

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ
по лабораторной работе №1
по дисциплине «Базы данных»
ТЕМА: ПРОЕКТИРОВАНИЕ ER МОДЕЛИ И СТРУКТУРЫ БД

Студентка гр. 2384

Соц Е.А.

Преподаватель

Заславский М.М.

Санкт-Петербург

2024

Цель работы

Изучить основные принципы построения ER модели и структуры баз данных, спроектировать ER модель и структуру базы данных по текстовому описанию предметной области.

Задание

Вариант 18

Пусть требуется создать программную систему, предназначенную для менеджера музыкальных групп. Такая система должна обеспечивать хранение сведений о группах, включающих название группы, год образования и страну, состав исполнителей, положение в последнем хит-параде; репертуар группы. Сведения о каждой песне из репертуара группы — это ее название, композитор, автор текста. Необходимо также хранить данные о последней гастрольной поездке каждой группы: название гастрольной программы, названия населенных пунктов, дата начала и окончания выступлений, средняя цена билета (зависит от места выступления и положения группы в хит-параде). Возможно появление новой группы и изменение состава исполнителей. Каждая песня может быть в репертуаре только одной группы. Менеджеру могут потребоваться следующие сведения:

- Автор текста, композитор и дата создания песни с данным названием? В репертуар какой группы она входит?
- Репертуар наиболее популярной группы?
- Цена билета на последний концерт указанной группы?
- Состав исполнителей группы с заданным названием, их возраст и амплуа?
- Место и продолжительность гастролей группы с заданным названием?

- Какие группы в текущем году отмечают юбилей
- Самый молодой вокалист? Какую группу он представляет?

Выполнение работы

Создание ER модели

В ходе выполнения работы была спроектирована ER модель, представленная на рис. 1:

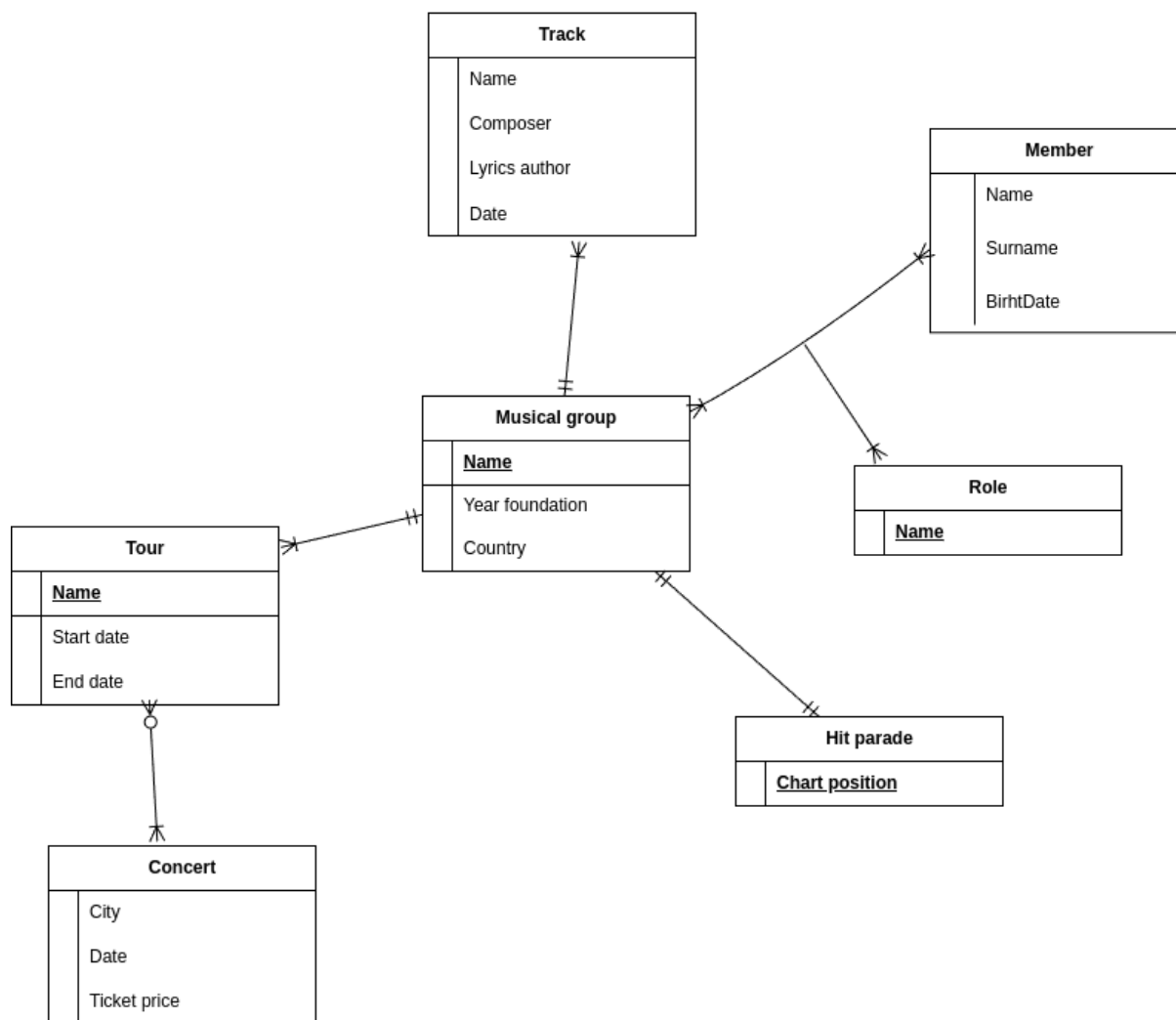


Рисунок 1 – ER модель

В ER модели были выделены следующие сущности:

