Задача 9.1.1 Транзакции

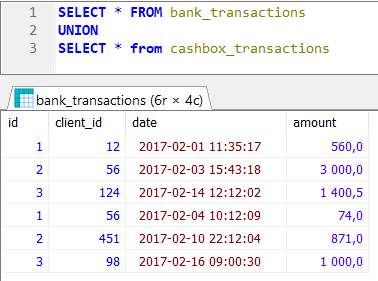
Объедините с помощью UNION таблицы bank\_transactions и cashbox\_transactions.

Таблица bank\_transactions

|  |  |  |  |
| --- | --- | --- | --- |
| id | client\_id | date | amount |
| 1 | 12 | 2017-02-01 11:35:17 | 560.00 |
| 2 | 56 | 2017-02-03 15:43:18 | 3000.00 |
| 3 | 124 | 2017-02-14 12:12:02 | 1400.50 |

Таблица cashbox\_transactions

|  |  |  |  |
| --- | --- | --- | --- |
| id | client\_id | date | amount |
| 1 | 56 | 2017-02-04 10:12:09 | 74.00 |
| 2 | 451 | 2017-02-10 22:12:04 | 871.00 |
| 3 | 98 | 2017-02-16 09:00:30 | 1000.00 |



Задача 9.1.2 Транзакции пользователя

Получите с помощью UNION из таблиц bank\_transactions и cashbox\_transactions данные обо всех транзакциях клиента 56. Выведите три столбца:

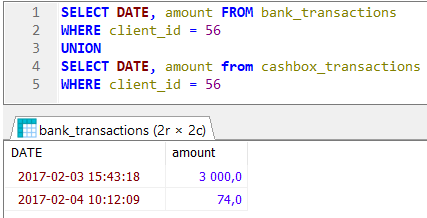
* date – дату транзакции в формате ДД.ММ.ГГ;
* amount – сумму транзакции;
* payment\_type – вид платежа: «bank» для банковских транзакций и «cash» для наличности.

Таблица bank\_transactions

|  |  |  |  |
| --- | --- | --- | --- |
| id | client\_id | date | amount |
| 1 | 12 | 2017-02-01 11:35:17 | 560.00 |
| 2 | 56 | 2017-02-03 15:43:18 | 3000.00 |
| 3 | 124 | 2017-02-14 12:12:02 | 1400.50 |

Таблица cashbox\_transactions

|  |  |  |  |
| --- | --- | --- | --- |
| id | client\_id | date | amount |
| 1 | 56 | 2017-02-04 10:12:09 | 74.00 |
| 2 | 451 | 2017-02-10 22:12:04 | 871.00 |
| 3 | 98 | 2017-02-16 09:00:30 | 1000.00 |



Задача 9.1.3 Слияние фирм

После слияния двух компаний, IT отделу поручили объединение баз данных. Слейте users и members в единую таблицу со следующими столбцами:

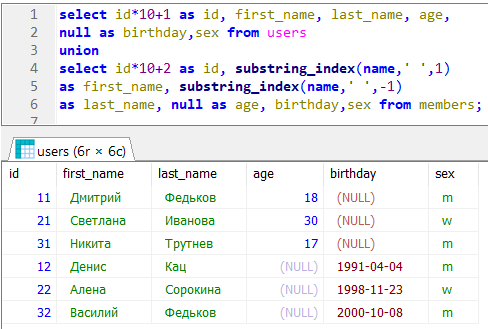
* id – идентификатор пользователя. Так как идентификаторы пользователей в таблицах одинаковые, то примените к ним следующую формулу id \* 10 + N, где N – равен одному для таблицы users и двум для members.
* first\_name – имя пользователя. Используйте оригинальное значение из таблицы users и извлеките имя из поля name таблицы members.
* last\_name – фамилия пользователя. Используйте оригинальное значение из таблицы users и извлеките фамилию из поля name таблицы members.
* age – возраст пользователя. Используйте оригинальное значение из таблицы users и NULL для данных из members.
* birthday  – день рождения пользователя. Используйте оригинальное значение из таблицы members и NULL для данных из users.
* sex – пол пользователя.

Таблица users

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| id | first\_name | last\_name | age | sex |
| 1 | Дмитрий | Федьков | 18 | m |
| 2 | Светлана | Иванова | 30 | w |
| 3 | Никита | Трутнев | 17 | m |

Таблица members

|  |  |  |  |
| --- | --- | --- | --- |
| id | name | birthday | sex |
| 1 | Денис Кац | 1991-04-04 | m |
| 2 | Алена Сорокина | 1998-11-23 | w |
| 3 | Василий Федьков | 2000-10-08 | m |



Задача 9.1.4 База ГИБДД

Каждый регион ведет базу автомобилей и их номеров в своей собственной уникальной базе.

Ваша задача — объединить таблицы из разных регионов так, чтобы получилась таблица со следующий структурой:

* number – номер автомобиля в формате а111аа.
* region – регион регистрации автомобиля (целое число).
* mark – марка автомобиля.
* model – модель автомобиля.

Таблица 1. cars

|  |  |  |  |
| --- | --- | --- | --- |
| id | number | mark | model |
| 1 | с065мк78 | Lada | Kalina |
| 2 | с165ор78 | Toyota | Prius |
| 3 | в764хх78 | Pegeuot | 307 |

Таблица 2. region39

|  |  |  |  |
| --- | --- | --- | --- |
| id | number | mark | model |
| 1 | а123ор | Ford | Fiesta |
| 2 | a455аа | Ford | Mondeo |
| 3 | х104ор | Mazda | Demio |

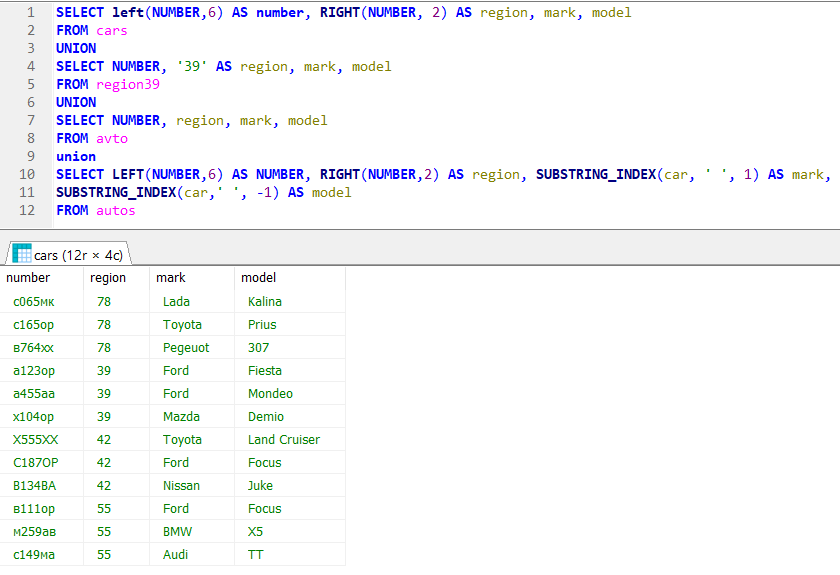
Таблица 3. avto

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| id | number | region | mark | model |
| 1 |  |  |  |  |
| 2 |  |  |  |  |
| 3 |  |  |  |  |

Таблица 4. autos

|  |  |  |
| --- | --- | --- |
| id | number | car |
| 1 | в111ор55 | Ford Focus |
| 2 | м259ав55 | BMW X5 |
| 3 | с149ма55 | Audi TT |

Таблицы должны идти в следующем порядке: cars, region39, avto, autos.



9.2 ОБЪЕДИНЕНИЕ С ПОМОЩЬЮ UNION: СОРТИРОВКА

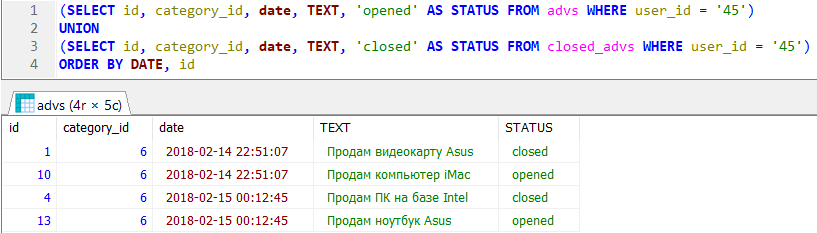
Задача 9.2.1 Объявления пользователя

В целях оптимизации производительности в базе данных есть две таблицы для хранения объявлений: advs и closed\_advs, в первой хранятся активные объявления, а во второй неактивные.

Основная часть запросов приходится на advs, но когда пользователь заходит в личный кабинет, то видит все свои объявления из обеих таблиц.

Объедините обе таблицы с помощью UNION и получите данные для пользователя с идентификатором 45. Выведите колонки id, category\_id, date, text, а также status (статус объявления). Для открытых объявлений в поле status укажите «opened» а для закрытых – «closed».

Итоговую таблицу отсортируйте по дате, а затем по id.

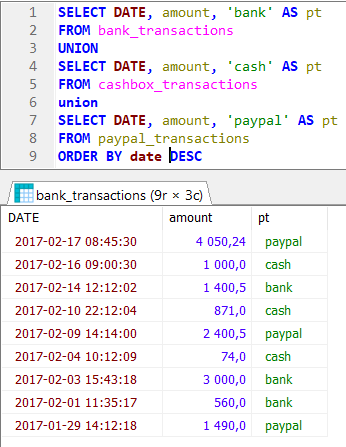


Задача 9.2.2 Транзакции по датам

Объедините с помощью UNION таблицы bank\_transactions, cashbox\_transactions и paypal\_transactions. Выведите три столбца:

* date – дату и время транзакции;
* amount – сумму транзакции;
* pt – вид платежа: «bank» для банковских транзакций и «cash» для наличности и «paypal» для платежей в системе PayPal.

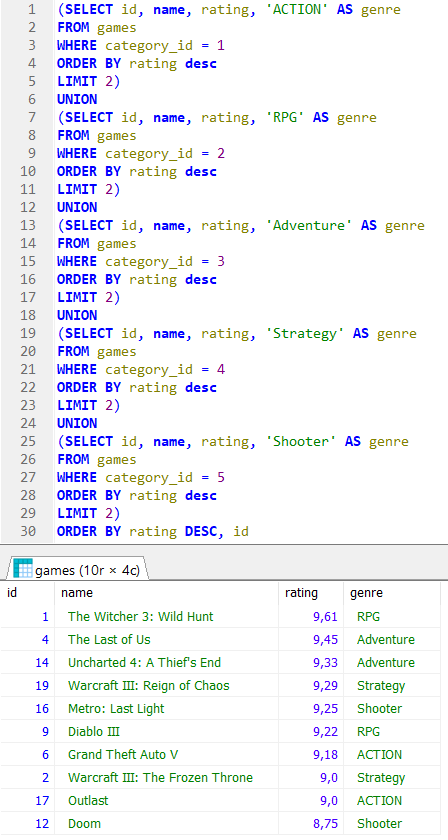
Итоговые данные отсортируйте по дате в обратном хронологическом порядке.



