЕЛЬСТВЕ РОССИЙСКОЙ
ФЕДЕРАЦИИ»
ПЕРМСКИЙ ФИНАНСОВО-ЭКОНОМИЧЕСКИЙ КОЛЛЕДЖ –
ФИЛИАЛ ФИНУНИВЕРСИТЕТА

РАССМОТРЕНО
на заседании ПЦК
информационных систем и
программирования
Протокол № 6 от «24» января 2024 г.
Председатель ПЦК
 /И.В. Галкин

**МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО
ВЫПОЛНЕНИЮ ЛАБОРАТОРНОЙ РАБОТЫ НА ТЕМУ:
«ПРОЕКТИРОВАНИЕ СТРУКТУРЫ. ПРИВЕДЕНИЕ БАЗЫ ДАННЫХ
К НОРМАЛЬНОЙ ФОРМЕ ЗНФ»
МДК. 11.01 ТЕХНОЛОГИЯ РАЗРАБОТКИ И ЗАЩИТЫ БАЗ ДАННЫХ**

для специальности 09.02.07 Информационные системы и программирование

Пермь — 2024

Разработчик

Тотьмянина Л.В., преподаватель высшей квалификационной категории
Пермского финансово-экономического колледжа - филиала федерального
государственного образовательного бюджетного учреждения «Финансовый
университет при Правительстве Российской Федерации».

Одобрено Научно-методическим советом колледжа
Протокол № 3 от 28 марта 2024 г.

Оглавление

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА.....	3
ЛАБОРАТОРНАЯ РАБОТА № 1 ПРОЕКТИРОВАНИЕ СТРУКТУРЫ. ПРИВЕДЕНИЕ БД К НОРМАЛЬНОЙ ФОРМЕ ЗНФ.....	4
Краткие теоретические сведения.....	4
Ход работы	7
Контрольные вопросы	8

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Методические рекомендации МДК. 11.01 Технология разработки и защиты баз данных разработаны в соответствии с Федеральными государственными образовательными стандартами специальности 09.02.07 Информационные системы и программирование.

В методических рекомендациях содержатся материалы, необходимые студентам для освоения инструментария СУБД и применения его на практике, повышения уровня формирования компетенций обучающихся темы курса «Проектирование структуры. Приведение базы данных к нормальной форме ЗНФ».

По выполнению работы обучающиеся должны предоставить отчет о выполненной работе, используя программные приложения.

На выполнение лабораторной работы отводится не менее 2 часов.

Методические рекомендации могут быть успешно использованы преподавателями в проведении практических(лабораторных) занятий на дисциплинах «Информатика», «Основы проектирования баз данных», а также в дополнительном профессиональном образовании в программах повышения квалификации.

ЛАБОРАТОРНАЯ РАБОТА № 1 ПРОЕКТИРОВАНИЕ СТРУКТУРЫ.

ПРИВЕДЕНИЕ БД К НОРМАЛЬНОЙ ФОРМЕ 3НФ

Цель: формирование практических умений и навыков логического проектирования базы данных: структуры базы данных; структуры таблиц, входящих в состав базы данных; связей между таблицами.

Закрепление знаний применения правил нормализации при проектировании.

Краткие теоретические сведения

Процесс нормализации заключается в приведении таблиц в так называемые нормальные формы. Существует несколько видов нормальных форм: первая нормальная форма (1НФ), вторая нормальная форма (2НФ), третья нормальная форма (3НФ), нормальная форма Бойса-Кодда (НФБК), четвертая нормальная форма (4НФ), пятая нормальная форма (5НФ). С практической точки зрения, достаточно трех первых форм - следует учитывать время, необходимое системе для «соединения» таблиц при отображении их на экране. Поэтому мы ограничимся изучением процесса приведения отношений к первым трем формам.

Этот процесс включает:

- устранение повторяющихся групп (приведение к 1НФ);
- удаление частично зависимых атрибутов (приведение к 2НФ);
- удаление транзитивно зависимых атрибутов (приведение к 3НФ).

Приведение к первой нормальной форме

Когда поле в данной записи содержит более одного значения для каждого вхождения первичного ключа, такие группы данных называются *повторяющимися группами*. 1НФ не допускает наличия таких многозначных полей.

