Московский Авиационный Институт

(Национальный Исследовательский Университет)

Институт информационных технологий и прикладной математики

Кафедра вычислительной математики и программирования

**Лабораторная работа по курсу**

**«Базы данных»**

**Тема: Каталог ноутбуков**

Выполнили: Лисин Р.С.,

Мальцев И.Д., Фролов М.А.

Группа: М8О-306Б-20

Преподаватель: Сеченых П.А.

Дата: \_\_\_\_\_\_\_\_\_\_\_

Оценка: \_\_\_\_\_\_\_\_\_\_\_

Подпись: \_\_\_\_\_\_\_\_\_\_\_

Москва, 2022

**Таблицы**

Таблица laptops:

* laptop\_id – идентификатор ноутбука
* laptop\_name – имя ноутбука
* producer – производитель ноутбука
* price – цена ноутбука
* characteric – характеристики ноутбука

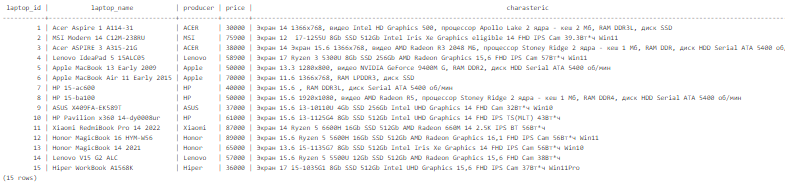


Таблица providers:

* provider\_id – идентификатор поставщика
* provide\_name – название поставщика
* address – адрес поставщика
* phone – телефон поставщика
* reputation – репутация поставщика

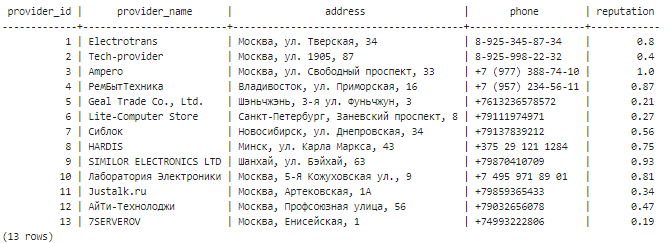


Таблица shops:

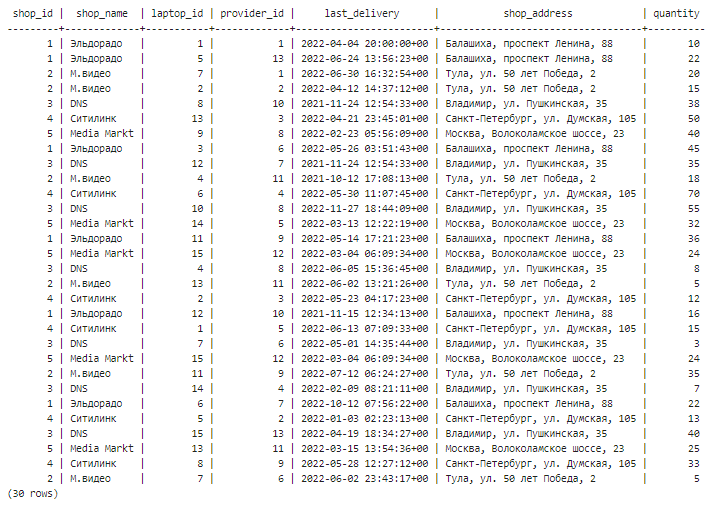
* shop\_id – идентификатор магазина
* last\_delivery – дата последней доставки в магазин данного товара
* shop\_address – адрес магазина
* quantity – количество товара данного типа
* laptop\_id – идентификатор ноутбука
* provider\_id – идентификатор поставщика

Первичные ключи:

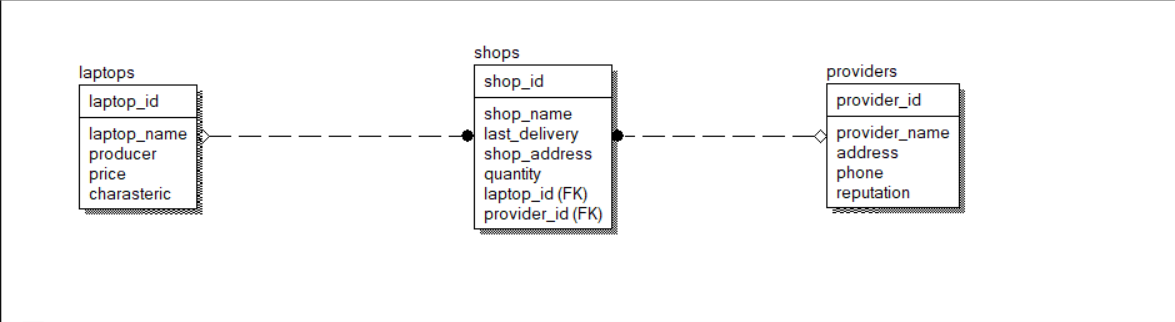
* для таблицы laptops – laptops\_id
* для таблицы providers – provider\_id
* для таблицы shops – shop\_id

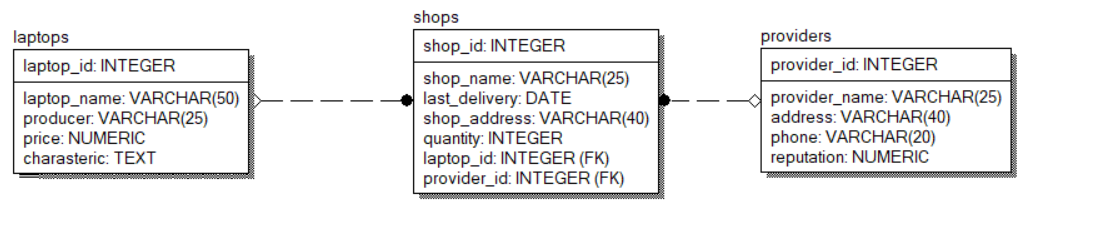
Ключи связи присутствуют только в таблице shops:

* laptop\_id из таблицы laptops
* provider\_id из таблицы providers



**База данных в Erwin**

*Логическая модель:* 

*Физическая модель:* 

*Код для генерации БД:*

CREATE TABLE laptops (

laptop\_id SERIAL PRIMARY KEY,

laptop\_name VARCHAR(50) NOT NULL,

producer VARCHAR(25) NOT NULL,

price NUMERIC NOT NULL,

charasteric TEXT NULL

);

CREATE TABLE providers (

provider\_id SERIAL PRIMARY KEY,

provider\_name VARCHAR(25) NOT NULL,

address VARCHAR(40) NOT NULL,

phone VARCHAR(20) NULL,

reputation NUMERIC NULL

);

CREATE TABLE shops (

shop\_id INTEGER NOT NULL,

shop\_name VARCHAR(25) NOT NULL,

laptop\_id INTEGER NOT NULL REFERENCES laptops (laptop\_id),

provider\_id INTEGER NOT NULL REFERENCES providers (provider\_id),

last\_delivery TIMESTAMPTZ NULL,

shop\_address VARCHAR(40) NOT NULL,

quantity INTEGER NULL

);

CREATE UNIQUE INDEX laptop\_idx ON laptops(laptop\_id);

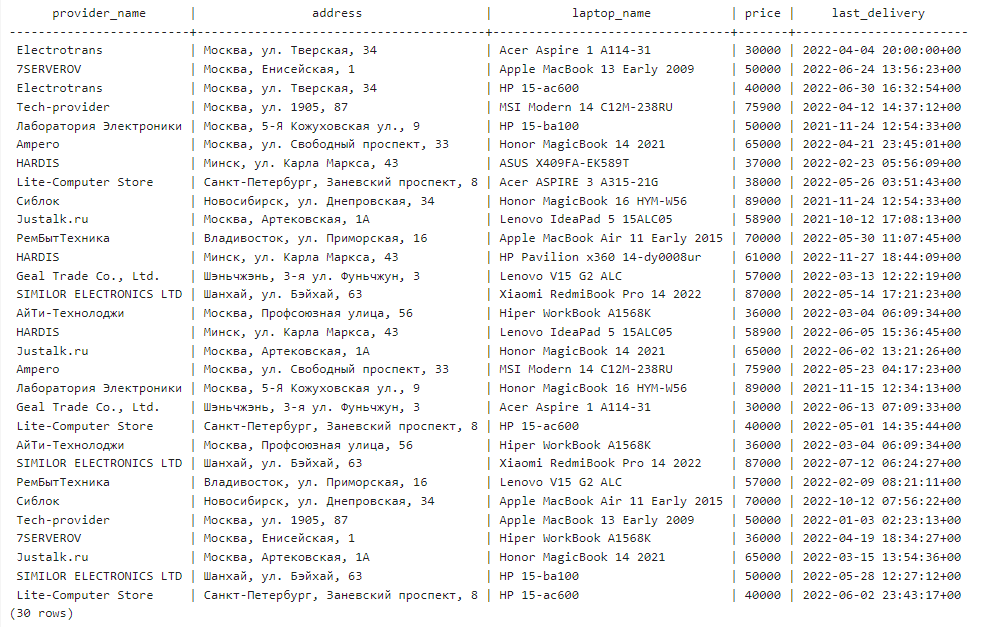
CREATE UNIQUE INDEX provider\_idx ON providers(provider\_id);