Задача 9.2.3 Главная страница

На главной странице сайта выводится ТОП10 компьютерных игр. Чтобы вывод был разнообразным — берут по две самых популярных игры из 5 категорий, а затем сортируют полученные данные по рейтингу в обратном порядке. Строки с одинаковым рейтингом отсортируйте по id.

Получите из таблицы games данные для вывода на главную страницу сайта для категорий: 1 - Action, 2 - RPG, 3 - Adventure, 4 - Strategy и 5 - Shooter. Выведите поля id, name, rating и genre, где genre — название категории.



9.3 ОБЪЕДИНЕНИЕ С ПОМОЩЬЮ UNION: ГРУППИРОВКА

Задача 9.3.1 Типы транзакций

Платежи клиентов в базе данных хранятся в трех таблицах:

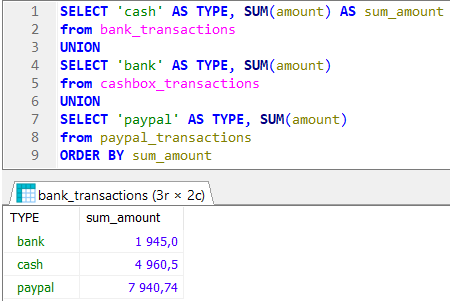
1. bank\_transactions — прямые платежи на расчетный счет,

2. cashbox\_transactions — платежи наличными в кассу организации,

3. paypal\_transactions — платежи через сервис PayPal.

Получите виды платежей и суммы по каждому виду. Виды платежей выведите в колонке type: «bank» для банковских транзакций и «cash» для наличности и «paypal» для платежей в системе PayPal. Суммы платежей в колонке sum\_amount.

Итоговые данные отсортируйте по сумме.



Задача 9.3.2 Ежемесячные доходы

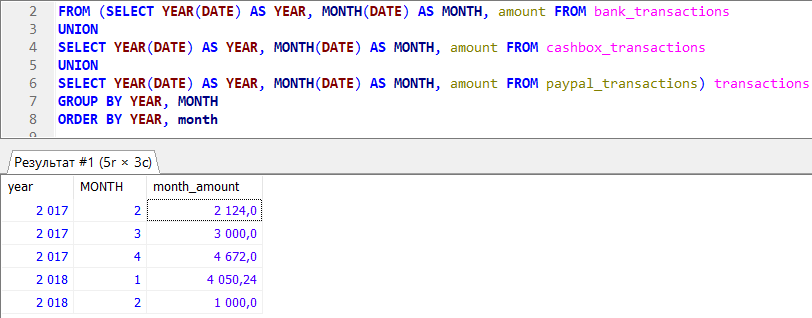
Платежи клиентов в базе данных хранятся в трех таблицах:

1. bank\_transactions — прямые платежи на расчетный счет,

2. cashbox\_transactions — платежи наличными в кассу организации,

3. paypal\_transactions — платежи через сервис PayPal.

Получите суммы входящих платежей по всем таблицам с разбивкой по годам и месяцам. Итоговая таблица должна содержать 3 столбца: year, month, month\_amount. Итоговые данные отсортируйте по году и месяцу.



Задача 9.3.3 Общая сумма

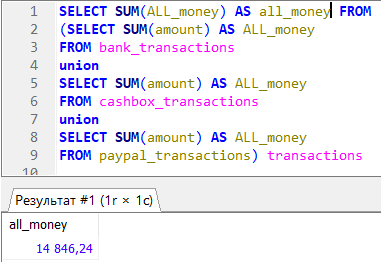
Платежи клиентов в базе данных хранятся в трех таблицах:

1. bank\_transactions — прямые платежи на расчетный счет,

2. cashbox\_transactions — платежи наличными в кассу организации,

3. paypal\_transactions — платежи через сервис PayPal.

Получите общую сумму всех платежей клиентов. Колонку с результатом назовите all\_money.

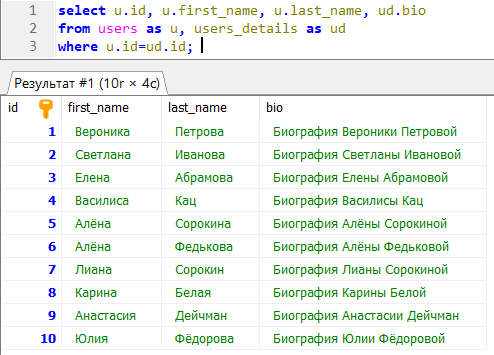


9.4 ОТНОШЕНИЕ ОДИН К ОДНОМУ

Задача 9.4.1 Биографии пользователей

В таблице users хранится базовая информация о пользователях, а в таблице users\_details — подробная. Таблицы связаны отношением один-к-одному так, что первичные ключи в таблицах совпадают.

Получите id пользователя, его имя, фамилию и биографию в одном запросе.



Задача 9.4.2 Интернет-магазин красок

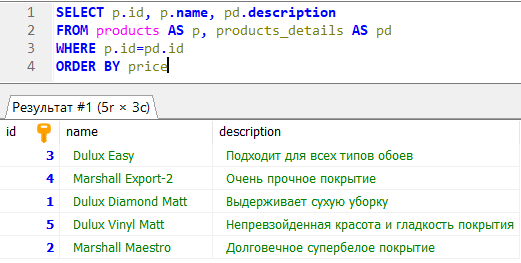
В базе данных интернет-магазина по продаже красок есть две таблицы: products и products\_details.

В первой хранится общая информация о краске: название, количество, цена, категория. А в таблице products\_details содержатся дополнительные текстовые данные: область применения, описание, схема распыления.

Таблицы связаны отношением один-к-одному, но не через первичные ключи, а альтернативным способом.

Первичный ключ id в таблице products связан с колонкой product\_id в таблице products\_details.

Получите в одном запросе id, название товара, а также всю информацию из таблицы products\_details кроме id и product\_id. Данные отсортируйте по цене.

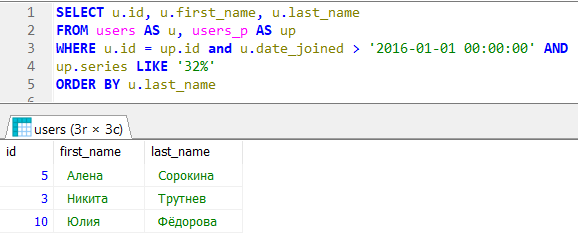


Задача 9.4.3 Персональные данные

Регистрационные данные пользователей хранятся в таблице users, а персональные данные в users\_p. Таблицы связаны друг с другом отношением один-к-одному так, что первичные ключи совпадают.

Получите идентификатор, имя и фамилию для пользователей зарегистрировавшихся (date\_joined) после полуночи 1 января 2016 года с паспортами серия которых начинается на 32.

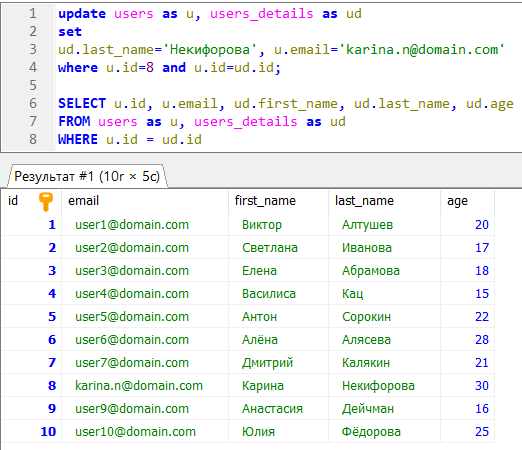
Отсортируйте данные по фамилии пользователей.



Задача 9.4.4 Обновление пользователя

В таблице users хранится базовая информация о пользователях, а в таблице users\_details — подробная. Таблицы связаны отношением один-к-одному так, что первичные ключи в таблицах совпадают.

Обновите данные о пользователе c id равным 8: измените email на «karina.n@domain.com», а фамилию на «Некифорова». Выполните все изменения в одном SQL запросе.



Задача 9.4.5 Неактивные товары

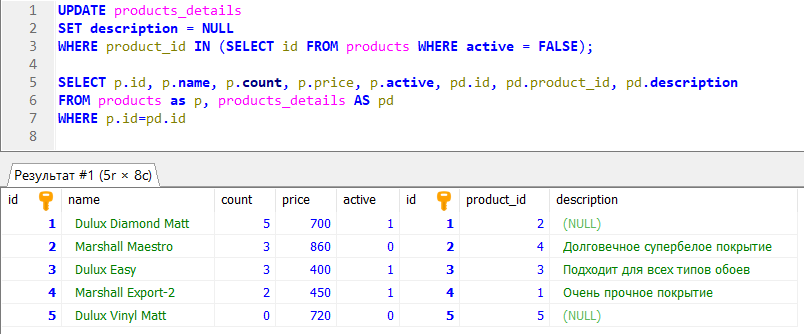
В базе данных интернет-магазина по продаже красок есть две таблицы: products и products\_details.

В первой хранится общая информация о краске: название, количество, цена, категория.

А в таблице products\_details содержатся дополнительные текстовые данные: область применения, описание, схема распыления.

Таблицы связаны отношением один-к-одному. Первичный ключ id в таблице products связан с колонкой product\_id в таблице products\_details.

Очистите поле description для всех неактивных товаров. Используйте один SQL-запрос.



Задача 9.4.6 Удаление товаров

В базе данных интернет-магазина по продаже красок есть две таблицы: products и products\_details.

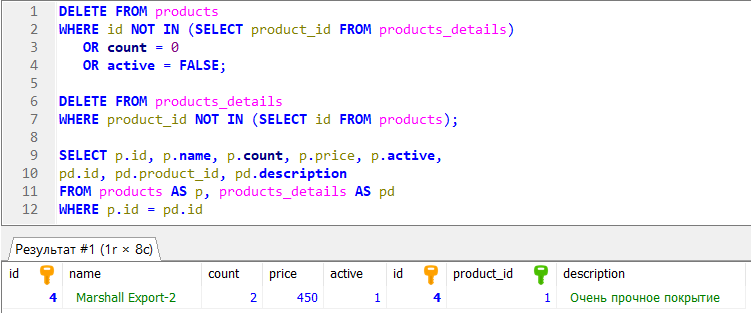
В первой хранится общая информация о краске: название, количество, цена, категория.

А в таблице products\_details содержатся дополнительные текстовые данные: область применения, описание, схема распыления.

Таблицы связаны отношением один-к-одному. Первичный ключ id в таблице products связан с колонкой product\_id в таблице products\_details, а также для него указано ON DELETE CASCADE.

Удалите из обеих таблиц все товары, которых нет в наличии или которые неактивны.

Используйте один SQL-запрос.



Задача 9.4.7 Биографии неактивных пользователей

Иногда нужно удалить данные из одной таблицы, основываясь на условиях другой. Для этого в конструкции DELETE предусмотрен специальный оператор USING.

Разберем на примере ниже, как он работает.

1. В первой строке мы пишем из какой таблицы будем удалять — table2.

2. Затем с помощью USING и JOIN указываем, что эта таблица связана с таблицей table1.