1. *Musical group* – музыкальная группа. Ключом сущности является название музыкальной группы (*name*), атрибутом сущности является год формирования музыкальной группы (*year foundation*) и страна образования (*country*)
2. *Repertoire* – репертуар группы. Атрибутами являются название песни (*name*), композитор (*composer*), автор слов (*lyrics author*) и дата выхода

песни (*date*). Ключевого атрибута у данной сущности нет, так как могут быть песни с одинаковым названием, написанные одним человеком и так далее.

3. *Member* – участник. Атрибутами сущности являются имя участника (*name*), фамилия (*surname*) и дата рождения (*birthdate*). Ключевого атрибута здесь нет, так как могут быть люди с одинаковыми фамилиями и именами, рожденные в один день.

4. *Role* – роль участника в группе (например: вокалист, гитарист, барабанщик, автор слов). Ключевым атрибутом является название роли (*name*).

5. *Hit parade* – хит-парад. Ключевым атрибутом является положение группы в последнем хит-параде (*chart position*).

6. *Tour* – музыкальный тур. Ключом сущности является название тура (*name*). Атрибутами являются дата начала(*start date*), дата окончания(*end date*).

7. *Concert* – концерт из музыкального тура. Атрибутами сущности являются название города (*city*), дата проведения концерта (*date*) и цена за билет (*ticket price*).

Между полученными сущностями были выявлены следующие отношения:

Музыкальная группа имеет свой набор песен, но каждая песня принадлежит только одной определенной группе.

Музыкальная группа имеет одного/нескольких участников с какой-либо ролью.

Музыкальная группа всегда имеет какое-то одно место в хит-параде, которое может занимать только одна группа.

Музыкальная группа может отправляться в туры, причем он может быть как один у музыкальной группы, так их может быть и несколько. Тур с определенным названием может принадлежать только одной группе.

Тур состоит из одного или нескольких концертов в каких-либо городах, но может быть такое, что в городе не будет концерта, то есть он не входит в состав тура.

Полученная ER-диаграмма отражает структуру системы для менеджера музыкальных групп, позволяющую хранить сведения о группах, их репертуаре, гастрольных поездках и составе исполнителей. Система учитывает возможность появления новых групп и изменения состава.

Структура БД

На основании созданной ER-диаграммы была описана структура ДБ, схема которой представлена на рис. 2:

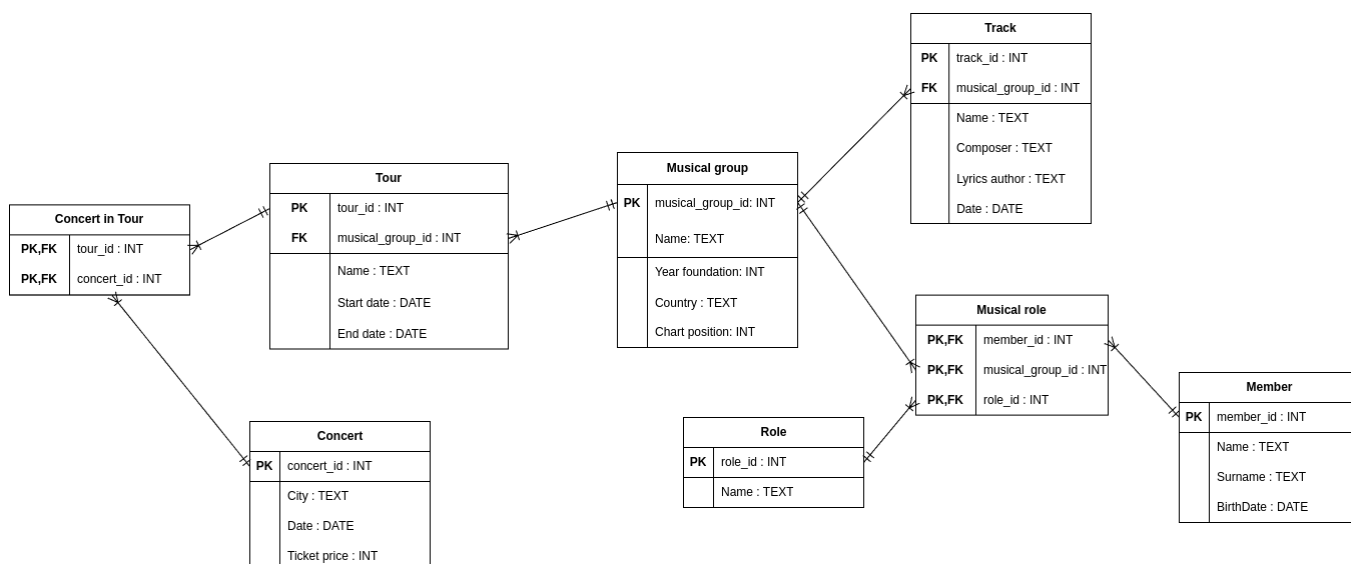


Рисунок 2 – Структура БД

В результате преобразования ER-диаграммы в структуру базы данных, произошли следующие изменения:

1. Добавление первичных ключей – идентификаторов

Для всех сущностей были добавлены поля идентификаторов (*_id), которые автоматически генерируются. Это касается сущностей: musical group (*musical_group_id*), repertoire (*track_id*), tour (*tour_id*), concert (*city_id*), member (*member_id*), role (*role_id*). Эти идентификаторы служат первичными ключами (PK).

2. Добавление типов данных

Чтобы описание структуры базы данных больше соответствовало настоящей БД были добавлены те типы данных, которые присутствуют в PostgreSQL. Все *ID и цены билетов – *INT*, названия и имена – *TEXT*, даты – *DATE*.

3. Замена многие ко многим отношений на таблицы

Отношения многие ко многим были заменены на отдельные таблицы:

Concert_in_tour — промежуточная таблица для связи между сущностями concert и tour. Она содержит составной первичный ключ, состоящий из внешних ключей (*concert_id* и *tour_id*), которые ссылаются на сущности concert и tour соответственно.

Musical_role — таблица для связи между сущностями member, musical group и role. Она содержит составной первичный ключ, включающий внешние ключи (*member_id*, *musical_group_id*, *role_id*), которые ссылаются на соответствующие сущности. Это позволяет отражать участие музыкантов в группах с указанием их ролей.

4. Добавление внешних ключей (FK) для отражения отношений один ко многим:

В сущностях repertoire и tour были добавлены внешние ключи (*musical_group_id*), чтобы отразить их связь с музыкальной группой. Это позволяет точно указать, какие треки и туры принадлежат той или иной группе.

Соответствие полученной структуры БД нормальной форме

Бойса — Кодда

Проверим структуру представленных таблиц на соответствие нормальной форме Бойса — Кодда (НФБК). Для этого рассмотрим каждую таблицу, определим функциональные зависимости и убедимся, что для

каждой нефункциональной зависимости левая часть является суперключом.

1. Таблица Musical Group

Зависимости:

- musical_group_id → name, year_foundation, chart_position, country

Musical_group_id – первичный ключ и суперключ; других неключевых зависимостей нет, поэтому таблица находится в НФБК, так как все функциональные зависимости тривиальны или левая часть является суперключом.

2. Таблица Repertoire

Зависимости:

- track_id → musical_group_id, name, date, composer, lyrics_author;
- Возможно musical_group_id, name → track_id, date, composer, lyrics_author, если имя трека уникально у данной группы.

Анализ зависимостей:

- track_id – первичный ключ и суперключ;
- musical_group_id & name является кандидатом на ключ, значит левая часть функциональной зависимости также является суперключом.

Таким образом таблица находится в НФБК.

3. Таблица Tour

Зависимости:

- tour_id → musical_group_id, name, start date, end date, avarage_ticket_price

tour_id – первичный ключ и суперключ, других неключевых зависимостей нет, поэтому таблица находится в НФБК.

4. Таблица Concert in tour

concert_id, tour_id – составной первичный ключ, нет дополнительных атрибутов или зависимостей, таблица находится в НФБК.

5. Таблица Concert

Зависимости:

- concert_id → city, date, ticket_price

Обратное неверно, значит таблица в НФБК.

6. Таблица Musical role

Зависимости:

- member_id, musical_group_id, role_id – составной первичный ключ
- Нет дополнительных атрибутов или зависимостей

Таблица находится в НФБК.

7. Таблица Member

Зависимости:

- member_id → name, surname, birthdate

Обратное неверно: два разных человека могут иметь одинаковые имена и фамилии и родиться в один день. Таблица в НФБК.

8. Таблица Role

Зависимости:

- role_id → name
- name → role_id (если названия ролей уникальны)

Анализ:

- role_id – первичный ключ и суперключ
- Нет зависимостей, где левая часть не является суперключом

Таблица находится в НФБК.

Все таблицы соответствуют нормальной форме Бойса — Кодда, так как для каждой нефункциональной зависимости левая часть является суперключом таблицы. Таким образом, структура базы данных правильно

нормализована и не содержит избыточностей, связанных с функциональными зависимостями.

Выводы

Приобретены практические навыки проектирования ER-модели и структуры базы данных по текстовому описанию предметной области. В ходе работы была разработана ER-модель на основе текстового описания предметной области, посвящённой менеджменту музыкальных групп. После этого была создана структура базы данных, соответствующая нормальной форме Бойса — Кодда, что обеспечивает эффективное и надёжное хранение данных без избыточности.

ПРИЛОЖЕНИЕ А
ССЫЛКА НА PULL REQUEST

Ссылка на PR – <https://github.com/moevm/sql-2024-2384/pull/5>