CREATE INDEX laptop\_provider\_idx ON shops(laptop\_id, provider\_id);

**Запросы к базе данных**

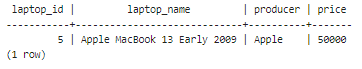
1. Получение списка список соответствия в каком магазине какой ноутбук находится.

SELECT providers.provider\_name, providers.address, tmp.laptop\_name, charasteric, tmp.price, last\_delivery from (SELECT \* FROM shops join laptops ON shops.laptop\_id = laptops.laptop\_id) tmp join providers on tmp.provider\_id = providers.provider\_id;



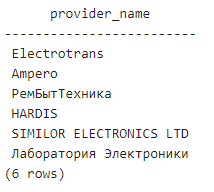
2. Выбрать все записи из таблицы ноутбуков, у которых цена меньше 60000 рублей и производитель Apple.

SELECT laptop\_id,laptop\_name,producer,price FROM laptops WHERE price < 60000 AND producer = 'Apple';



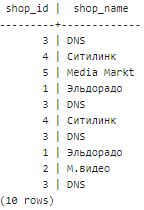
3. Выбрать имена всех поставщиков, у которых репутация выше средней.

SELECT provider\_name FROM providers WHERE providers.reputation > ANY(SELECT AVG(reputation) FROM providers);



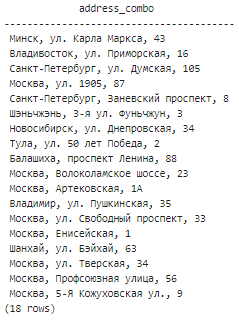
4. Получение магазинов, количество ноутбуков в которых больше половины от максимального по всем магазинам.

SELECT shop\_id, shop\_name FROM shops WHERE quantity >= (SELECT MAX(quantity)/2 FROM shops);



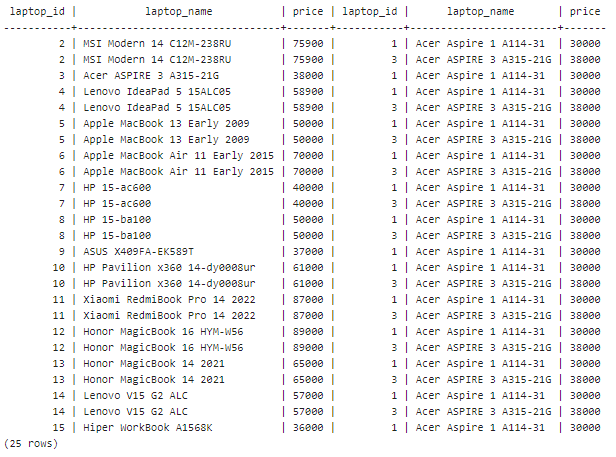
5. Выбрать объединение адресов поставщиков и магазинов.