3. Далее идет блок WHERE, где мы сперва задаем связь между двумя таблицами — через их id.

4. И в конце мы указываем условие удаления table1.id < 10.

Получается, что мы удаляем данные из таблицы table2, но по условию в таблице table1.

DELETE FROM table2

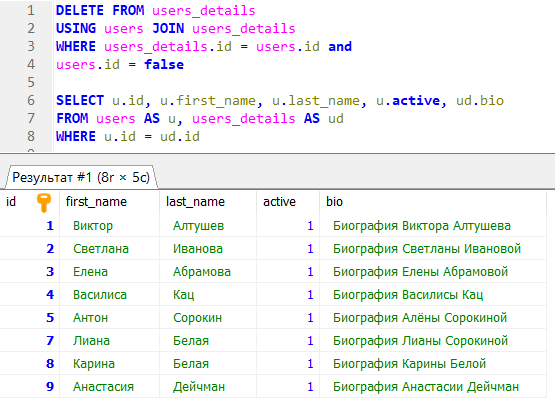
USING table1 JOIN table2

WHERE

table2.id = table1.id AND

table1.id < 10;

Пользуясь этим шаблоном, удалите из таблицы users\_details биографии неактивных пользователей.



9.5 ВНЕШНИЙ КЛЮЧ

Задача 9.5.1 Пользовательские данные

В базе данных есть таблица users, которая была создана с помощью следующего запроса:

CREATE TABLE users (

id INT UNSIGNED NOT NULL PRIMARY KEY AUTO\_INCREMENT,

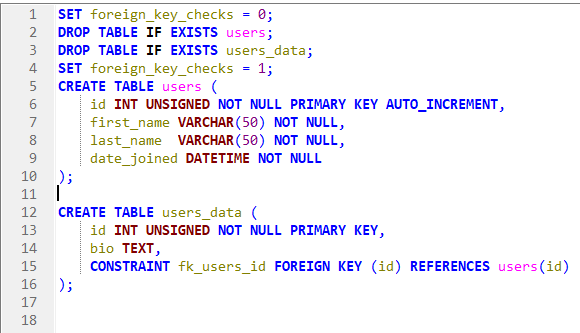
first\_name VARCHAR(50) NOT NULL,

last\_name VARCHAR(50) NOT NULL,

date\_joined DATETIME NOT NULL

);

Создайте таблицу users\_data, в которой первичный ключ id будет также внешним ключом с ссылкой на таблицу users. Помимо первичного ключа, таблица должна содержать поле bio типа TEXT.



Задача 9.5.2 Информация о краске

В базе данных интернет-магазина по продаже красок есть таблица **products**, которая была создана с помощью следующего запроса:

CREATE TABLE products (

id INT UNSIGNED NOT NULL PRIMARY KEY AUTO\_INCREMENT,

name VARCHAR(50) NULL,

count INTEGER NULL,

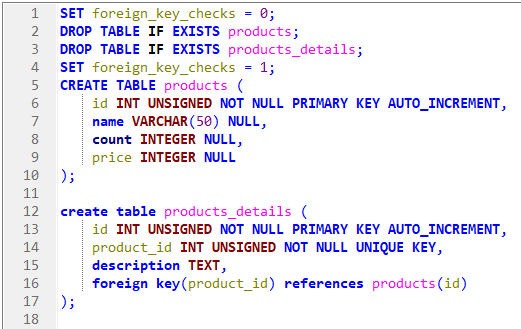
price INTEGER NULL

);

Создайте таблицу **products\_details**, которая будет содержать дополнительные текстовые данные о краске и связана с таблицей **products** отношением один-к-одному.

В таблице должно быть три поля: id, product\_id и description.  
Поле **product\_id** должно быть внешним ключом, который ссылается на поле id в таблице **products**. (обратите внимание, что внешний ключ должен быть точно такого же типа, что и поле, на которое он ссылается)

Поле **description** должно быть типа TEXT.



Задача 9.5.3 Новый пользователь

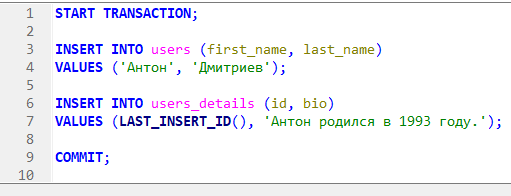
В таблице users хранится базовая информация о пользователях, а в таблице users\_details — подробная. Таблицы связаны отношением один-к-одному так, что первичные ключи в таблицах совпадают, но могут быть рассинхронизированы (id в одной таблице могут опережать ключи в другой).

Добавьте нового пользователя со следующими данными:

Имя: Антон

Фамилия: Дмитриев

Биография: «Антон родился в 1993 году.»

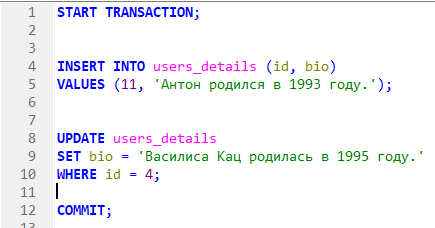


Задача 9.5.4 Обновление и добавление

В таблице users хранится базовая информация о пользователях, а в таблице users\_details — подробная. Таблицы связаны отношением один-к-одному так, что первичные ключи в таблицах совпадают.

Добавьте Антону Дмитриеву следующую биографию: «Антон родился в 1993 году.»

А Василисе Кац измените биографию на «Василиса Кац родилась в 1995 году.»



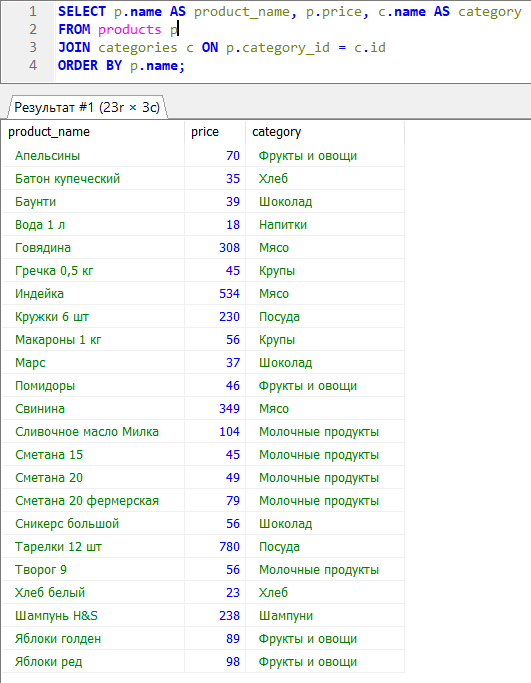
9.6 ОТНОШЕНИЯ ОДИН-КО-МНОГИМ

Задача 9.6.1 Товары с категориями

В таблице products находятся товары, а в categories — категории, к которым они относятся. Поле category\_id является внешним ключом, который указывает на первичный ключ категории.

Получите в одном запросе имя товара, цену товара и имя категории, к которой принадлежит товар.

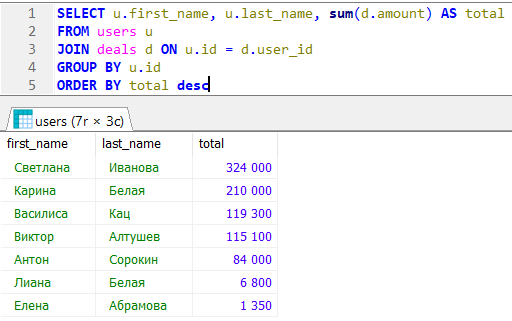
Отсортируйте данные в алфавитном порядке по имени товара.  
Имя категории выведите в колонке с именем category.



Задача 9.6.2 Сделки

В таблице users содержатся данные сотрудников, а в deals сделки этих сотрудников. Таблицы связаны через внешний ключи user\_id, который находится в deals и указывает на столбец id в таблице users.

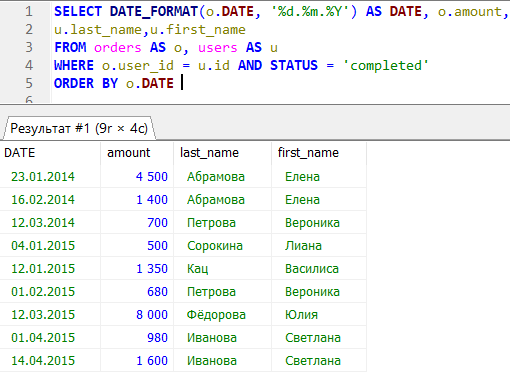
Получите фамилии, имена и суммы сделок всех сотрудников. Вывод отсортируйте по сумме в обратном порядке, а сами значения введите в колонке total.



Задача 9.6.3 Завершенные заказы

Таблицы orders и users связанны отношением многие к одному.  
orders содержит внешний ключ user\_id, который ссылается на id пользователя.

Получите из таблиц дату и стоимость каждого выполненного заказа, а также фамилию и имя пользователя, который этот заказ оформил.  
Информацию отсортируйте по дате заказа, дату выведите в формате ДД.ММ.ГГГГ в поле date

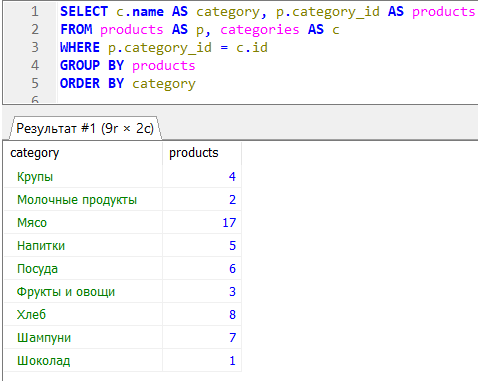


Задача 9.6.4 Количество наименований в категориях

В таблице products хранятся товары, а в categories — категории, к которым они относятся. Поле category\_id является внешним ключом, который указывает на первичный ключ категории.

Получите список категорий с количеством наименований товаров в каждой.

Данные отсортируйте в алфавитном порядке по имени категории.  
Имя категории выведите в поле category, а количество товаров в products.



Задача 9.6.5 Расписание врача

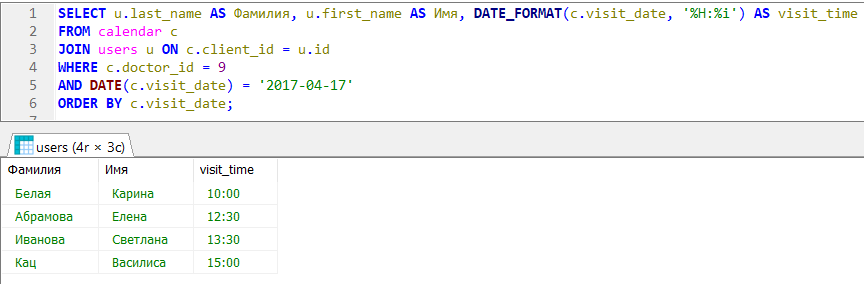
В таблице users хранятся данные по всем пользователям сайта поликлиники: клиентам и докторам. Доктора отмечены в поле is\_doctor.  
В таблице calendar содержатся данные о времени записи (visit\_date) пациентов (client\_id) к докторам (doctor\_id).