Приведение ко второй нормальной форме

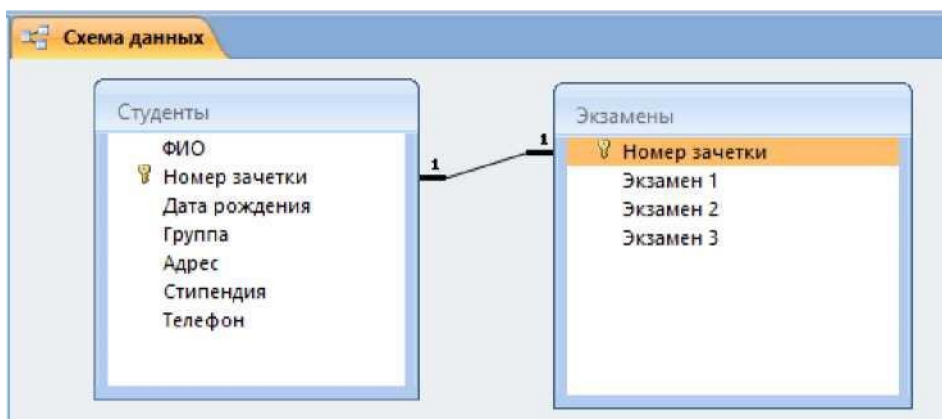
Следующий важный шаг в процессе нормализации состоит в удалении всех неключевых атрибутов, которые зависят только от части первичного ключа. Такие атрибуты называются частично зависимыми. Неключевые атрибуты заключают в себе информацию о данной сущности предметной области, но не идентифицируют ее уникальным образом.

Приведение к третьей нормальной форме

Третий этап процесса приведения таблиц к нормальной форме состоит в удалении всех неключевых атрибутов, которые зависят от других неключевых атрибутов. Каждый неключевой атрибут должен быть логически связан с атрибутом (атрибутами), являющимся первичным ключом.

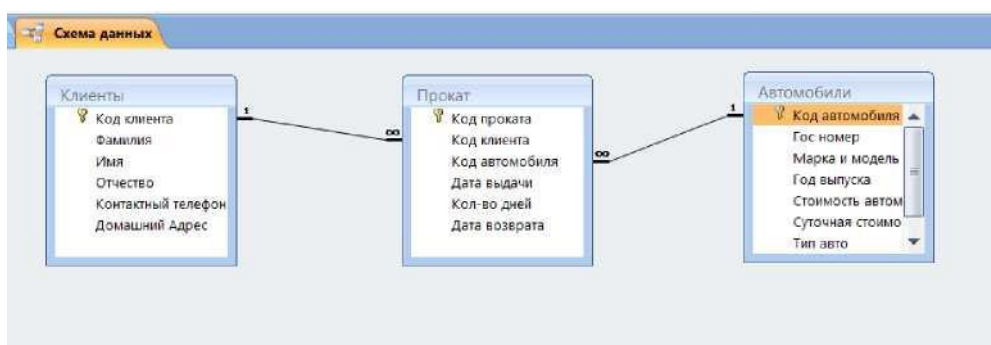
Каждый из различных типов связей должен быть смоделирован в базе данных.

Связь «один-к-одному» представляет собой простейший вид связи данных, когда первичный ключ таблицы является в то же время внешним ключом, ссылающимся на первичный ключ другой таблицы. Такую связь бывает удобно устанавливать тогда, когда невыгодно держать разные по размеру (или по другим критериям) данные в одной таблице. Например, можно выделить данные с подробным описанием изделия в отдельную таблицу с установлением связи «один-к-одному» для того чтобы не занимать оперативную память, если эти данные используются сравнительно редко.