SELECT address AS address\_combo FROM providers UNION SELECT shop\_address FROM shops;



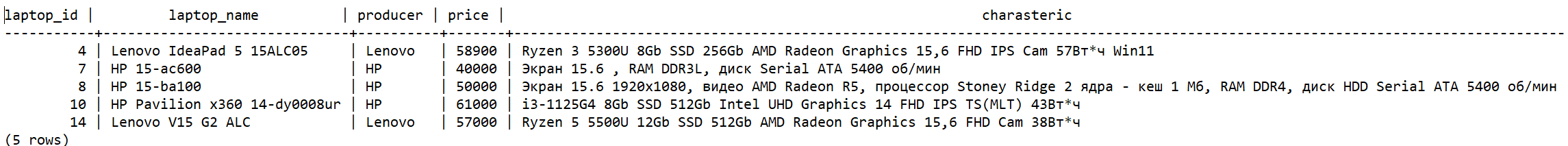
6. Показать для каждого ноутбука более дешевые аналоги, если они существуют, у которых диагональ 14 дюймов и производитель - Acer. Ноутбуки выводятся в порядке возрастания id, а их более дешевые аналоги – в порядке возрастания цены.

SELECT t1.laptop\_id, t1.laptop\_name, t1.price, t2.laptop\_id, t2.laptop\_name, t2.price FROM laptops t1 RIGHT JOIN (SELECT \* FROM laptops WHERE charasteric LIKE '%Экран 14%' AND producer NOT LIKE 'ACER') t2 ON t2.price < t1.price ORDER BY t1.laptop\_id, t2.price ASC;



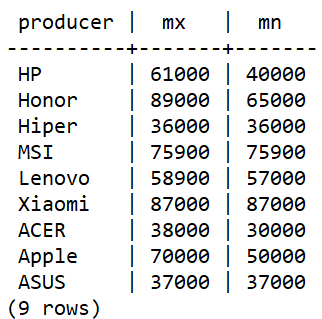
7. Выбрать из таблицы laptops только ноутбуки от производителей HP и Lenovo.

SELECT \* FROM laptops WHERE producer IN ('Lenovo', 'HP');



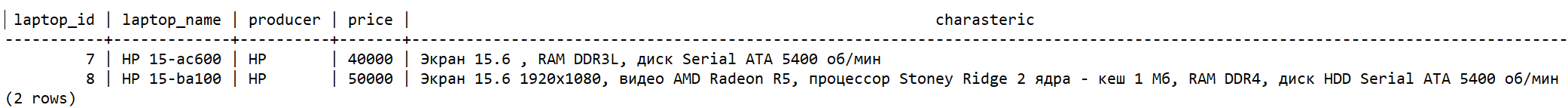
8. Показать наибольшую и наименьшую цены ноутбука от каждого производителя.

SELECT producer, MAX(price) AS MX, MIN(price) AS MN FROM laptops GROUP BY producer;