И doctor\_id и client\_id ссылаются на поле id в таблице users.

Составьте расписание для доктора Анастасии Дейчман (id равен 9) на 17 апреля 2017 года.

Выведите фамилию пациента, имя пациента и время приема в формате ЧЧ:ММ.

Время приема выведите в колонке visit\_time. Данные должны быть отсортированы в хронологическом порядке.



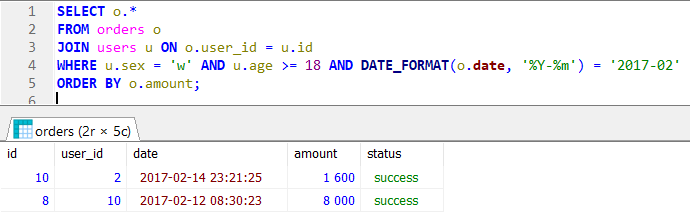
Задача 9.6.6 Февральские заказы

Таблицы orders и users связанны отношением многие к одному.  
orders содержит внешний ключ user\_id, который ссылается на id пользователя.

Получите все заказы, которые сделали женщины от 18 лет и старше в феврале 2017 года.

Выводить нужно только столбцы таблицы orders.

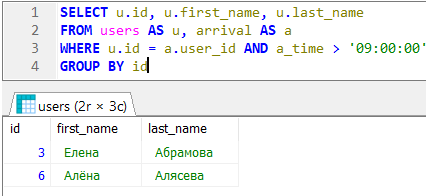
Данные отсортируйте по стоимости заказа.



Задача 9.6.7 Опоздавшие сотрудники

В таблице users хранится информация о сотрудниках компании, а в arrival содержатся данные о дате (a\_date) и времени (a\_time) прихода сотрудников на работу.

Сотрудники, которые пришли на работу после 9 часов утра, считаются опоздавшими. Выведите id, фамилию и имя сотрудников, которые опоздали хотя бы раз в марте 2017 года.

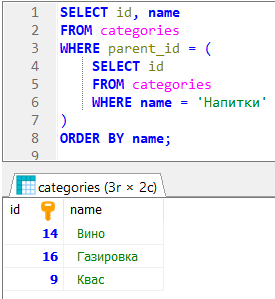


Задача 9.6.8 Напитки

Один из популярных приемов организации иерархии — это создание внешнего ключа, который ссылается на ту же таблицу, в которой он находится.

Например, в таблице categories есть поле parent\_id, которое указывает на id. Если посмотреть на записи с id равным 11 (Кофе) и 13 (Чай), то увидим, что parent\_id для них равен 10 — Кофе и чай. Сама запись с id равным 10, в parent\_id содержит NULL, то есть не имеет родительской категории или, другими словами, находится на верхнем уровне иерархии.

Получите из таблицы categories все категории, у которых родительской является «Напитки». Выведите только id и название категорий. Данные отсортируйте в алфавитном порядке.

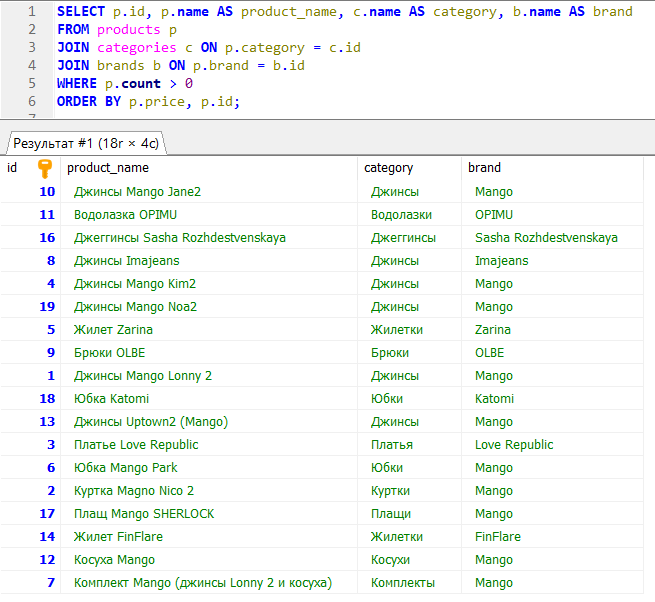


Задача 9.6.9 Товары с категориями и брендами

В базе данных есть три таблицы: products, categories и brands.

Выведите id и название товара, название категории и название бренда для товаров, которые есть на складе.

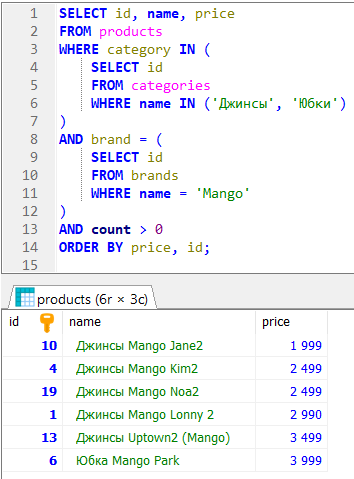
Название товара выведите в поле name, категорию в category, бренда в brand. Отсортируйте данные по цене, а затем по id товара.



Задача 9.6.10 Джинсы и юбки, 2

В базе данных есть три таблицы: products, categories и brands.

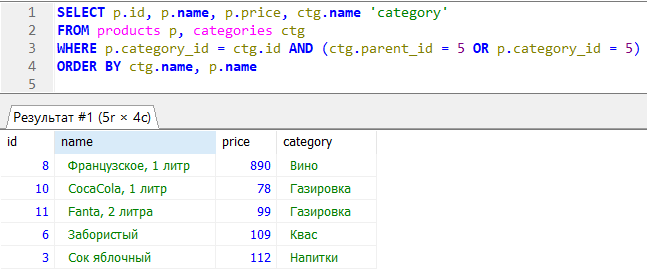
Получите из таблицы products все джинсы и юбки компании Mango.  
Выведите только id, название и цену. Данные отсортируйте по цене, а затем по id. Не выводите товары, которых нет в наличии.



Задача 9.6.11 Напитки, часть 2

Получите все товары из категории «Напитки» и всех её подкатегорий.  
Выведите id, имя и стоимость товара, а также название категории.  
Название категории выведите в колонке category.

Данные отсортируйте по названию категории, а затем по имени товара.

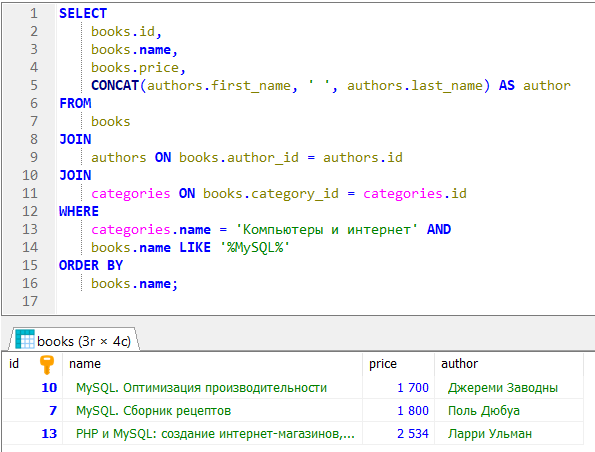


Задача 9.6.12 Книги по MySQL

Пользователь книжного интернет-магазина решил найти все книги по MySQL. Помогите ему.

В каждой строке итоговой таблицы выведите id, название и стоимость книги, а также автора: имя и фамилию в одном поле с названием author.

Данные отсортируйте по названию книг.



9.7 СОЗДАНИЕ СВЯЗЕЙ ОДИН-КО-МНОГИМ

Задача 9.7.1 Связь жанров и исполнителей

В базе данных есть две таблицы: **genres** и **artists**. В первой находится список жанров, а во второй — исполнители. Сейчас таблицы никак не связаны между собой.

Создайте в таблице **artists** новое поле **genre\_id** и сделайте его внешним ключом на поле **id** в таблице **genres**.

Исходные запросы для создания таблиц были такие:

CREATE TABLE genres (

id INT UNSIGNED NOT NULL PRIMARY KEY AUTO\_INCREMENT,

name VARCHAR(50) NULL

);

CREATE TABLE artists (

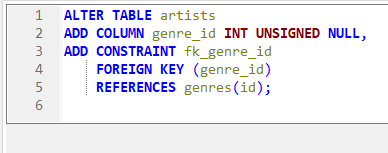
id INT UNSIGNED NOT NULL PRIMARY KEY AUTO\_INCREMENT,

name VARCHAR(50) NULL,

is\_group BOOLEAN NULL

);

Учтите, что обе таблицы уже заполнены данными.



Задача 9.7.2 Товары и категории

В базе данных есть две таблицы: **products** и **categories**. В первой находится список товаров, а во второй — категории. Сейчас таблицы связаны между собой через поле **category\_id**, однако это поле не является настоящим внешним ключом.

Сделайте **category\_id** внешним ключом на поле **id** в таблице **categories**.

Исходные запросы для создания таблиц были такие:

CREATE TABLE products (

id INT UNSIGNED NOT NULL PRIMARY KEY AUTO\_INCREMENT,

name VARCHAR(50) NULL,

count INTEGER NULL,

price INTEGER NULL,

category\_id INTEGER UNSIGNED NOT NULL

);

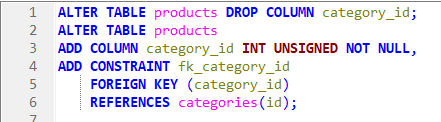
CREATE TABLE categories (

id INT UNSIGNED NOT NULL PRIMARY KEY AUTO\_INCREMENT,

name VARCHAR(50) NULL

);

Учтите, что обе таблицы уже заполнены данными.



Задача 9.7.3 Календарь врача

В таблице users хранятся данные по всем пользователям сайта поликлиники: клиентам и докторам. Доктора отмечены в поле is\_doctor. Создайте таблицу calendar для хранения записей пациентов к врачам. Таблица должна иметь 4 поля:

* id – идентификатор записи.
* doctor\_id – внешний ключ на колонку id в таблице users, NULL запрещен.
* client\_id – также внешний ключ на колонку id в таблице users, NULL запрещен.
* visit\_date – дата и время визита, NULL запрещен.