Связь «один-ко-многим» в большинстве случаев отражает реальную взаимосвязь сущностей в предметной области. Она реализуется уже описанной парой «внешний ключ - первичный ключ», т.е. когда определен

внешний ключ, ссылающийся на первичный ключ другой таблицы. Именно эта связь описывает широко распространенный механизм классификаторов. Имеется справочная таблица, содержащая названия, имена и т.п. и некие коды, причем, первичным ключом является код. В таблице, собирающей информацию - назовем ее информационной таблицей - определяется внешний ключ, ссылающийся на первичный ключ классификатора. После этого в нее заносится не название из классификатора, а код. Такая система становится устойчивой от изменения названия в классификаторах. Имеются способы быстрой «подмены» в отображаемой таблице кодов на их названия как на уровне сервера БД (для клиент-серверных СУБД), так и на уровне пользовательского приложения.



Связь «многие-ко-многим» в явном виде в реляционных базах данных не поддерживается. Однако имеется ряд способов косвенной реализации такой связи, которые с успехом возмещают ее отсутствие. Один из наиболее распространенных способов заключается во введении дополнительной таблицы, строки которой состоят из внешних ключей, ссылающихся на первичные ключи двух таблиц. Например, имеются две таблицы: **КЛИЕНТ** и **ГРУППА_ИНТЕРЕСОВ**. Один человек может быть включен в различные группы, в то время как группа может объединять различных людей. Для реализации такой связи «многие-ко-многим» вводится дополнительная таблица, назовем ее **КЛИЕНТЫ_В_ГРУППЕ**, строка которой будет иметь два внешних ключа: один будет ссылаться на первичный ключ в таблице **КЛИЕНТ**, а другой - на первичный ключ в таблице **ГРУППА_ИНТЕРЕСОВ**.

Таким образом, в таблицу КЛИЕНТЫ_В_ГРУППЕ можно записывать любое количество людей и любое количество групп.

Ход работы

1 Определить информационные потребности базы данных (вариант определяет преподаватель):

- 1) База данных библиотеки
- 2) База данных кинотеатра
- 3) База данных учебного заведения
- 4) База данных книжного магазина
- 5) База данных коммерческой организации
- 6) База данных театра
- 7) База данных железнодорожного вокзала
- 8) База данных автовокзала
- 9) База данных спортивного клуба
- 10) База данных бассейна
- 11) База данных больницы
- 12) База данных поликлиники
- 13) База данных музыкальной школы
- 14) База данных цветочного магазина
- 15) База данных магазина бытовой химии
- 16) База данных магазина мебели
- 17) База данных центра связи
- 18) База данных магазина компьютерной техники
- 19) База данных магазина канцелярских товаров
- 20) База данных кафе-ресторана
- 21) База данных телевизионного канала
- 22) База данных радиостанции
- 23) База данных фермерского хозяйства
- 24) База данных автоклуба
- 25) База данных аэропорта

2 Проанализировать объекты реального мира, которые необходимо сформулировать в базе данных.

3 Сформировать из этих объектов сущности и их характеристики.

Например, сущность - «деталь»;

характеристики - «марка», «цвет» и т.д.

Сформировать список сущностей и характеристик.

4 Поставить в соответствие сущностям и характеристикам - таблицы и атрибуты таблиц.

5 Определить атрибуты, которые будут уникальным образом идентифицировать каждый объект.

6 Провести нормализацию таблиц, что бы избежать проблем при работе с базой данных: аномалий удаления, обновления, вставки. Установить связи между таблицами, указать первичные и внешние ключи. Связи должны быть организованы таким образом, чтобы по данным одной таблицы можно было получить данные из любой другой таблицы.

7 Спланировать вопросы надежности данных и при необходимости сохранение секретности информации, ответив на вопросы: кто пользуется информацией, способы доступа к информации, степень защищенности информации, кто сопровождает базу данных и другие вопросы.

8 Оформить подробный отчет и сделать вывод по выполненной работе.

Контрольные вопросы

- 1 В чем заключается процесс проектирования базы данных?
- 2 Логическое проектирование. Этапы.
- 3 Физическое проектирование. Назначение.
- 4 Что представляет собой структура базы данных?
- 5 Что представляет собой структура таблицы?
- 6 Виды связей и их реализация.
- 7 Процесс нормализации. Ошибки, возникающие при работе с ненормализованной базой данных. Нормальные формы.
- 8 Понятие таблица. Составляющие таблицы.
- 9 Способы представления таблиц.
- 10 Организация связи «многие-ко-многим» (*-*). Пример по конкретной базе данных