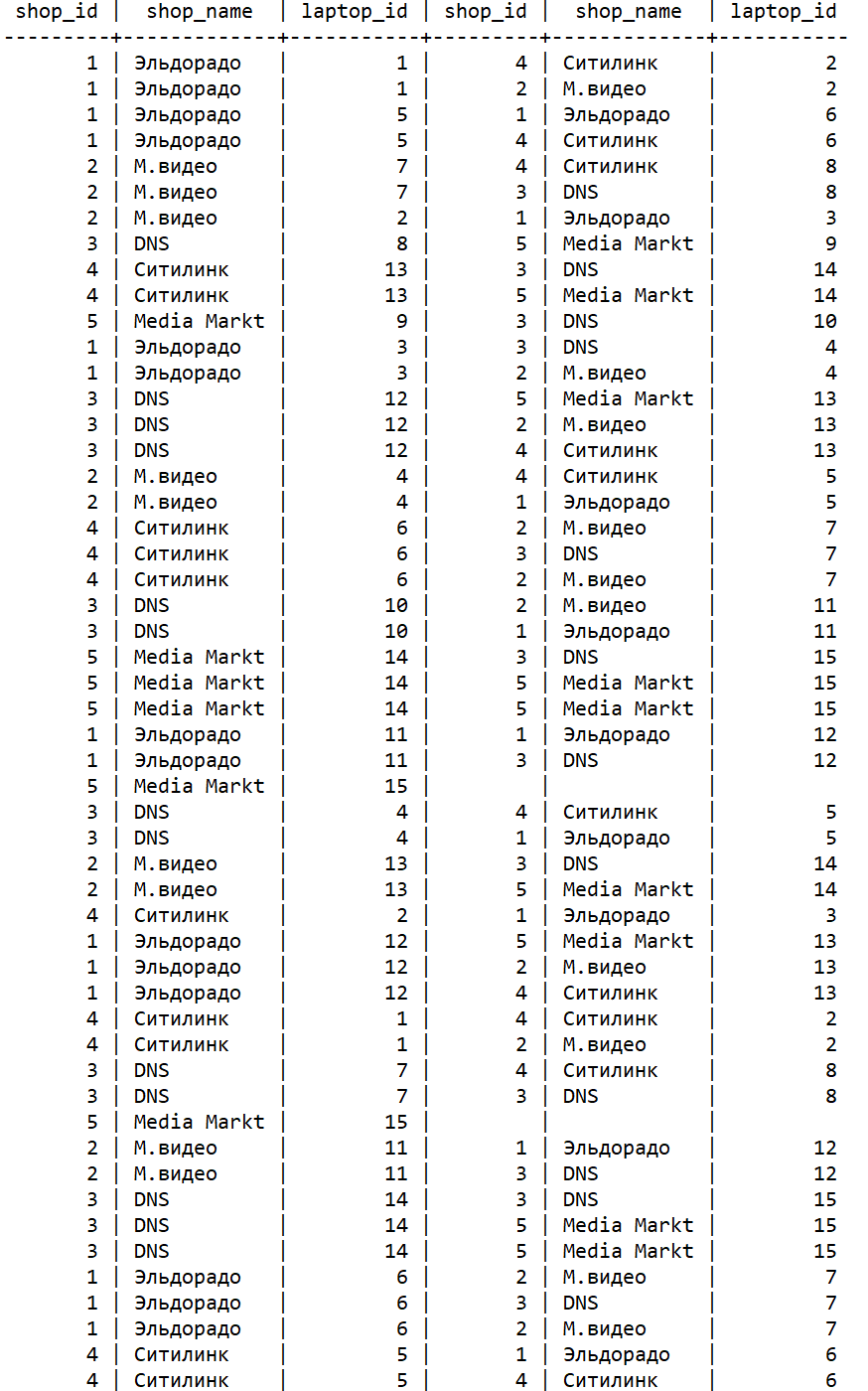
9. Выбрать ноутбуки HP в ценовом диапазоне от 30000 до 50000 рублей.

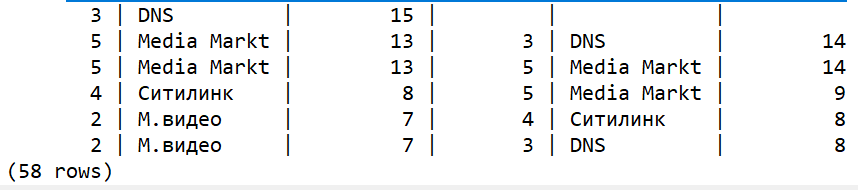
SELECT \* FROM laptops WHERE price BETWEEN 30000 AND 50000 AND producer LIKE 'HP';



10. Соединить таблицу shops саму с собой таким образом, чтобы каждому ноутбуку соответсвовал ноутбук с id, большим на единицу

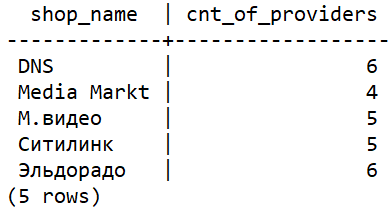
SELECT t1.shop\_id, t1.shop\_name, t1.laptop\_id, t2.shop\_id, t2.shop\_name, t2.laptop\_id FROM shops t1 LEFT JOIN shops t2 ON t2.laptop\_id = t1.laptop\_id + 1;





11 Показать количество различных поставщиков у каждого магазина

SELECT shop\_name, COUNT(DISTINCT provider\_id) AS cnt\_of\_providers FROM shops GROUP BY shop\_name;



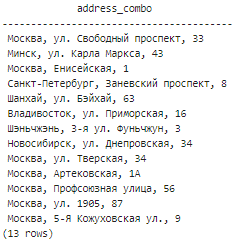
12. Выбрать пересечение адресов поставщиков и магазинов.