Таблица users была создана с помощью запроса:

CREATE TABLE users (

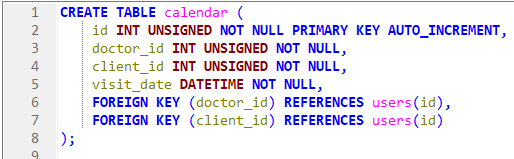
id INT UNSIGNED NOT NULL PRIMARY KEY AUTO\_INCREMENT,

first\_name VARCHAR(50) NULL,

last\_name VARCHAR(50) NULL,

is\_doctor BOOLEAN NULL

);

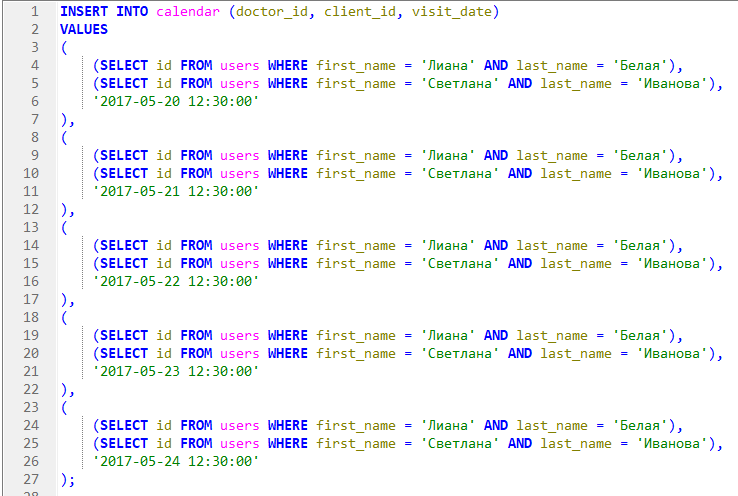


Задача 9.7.4 Запись на массаж

В таблице users хранятся данные по всем пользователям сайта поликлиники: клиентам и докторам. Доктора отмечены в поле is\_doctor.  
В таблице calendar содержатся данные о времени записи (visit\_date) пациентов (client\_id) к докторам (doctor\_id).

И doctor\_id и client\_id ссылаются на поле id в таблице users.

Запишите Светлану Иванову к массажисту Лиане Белой.  
Добавьте 5 ежедневных сеансов начиная с 20 мая 2017 года.  
Каждый сеанс начинается в 12:30.



9.8JOIN

Задача 9.8.1 Первый JOIN

Перепишите SQL-запрос с помощью конструкции JOIN:

SELECT

p.id,

p.name,

p.price,

cats.name as category

FROM

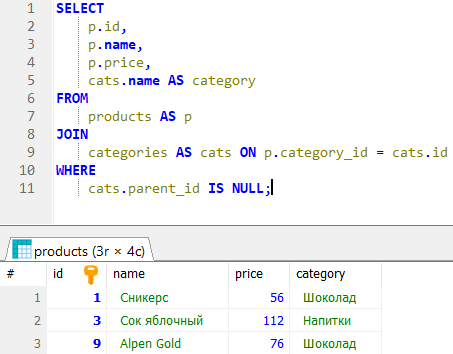
products as p,

categories as cats

WHERE

p.category\_id = cats.id AND

cats.parent\_id IS NULL



Задача 9.8.2 Исполнители и жанры

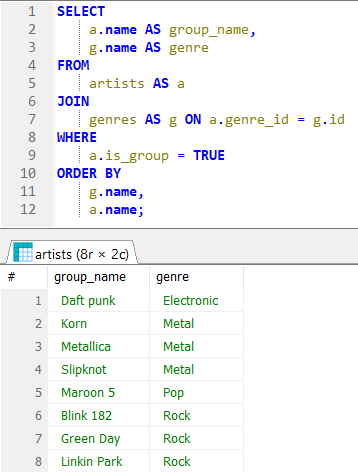
В таблицеartistsпредставлены исполнители, а в таблицеgenres—

жанры в которых они играют.

Получите названия всех групп и жанры, в которых они играют.

Жанры выведите в полеgenre

Отсортируйте данные по имени жанра, а затем по названию группы.



Задача 9.8.3 Количество исполнителей

Получите id и названия жанров, а также количество исполнителей в

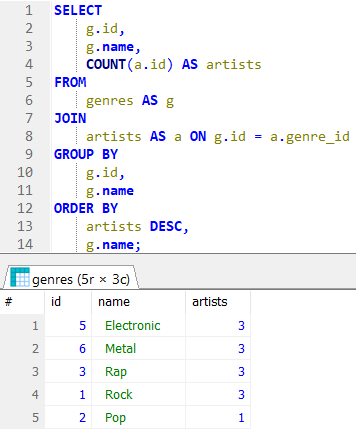
них.

Количество исполнителей выводите в полеartists, а названия жанров вname.

Жанры, в которых нет исполнителей, выводить не надо.

Данные отсортируйте сперва по количеству исполнителей в обратном

порядке, а потом по названию жанра.



Задача 9.8.4 Города-миллионники

Получите список городов-миллионников в Европе.

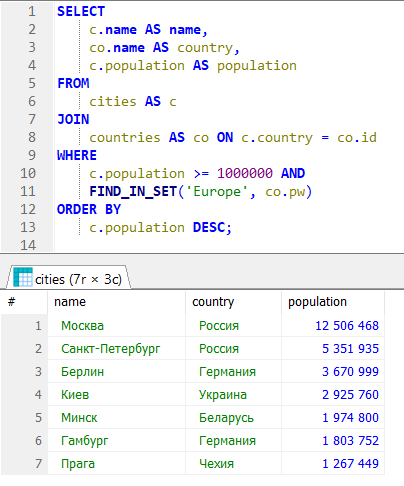
Выведите название города,название страны и население города.

Название города выведите в поле name, страны в поле country,​​​​​​а

население в population.

Данные отсортируйте по населению в обратном порядке (крупные

города выше).

Поле pw в таблице countries представлено типом SET.  


Задача 9.8.5 Активные категории

На сайте интернет-магазина нужно выводить только те категории, в

которых есть товары.

Получите имена категорий, в которых есть товары, а также количество

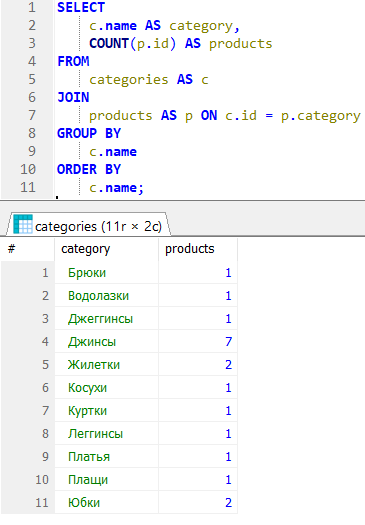
товаров в каждой категории.

Данные выведите в двух колонках:

1.categoryс именем категории;

2.productsс количеством товаров.

Отсортируйте итоговую таблицу по имени категории.



Задача 9.8.6 Новый жанр и исполнитель

Добавьте в таблицы новый жанр «Rap» и исполнителя «Eminem» в этом

жанре. 

9.9ПОНИМАНИЕ JOIN

Задача 9.9.1 Сотрудники и должности

В таблице rolesх ранится список ролей (должностей) в компании, а

В employees — сотрудники с указанием их роли через поле role\_id.

Некоторые сотрудники не имеют ролей.

Получите список всех активных сотрудников с указанием их

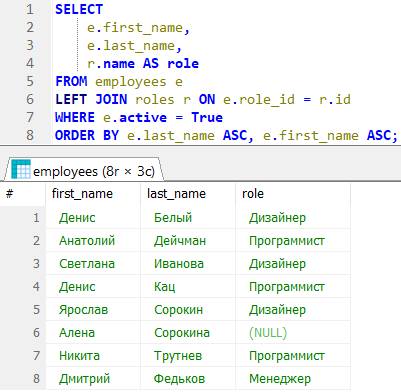
должностей.

Если у сотрудника нет должности, то нужно вывести NULL.

Выведите 3 столбца: имя, фамилию и должность (role).

Данные отсортируйте по фамилии, а после по имени в алфавитном

порядке.



Задача 9.9.2 Наполненность жанров

В таблицеartistsпредставлены исполнители, а в таблицеgenres–

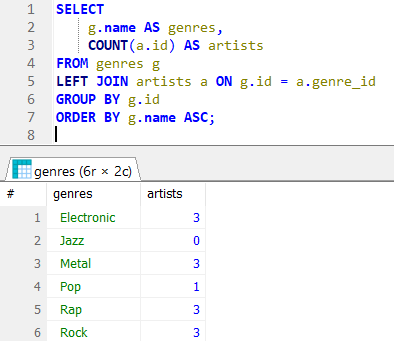
жанры в которых они играют.

Получите названия всех жанров и количество исполнителей в каждом

жанре.

Жанры выведите в полеgenres, а количество исполнителей вartists.

Отсортируйте данные по названию жанра.



Задача 9.9.3 Все категории

На сайте интернет-магазина нужно выводить все категории, а не только

те, в которых есть товары.

Получите имена категорий, а также количество товаров в каждой

категории.

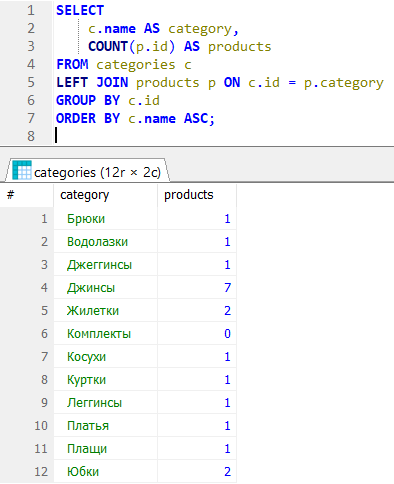
Данные выведите в двух колонках:

1.categoryс именем категории;

2.productsс количеством товаров.

Если для какой-то из категорий выводится NULL, то следует выводить

3. Отсортируйте итоговую таблицу по имени категории.



Задача 9.9.4 Товары с категориями

В таблице products находятся товары, а в categories– категории, к

которым они относятся. Поле category\_id является внешним ключом,

который указывает на первичный ключ категории.

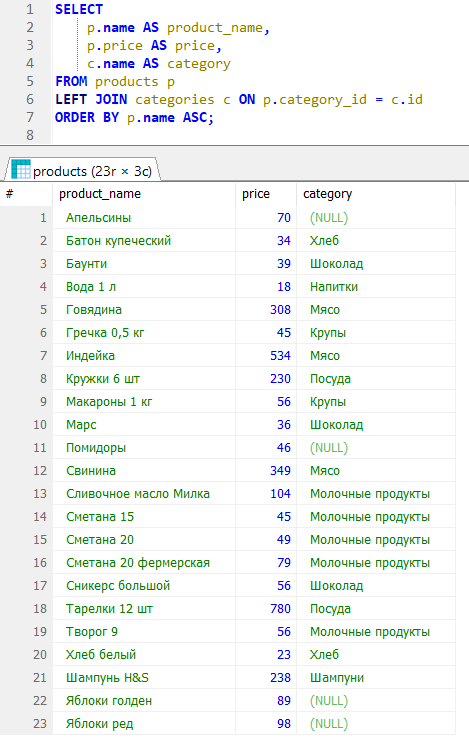
Получите в одном запросе имя товара, цену товара и имя категории, к

которой принадлежит товар.

Отсортируйте данные в алфавитном порядке по имени товара.

Имя категории выведите в колонке с именем category.

Если у товара нет категории, то выведите NULL.



Задача 9.9.5 Структура компании

В таблице roles хранится список ролей (должностей) в компании, а

В employees— сотрудники с указанием их роли через поле role\_id.

