

Министерство образования Республики Беларусь

Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра электронных вычислительных машин

Дисциплина: Базы данных

ОТЧЁТ
по лабораторной работе № 4
Вариант № 28 (Студия звукозаписи).

Студент:

П. В. Сякачёв

Проверил:

Л. П. Поденок

МИНСК 2022

1 Цель работы

В лабораторной работе выполняется создание простых запросов на выборку данных на языке SQL с использованием предложений SELECT, FROM (JOINS), WHERE и ORDER BY оператора SELECT. В работе также требуется рассмотреть использование скалярных функций.

2 Выполнение работы

В данной лабораторной работе создаются рациональные, то есть эффективные для построенной базы данных «Студия звукозаписи» реализации оператора SELECT. Для построения запросов данные в таблицах были изучены, таким образом результаты выборки являются полностью ожидаемыми и не пустыми.

Далее приведены несколько примеров работы с требуемыми предложениями оператора.

Оператор SELECT с предложением FROM выводит строки по заданным в SELECT столбцам из заданной в FROM таблицы:

```
SELECT
    last_name AS "Surname",
    phone_number AS "Phone"
FROM member;
```

Ключевое слово AS – это сокращённое слово ALIAS, которое позволяет временно дать пользовательские названия для столбцов в выборке и не только.

Предложение ORDER BY сортирует полученные строки таблицы в заданном порядке по столбцу таблицы. В данном случае строки отсортированы по столбцу в алфавитном порядке:

```
SELECT *
FROM worker
ORDER BY last_name ASC;
```

Предложение DISTINCT исключает из выборки дубликаты:

```
SELECT DISTINCT speciality
FROM worker
ORDER BY speciality;
```

Предложение WHERE позволяет выбрать условие для выборки строк данных:

```
SELECT * FROM member WHERE gender = 'F';
```

Предложение LIKE позволяет искать строки таблицы по заданному паттерну:

```
SELECT *  
FROM member  
WHERE phone_number LIKE '%892%';
```

Пример использования предложения JOIN:

```
SELECT      recording_session.id AS "Session ID",  
            recording_session.session_type AS "Recording type",  
            band.id AS "Band ID",  
            band.name AS "Band name",  
            band.style AS "Band style",  
            room.id AS "Room ID",  
            room.room_number AS "Room number"  
FROM recording_session  
JOIN room ON recording_session.room_id = room.id  
JOIN band ON recording_session.band_id = band.id  
WHERE band.style LIKE '%rock%';
```

Также здесь используется функция TO_CHAR с заданным форматом вывода, которая преобразует формат данных timestamp в строку. В результате получаем удобочитаемый вид.

Предложение LENGTH подсчитывает количество символов в строке:

```
SELECT phone_number,  
       LENGTH(phone_number)  
FROM worker;
```

Полный список реализованных заданий для выборки представлен ниже:

1. Вывести данные о всех участниках всех музыкальных групп.
2. Вывести фамилии и номера всех участников всех музыкальных групп.
3. Вывести список работников, фамилии отсортированы в алфавитном порядке.
4. Вывести список всех должностей работников.
5. Вывести список участников групп женского пола.
6. Вывести список сессий звукозаписи, длительность которых меньше трёх часов, сессии были после 1-го сентября.
7. Вывести всю информацию о сессиях звукозаписи и группах, которые в них участвуют.

8. Вывести участников групп, телефонные номера которых содержат в себе "892".
9. Вывести тип сессии звукозаписи, группу и помещение, в котором оно проходит, при условии, что группа играет в стиле рок.
10. Вывести количество знаков в графе номер телефона у работников.
11. Вывести информацию о участниках группы и ролях в группе.

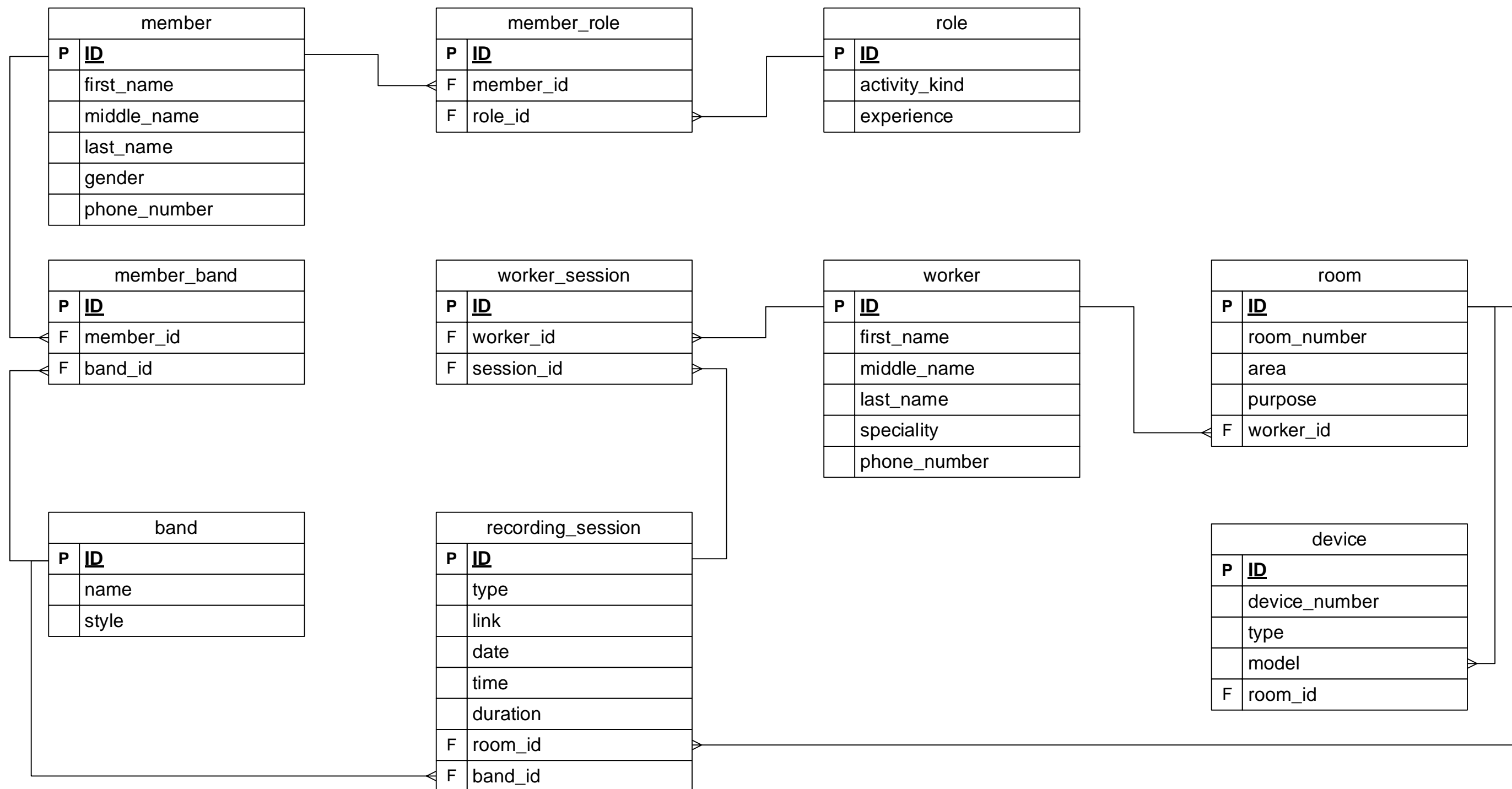


Рисунок 1 – Реляционная схема данных модели «Студия звукозаписи»