SELECT address AS address\_combo FROM providers INTERSECT SELECT shop\_address FROM shops;



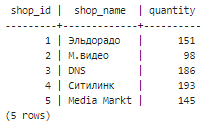
13. Выбрать только адреса поставщиков.

SELECT address AS address\_combo FROM providers EXCEPT SELECT shop\_address FROM shops;



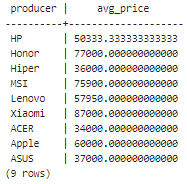
14. Выбрать количество товаров в каждом из магазинов

SELECT shop\_id, shop\_name, SUM(quantity) AS quantity FROM shops GROUP BY shop\_id, shop\_name ORDER BY shop\_id;



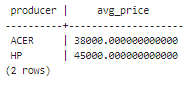
15. Получение средней цены ноутбука каждого из производителей.

SELECT producer, AVG(price) as avg\_price FROM laptops GROUP BY producer;



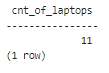
16. Выбор производителей, у которых есть ноутбуки с диагональю 15.6’ и вывод их средней цены.

SELECT producer, AVG(price) as avg\_price FROM laptops WHERE charasteric LIKE '%15.6%' GROUP BY producer;



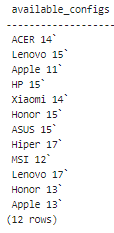
17. Получение количества ноутбуков, цена которых находится в диапазоне от 30000 до 65000.

SELECT COUNT(1) as cnt\_of\_laptops FROM laptops WHERE price BETWEEN 30000 AND 65000;



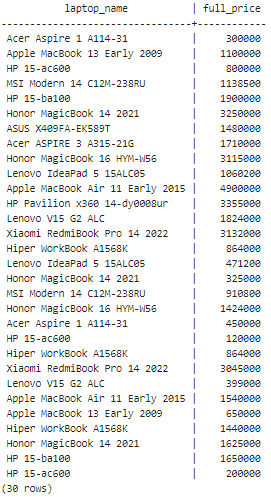
18. Выбор всевозможных конфигураций представленных в магазинах.

SELECT DISTINCT CONCAT(producer, ' ', SUBSTRING(charasteric FROM 7 FOR 2), '`') AS available\_configs FROM laptops;



19 Получение общей цена каждой из партий ноутбуков.

SELECT laptop\_name, tmp\_laptops.price \* quantity AS full\_price FROM (SELECT \* FROM laptops) as tmp\_laptops JOIN shops on shops.laptop\_id = tmp\_laptops.laptop\_id;



20. Удаление колонки адреса из таблицы providers.

ALTER TABLE providers DROP COLUMN IF EXISTS address;

SELECT \* FROM providers;

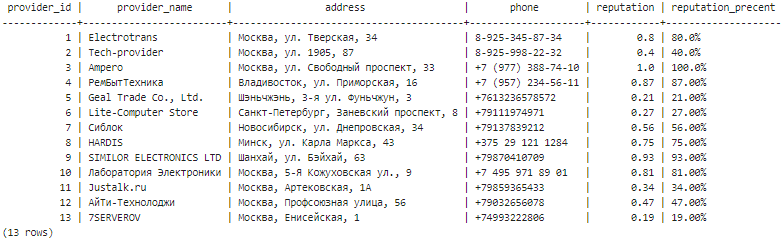


21. Добавление новой колонки в таблицу providers, в которой содержится репутация в процентах.

ALTER TABLE providers ADD COLUMN IF NOT EXISTS reputation\_precent VARCHAR(10);

UPDATE providers SET reputation\_precent = CONCAT(reputation \* 100, '%') WHERE reputation IS NOT NULL;

SELECT \* FROM providers;



22. Добавление ограничения на поле quantity в таблице shops.

ALTER TABLE shops ADD CONSTRAINT constraintname CHECK(quantity IS NOT NULL);

INSERT INTO shops VALUES (101, 'Эльдорадо', 1, 1, '2022-04-04 20:00:00', 'Балашиха, проспект Ленина, 88');



23. Выдача ноутбуков, цена которых больше цены всех ноутбуков марки HP.

SELECT laptop\_id,producer FROM laptops WHERE price > ALL(SELECT price FROM laptops WHERE producer='HP');