Получите список всех должностей и количество активных сотрудников

для каждой должности.

Выведите 2 столбца: название должности (name) и количество

сотрудников (employees).

Данные сперва отсортируйте по числу сотрудников в обратном

порядке, а затем по имени роли.

Данную задачу лучше решать в два этапа:

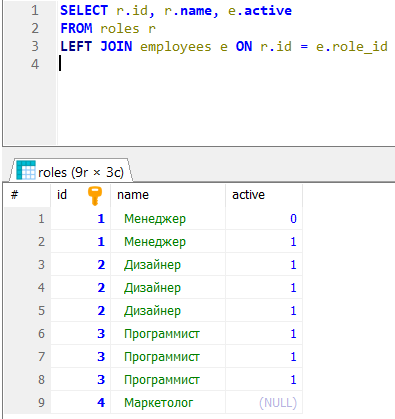
Сперва соединить таблицы с помощью JOIN и проанализировать

данные.

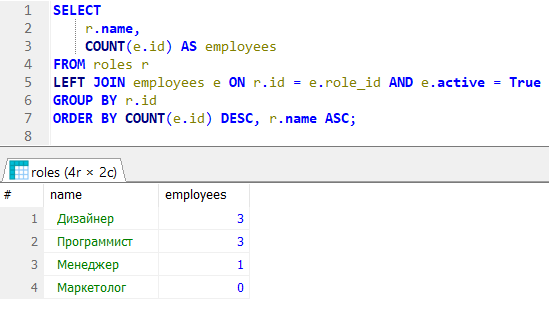
Затем уже применять GROUP BY, COUNT или SUM с

Правильным WHERE.

1-й этап:



2-й этап:



9.10 ПОНИМАНИЕ JOIN, ЧАСТЬ 2

Задача 9.10.1 Книги и писатели

Получите все данные о книгах и писателях. Свяжите книги с их

авторами если это возможно.

Выведите 5 столбцов:

author\_id– id автора, NULL для книг, у которых автор не указан;

book\_id– id книги, которую написал автор, NULL — если у

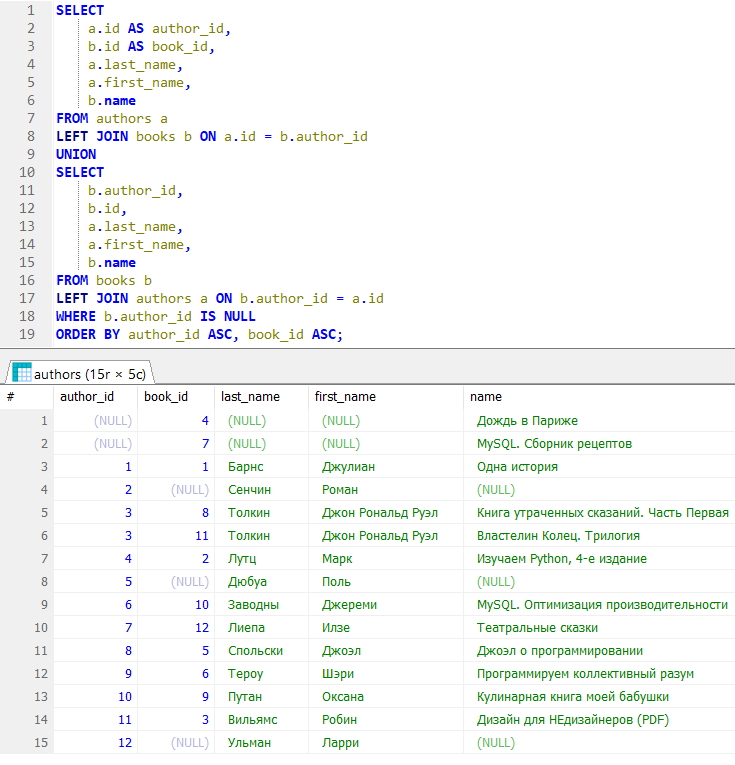
автора нет книг;

last\_name– фамилия автора;

first\_name– имя автора;

name– название книги.

Данные отсортируйте сперва по id авторов, а затем по id книг.



Задача 9.10.2 Страны и города

Получите все страны и города из таблиц countries и cities.

Свяжите города со странами, в которых они находятся.

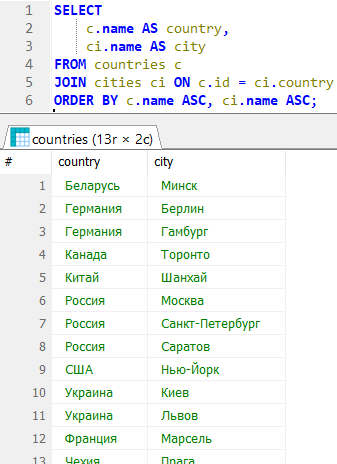
Выведите 2 столбца:

1.country— название страны;

2.city— название города.

Данные отсортируйте сперва по названию страны, а затем по названию

города.



9.11ВЫБОРКА ИЗ ТРЕХ И БОЛЕЕ ТАБЛИЦ

Задача 9.11.1 Пользователи, роли, отделы

Получите список всех пользователей (users) с указанием их ролей

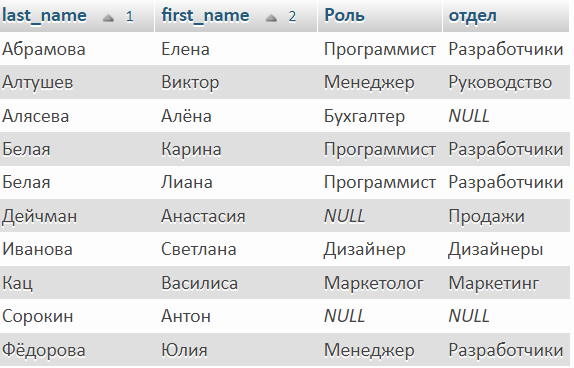
(roles) и отделов (departments).

Выведите 4 поля:last\_name,first\_name,role(название роли)

иdepartment(название отдела).

Данные отсортируйте сперва по фамилии, а затем по имени.

[select](http://localhost/phpMyAdmin/url.php?url=https://dev.mysql.com/doc/refman/5.7/en/select.html) u.last\_name, u.first\_name, r.name as role, d.name as department from users as u [left](http://localhost/phpMyAdmin/url.php?url=https://dev.mysql.com/doc/refman/5.7/en/string-functions.html%23function_left) join roles as r on r.id=u.role\_id [left](http://localhost/phpMyAdmin/url.php?url=https://dev.mysql.com/doc/refman/5.7/en/string-functions.html%23function_left) join departments as d on d.id=u.department\_id order by last\_name, first\_name;



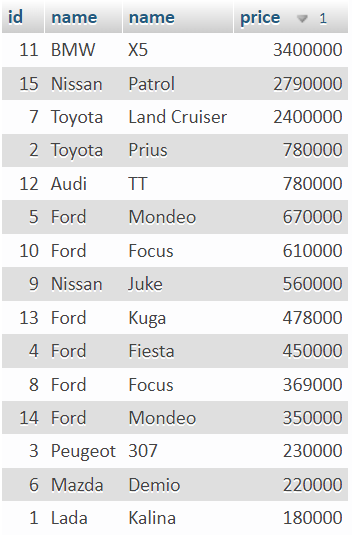
Задача 9.11.2 Машины автосалона

База данных автосалона содержит 3 таблицы: marks, models и cars.

Таблица models связана с marks через поле mark\_id, а таблица cars с models через поле model\_id. То есть cars напрямую не связана с marks, однако эту связь можно проследить через models.

Получите все автомобили салона с указанием модели и марки. Выведите следующие поля: id автомобиля, марку автомобиля (поле mark), модель автомобиля (поле model) и его цену. Данные отсортируйте по цене в обратном порядке.

SELECT cars.id,marks.name,models.name,price from cars join models on models.id=model\_id join marks on marks.id=mark\_id order by price desc



Задача 9.11.3 Стоимость по маркам

База данных автосалона содержит 3 таблицы: marks, models и cars.

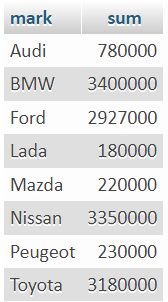
Получите все марки представленные в салоне, а также суммарную стоимость автомобилей каждой марки. Выведите 2 колонки:

1. mark — название марки;

2. sum — стоимость всех автомобилей этой марки.

Данные отсортируйте по названию марки.

[select](http://localhost/phpMyAdmin/url.php?url=https://dev.mysql.com/doc/refman/5.7/en/select.html) marks.name as mark,[sum](http://localhost/phpMyAdmin/url.php?url=https://dev.mysql.com/doc/refman/5.7/en/aggregate-functions.html%23function_sum)(price) as [sum](http://localhost/phpMyAdmin/url.php?url=https://dev.mysql.com/doc/refman/5.7/en/aggregate-functions.html%23function_sum) from cars join models on model\_id=models.id join marks on mark\_id=marks.id GROUP by marks.name order by marks.name;



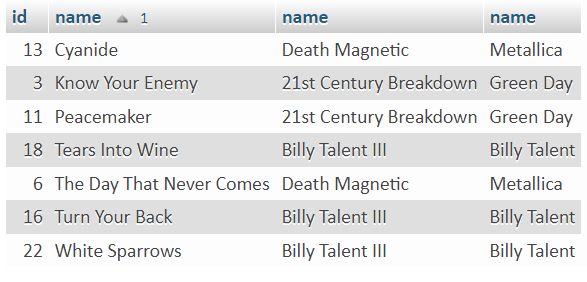
Задача 9.11.4 Музыкальный сайт

База данных музыкального сайта состоит из 4 таблиц: genres, artists, albums и songs.

Получите все песни в жанрах Rock и Metal с 2008 по 2010 год включительно.

Выведите id песни, её название, а также название альбома в поле album и название исполнителя в поле artist. Данные отсортируйте по названию композиции.

[select](http://localhost/phpMyAdmin/url.php?url=https://dev.mysql.com/doc/refman/5.7/en/select.html) songs.id,songs.name,albums.name,artists.name from genres join artists on genre\_id=genres.id join albums on artists.id=artist\_id join songs on album\_id=albums.id where [year](http://localhost/phpMyAdmin/url.php?url=https://dev.mysql.com/doc/refman/5.7/en/date-and-time-types.html)>=2008 [and](http://localhost/phpMyAdmin/url.php?url=https://dev.mysql.com/doc/refman/5.7/en/logical-operators.html%23operator_and) [year](http://localhost/phpMyAdmin/url.php?url=https://dev.mysql.com/doc/refman/5.7/en/date-and-time-types.html)<=2010 [and](http://localhost/phpMyAdmin/url.php?url=https://dev.mysql.com/doc/refman/5.7/en/logical-operators.html%23operator_and) (genres.name="Rock" [or](http://localhost/phpMyAdmin/url.php?url=https://dev.mysql.com/doc/refman/5.7/en/logical-operators.html%23operator_or) genres.name="Metal") order by songs.name;



Задача 9.11.5 Количество композиций

База данных музыкального сайта состоит из 4 таблиц: genres, artists, albums и songs.

Получите жанры и количество композиций в каждом из них.  
Жанры выведите в поле name, а количество композиций в songs.  
Данные отсортируйте по имени жанра.

Жанры, в которых нет композиций, выводить не надо.

Задача 9.11.6 Отчет о звонках

В CRM системе компании есть 4 таблицы:

1. managers – менеджеры (сотрудники) компании;

2. companies – контрагенты компании;

3. clients – сотрудники контрагентов;

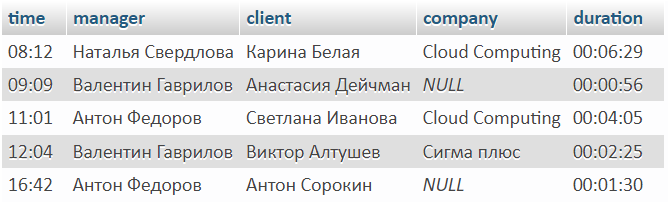
4. calls – звонки сотрудников контрагентов менеджерам компании. В таблице также фиксируется дата (date) и продолжительность (duration\_sec) звонка в секундах.

Получите данные о звонках за 5 апреля 2018 года.  
Выведите 5 колонок:

1. time – время звонка в формате ЧЧ:ММ;
2. manager – имя и фамилия менеджера, который принял звонок;
3. client – имя и фамилия сотрудника контрагента, который позвонил;
4. company – название компании контрагента, NULL если названия нет;
5. duration – продолжительность звонка в формате Ч:ММ:СС.

Данные отсортируйте по дате и времени звонка.

[select](http://localhost/phpMyAdmin/url.php?url=https://dev.mysql.com/doc/refman/5.7/en/select.html) time\_format(calls.date, '%H:%i') as [time](http://localhost/phpMyAdmin/url.php?url=https://dev.mysql.com/doc/refman/5.7/en/date-and-time-types.html), concat(m.first\_name, ' ', m.last\_name) as manager, concat(cl.first\_name, ' ', cl.last\_name) as client, com.name as company, SEC\_TO\_TIME(calls.duration\_sec) as duration from calls join managers as m on calls.manager\_id=m.id join clients as cl on calls.client\_id=cl.id [left](http://localhost/phpMyAdmin/url.php?url=https://dev.mysql.com/doc/refman/5.7/en/string-functions.html%23function_left) join companies as com on com.id=cl.company\_id where [date](http://localhost/phpMyAdmin/url.php?url=https://dev.mysql.com/doc/refman/5.7/en/date-and-time-types.html)(calls.date)='2018-04-05' order by calls.date;



Задача 9.11.7 Продолжительность звонков

В CRM системе компании есть 4 таблицы:

1. managers — менеджеры (сотрудники) компании;

2. companies — контрагенты компании;

3. clients — сотрудники контрагентов;

4. calls — звонки сотрудников контрагентов менеджерам компании. В таблице также фиксируется дата (date) и продолжительность (duration\_sec) звонка в секундах.

Получите список всех компаний и общую продолжительность звонков по каждой из них.

Выведите 2 колонки:

1. company — название компании;
2. duration — общая продолжительность звонков в формате Ч:ММ:СС.

Данные отсортируйте по продолжительности разговоров.

Задача 9.11.8 Отчет по менеджерам

В CRM системе компании есть 4 таблицы:

1. managers — менеджеры (сотрудники) компании;

2. companies — контрагенты компании;

3. clients — сотрудники контрагентов;

4. calls — звонки сотрудников контрагентов менеджерам компании. В таблице также фиксируется дата (date) и продолжительность (duration\_sec) звонка в секундах.

Получите среднее время общения менеджеров с сотрудниками Cloud Computing. Выведите 3 колонки:

1. first\_name — имя менеджера;
2. last\_name — фамилия менеджера;
3. avg\_duration — средняя продолжительность звонка в формате ЧЧ:ММ:СС.

Данные отсортируйте по средней продолжительности разговоров в обратном порядке.

[Select](http://localhost/phpMyAdmin/url.php?url=https://dev.mysql.com/doc/refman/5.7/en/select.html) managers.first\_name, managers.last\_name, DATE\_FORMAT(sec\_to\_time(IFNULL([avg](http://localhost/phpMyAdmin/url.php?url=https://dev.mysql.com/doc/refman/5.7/en/aggregate-functions.html%23function_avg)(duration\_sec), 0)), "%H:%i:%s") as avg\_duration from managers JOIN calls on calls.manager\_id = managers.id [Left](http://localhost/phpMyAdmin/url.php?url=https://dev.mysql.com/doc/refman/5.7/en/string-functions.html%23function_left) JOIN clients on clients.id = calls.client\_id [Left](http://localhost/phpMyAdmin/url.php?url=https://dev.mysql.com/doc/refman/5.7/en/string-functions.html%23function_left) JOIN companies on clients.company\_id = companies.id WHERE companies.id = 2 Group by managers.id Order by avg\_duration desc;



9.12ССЫЛОЧНАЯ ЦЕЛОСТНОСТЬ

Задача 9.12.1 Удаление категории

В таблицеproductsнаходятся товары, а вcategories– категории, к

которым они относятся. Полеcategory\_idявляется внешним ключом,

который указывает на первичный ключ категории. При создании ключа

в запросе указалиON DELETE RESTRICT.

Удалите категории «Молочные продукты» и «Мясо» вместе с

товарами.





Задача 9.12.2 Удаление категорий, часть 2

В таблицеproductsнаходятся товары, а вcategories— категории, к

которым они относятся. Полеcategory\_idявляется внешним ключом,

который указывает на первичный ключ категории. При создании ключа в

запросе указалиON DELETE SET NULL.

Удалите категории «Молочные продукты» и «Мясо» без удаления

товаров.



Задача 9.12.3 Удаление категорий, часть 3

В таблицеproductsнаходятся товары, а вcategories— категории, к

которым они относятся. Полеcategory\_idявляется внешним ключом,

который указывает на первичный ключ категории. При создании ключа в

запросе указалиON DELETE CASCADE.

Удалите категории «Молочные продукты» и «Мясо» вместе с

товарами.



Задача 9.12.4 Перенос и удаление категорий

В таблице products находятся товары, а в categories— категории, к

которым они относятся. Поле category\_id является внешним ключом,

который указывает на первичный ключ категории. При создании ключа

в запросе указали ON DELETE SET NULL.

Перенесите товары из категории «Фрукты и овощи» в категории

«Фрукты» и «Овощи» а затем удалите «Фрукты и овощи»

Также удалите категории «Молочные продукты» вместе с товарами.



9.13ОТНОШЕНИЯ МНОГИЕ КО МНОГИМ

Задача 9.13.1 Программисты

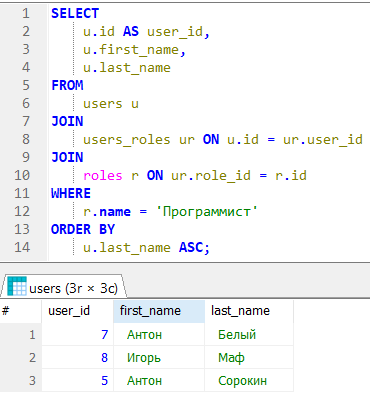
Таблицы users и roles связаны отношением многие ко многим через

Таблицу users\_roles.

Получите всех программистов компании.

Выведите id пользователя, его имя и фамилию. Данные отсортируйте

по фамилии.



Задача 9.13.2 Сотрудники компании

Таблицыusersиrolesсвязаны отношением многие ко многим через

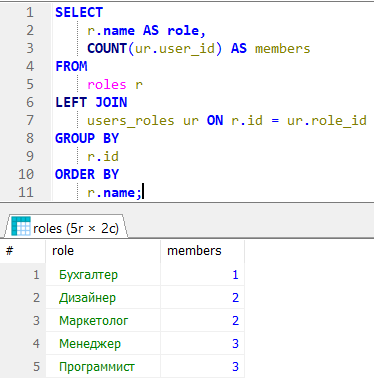
таблицуusers\_roles. Получите список всех ролей и количество сотрудников в

каждой роли.

Выведите два поля:role– название должности иmembers– количество

сотрудников в этой должности. Данные отсортируйте по названию

должности.



Задача 9.13.3 Несколько ролей

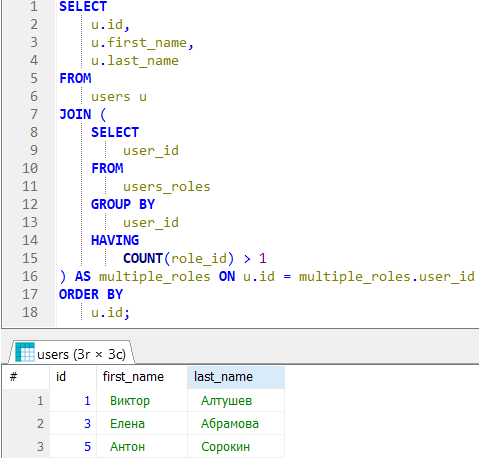
Таблицыusersиrolesсвязаны отношением многие ко многим через

таблицуusers\_roles.

Получите список всех сотрудников, которые выполняют более одной роли.

Выведитеid, имена и фамилии таких сотрудников. Данные отсортируйте

поid.



Задача 9.13.4 Сотрудники без ролей

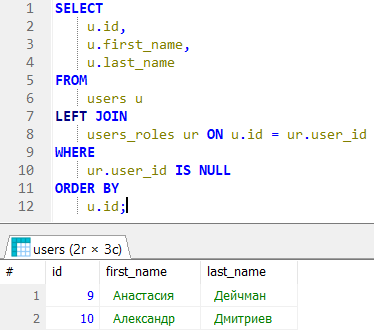
Таблицыusersиrolesсвязаны отношением многие ко многим через

таблицуusers\_roles.

Получите список всех сотрудников, которые не выполняют ни одной

роли. Выведите id, имена и фамилии таких сотрудников. Данные

отсортируйте по id.



Задача 9.13.5 Изменение ролей

Таблицыusersиrolesсвязаны отношением многие ко многим через

таблицуusers\_roles.

В компании произошли небольшие изменения и вам необходимо

внести соответствующие правки в таблицы:

1. Уберите Светлану Иванову из менеджеров.

2. Сделайте менеджером Анастасию Дейчман.

3. Назначьте Александра Дмитриева программистом.







Задача 9.13.6 Команда проекта

Таблицыusers,rolesиprojectsсвязаны отношением многие ко многим

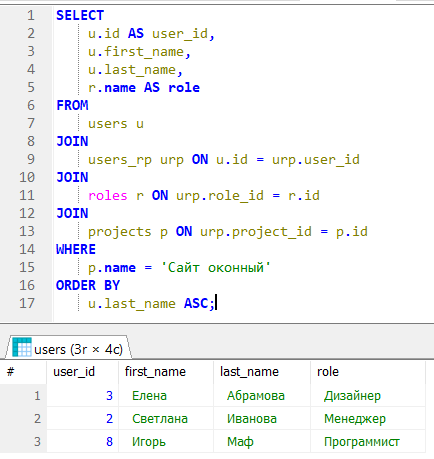
через таблицуusers\_rp. Такая «тройная» связь позволяет задавать роли

пользователей в конкретных проектах.

Получите состав команды с указанием ролей для проекта «Сайт

оконный». Выведите id, имена и фамилии сотрудников, а также их роли в

полеrole. Данные отсортируйте по фамилии.



Задача 9.13.7 Заказы

Рассмотрим базу данных интернет-магазина:

1. Таблицаordersсодержит список заказов пользователей и связана

сusersотношением один ко многим через полеuser\_id.

2. Такжеordersсвязана отношением многие ко многим с

таблицейproductsчерез таблицуorders\_details, что дает возможность

указывать какие именно товары находятся в заказе.Одна запись

вorders\_detailsсоответствует одному заказанному товару.

3. Полеcountв таблицеproductsотвечает за количество товаров на

складе.

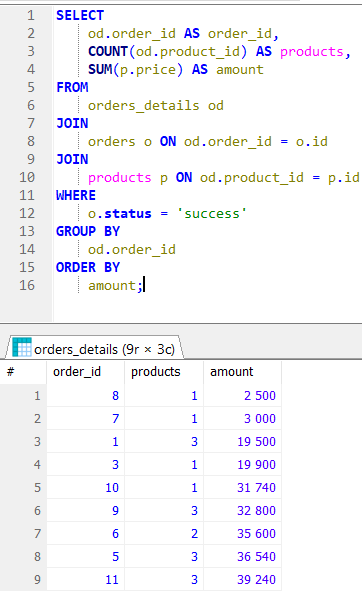
Получите выполненные заказы с указанием количества и стоимости

товаров в них.

Выведите три поля: id заказа, количество товаров в заказе (в

полеproducts) и сумму заказа (в полеamount).

Данные отсортируйте по сумме заказа.



Задача 9.13.8 Битые заказы

Продолжаем работать с базой данных интернет-магазина:

1. Таблицаordersсодержит список заказов пользователей и связана

сusersотношением один ко многим через полеuser\_id.

2. Такжеordersсвязана отношением многие ко многим с

таблицейproductsчерез таблицуorders\_details, что дает возможность

указывать какие именно товары находятся в заказе.

При проектировании программист ошибся и сделал возможность

удаления товаров из таблицыproductsс каскадным удалением из

таблицыorders\_details. И забыл. Через год в магазине каких-то товаров не

стало, и менеджер удалил их из таблицы, потянув за собой все данные о

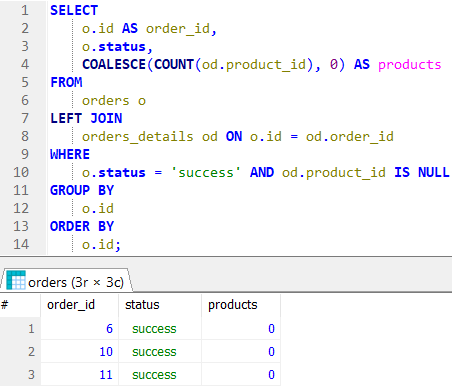
заказе, кроме самого заказа. В итоге у нас есть выполненные заказы без

товаров.

Получите выполненные заказы, в которых нет товаров.

Выведите три поля: id заказа, статус заказа, а также количество товаров в

заказе (в полеproducts).



Задача 9.13.9 Рок и метал композиции

В базе данных музыкального сайта 5 таблиц:genres– для

жанров,artists– для исполнителей,artists\_genres– таблица для связи

исполнителей с жанрами,albums– для альбомов (связаннаяс исполнителями

через поле artist\_id) иsongs– для хранения композиций (связанная с

альбомами через поле album\_id).

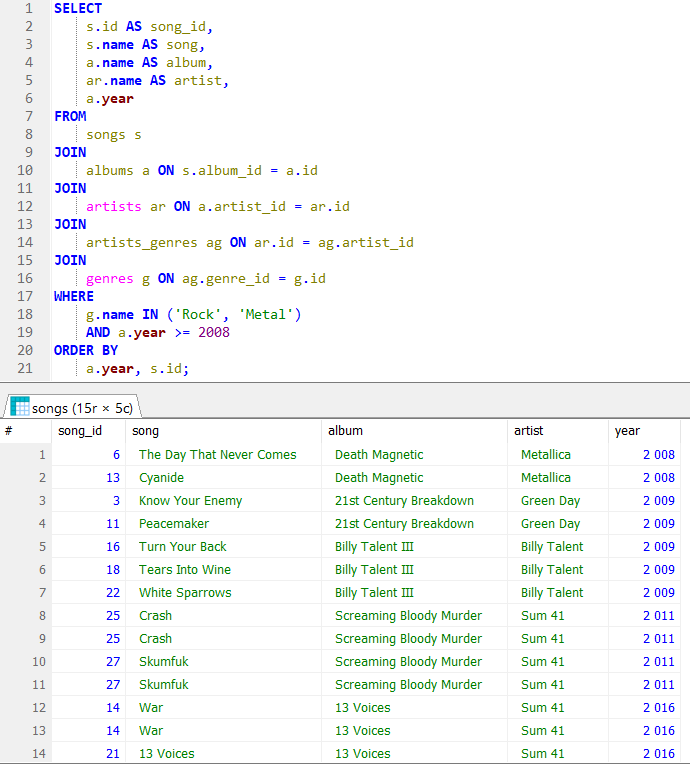
Получите все композиции исполнителей в жанрах Rock и Metal

начиная с 2008 года.

Выведите id и название композиции, альбом (в поле album),

исполнителя (в поле artist), а также год альбома (в поле year).

Данные отсортируйте по году выхода, а затем по id композиции.



Задача 9.13.10 Самые продаваемые товары

Продолжаем работать с базой данных интернет-магазина:

1. Таблицаordersсодержит список заказов пользователей и связана

сusersотношением один ко многим через полеuser\_id.

2. Такжеordersсвязана отношением многие ко многим с

таблицейproductsчерез таблицуorders\_details, что дает возможность

указывать какие именно товары находятся в заказе.

Получите 5 самых продаваемых товаров (тех, которых больше всего

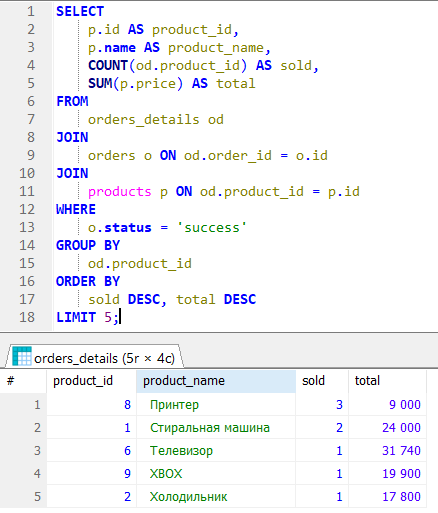
покупали). Учитывайте только выполненные заказы.

Выведите четыре поля: id и название товара, количество проданных товаров

(в полеsold), общую стоимость проданных товаров (в полеtotal).

Данные отсортируйте сперва по полюsoldв обратном порядке, а затем

поtotal, также в обратном порядке.



Задача 9.13.11 Корзина покупок

Продолжаем работать с базой данных интернет-магазина:

1. Таблицаordersсодержит список заказов пользователей и связана

сusersотношением один ко многим через полеuser\_id.

2. Такжеordersсвязана отношением многие ко многим с

таблицейproductsчерез таблицуorders\_details, что дает возможность

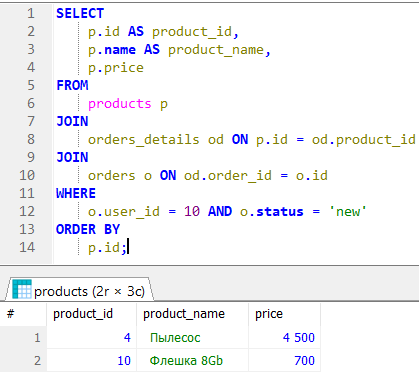
указывать какие именно товары находятся в заказе.

Заказы со статусом «new» считаются корзиной. Получите состав

корзины Александра Дмитриева.

Выведите три поля: id, название и стоимость товара.

Данные отсортируйте по id товара.



Задача 9.13.12 Ценные клиенты

Продолжаем работать с базой данных интернет-магазина:

1. Таблицаordersсодержит список заказов пользователей и связана

сusersотношением один ко многим через полеuser\_id.

2. Такжеordersсвязана отношением многие ко многим с

таблицейproductsчерез таблицуorders\_details, что дает возможность

указывать какие именно товары находятся в заказе.

Получите список из пяти клиентов, которые потратили больше всего

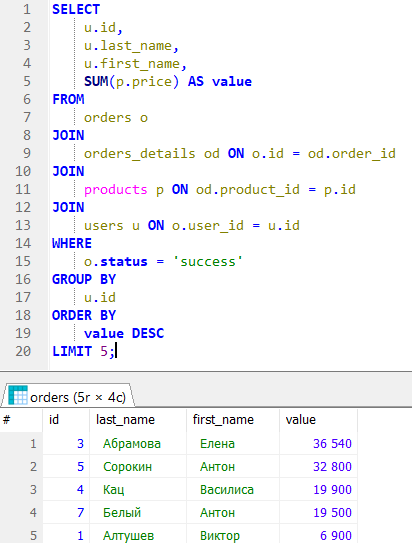
денег в интернет-магазине.

Учитывайте только выполненные заказы.

Выведите четыре поля: id, фамилию и имя клиента, а также потраченную им

сумму (в полеvalue).

Данные отсортируйте по сумме в обратном порядке.



Задача 9.13.13 Меняем корзину

Продолжаем работать с базой данных интернет-магазина:

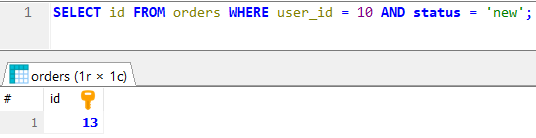
1. Таблица orders содержит список заказов пользователей и связана с users отношением один ко многим через поле user\_id.

2. Также orders связана отношением многие ко многим с таблицей products через таблицу orders\_details, что дает возможность указывать какие именно товары находятся в заказе.

Заказы со статусом «new» считаются корзиной. Измените состав товаров в корзине Александра Дмитриева.

1. Добавьте в корзину холодильник.

2. Удалите из корзины пылесос.







Задача 9.13.14 Обувной магазин

Таблицы ниже содержат информацию о наличии обуви разных моделей, цветов и размеров в магазинах города.

В один из магазинов пришла девушка и решила примерить розовые Туфли Elite (EN1345) 39 размера, однако в этом магазине её размера не было и продавщица подошла к компьютеру, чтобы посмотреть в каких магазинах данная модель есть.

Напишите SQL запрос, который вернет все магазины, в которых есть нужная обувь. В финальной таблице должно быть два столбца: название магазина (shop) и количество пар обуви в нём (quantity).

Данные нужно отсортировать по количеству пар обуви.

