Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Новосибирский государственный технический университет»

Кафедра вычислительной техники

**Отчет**

По лабораторной работе №3

По дисциплине: «Базы данных»

Использование агрегатных функций в среде PostgreSQL. Массивы.

Вариант 9

Выполнили: Павлюк А.С. Преподаватель: Харюткина С. А.

Группа: АВТ-008

Факультет: АВТ

Новосибирск, 2022

Оглавление

[Цель работы 3](file:///C:\Users\sasha\OneDrive\Рабочий%20стол\НГТУ\5\бд\лр%202\бд%20лр2.docx#_Toc112710446)

[Задание 3](file:///C:\Users\sasha\OneDrive\Рабочий%20стол\НГТУ\5\бд\лр%202\бд%20лр2.docx#_Toc112710447)

[Структура базы данных 4](file:///C:\Users\sasha\OneDrive\Рабочий%20стол\НГТУ\5\бд\лр%202\бд%20лр2.docx#_Toc112710448)

[Ход работы 4](file:///C:\Users\sasha\OneDrive\Рабочий%20стол\НГТУ\5\бд\лр%202\бд%20лр2.docx#_Toc112710449)

[Вывод 10](file:///C:\Users\sasha\OneDrive\Рабочий%20стол\НГТУ\5\бд\лр%202\бд%20лр2.docx#_Toc112710450)

[Приложение 11](file:///C:\Users\sasha\OneDrive\Рабочий%20стол\НГТУ\5\бд\лр%202\бд%20лр2.docx#_Toc112710451)

Цель работы

Изучить базовые операции по работе с массивами. Изучить синтаксис команд. Приобрести навыки работы с агрегатными функциями в PostgreSQL.

Задание

Ознакомиться с теоретическими сведениями о создании массивов. Создать таблицу с полем-массивом, таблицу с полем, содержащим многомерный массив. Выполнить вставку значений в созданные таблицы (минимум по 6 записей в каждой). Выполнить выборку из созданных таблиц (в том числе продемонстрировать предотвращение выборки NULL в массивах). Осуществить выборку с использованием среза. Продемонстрировать работу функции array\_dims(). Выполнить обновление данных в созданных таблицах. Осуществить модификацию среза массива, отдельного элемента массива. Проанализировать полученную в результате выполнения операций информацию.

*Вариант 9:* заполнить базу данных ремонтной мастерской (при необходимости). Найти предметы с минимальной стоимостью ремонта. Найти предметы с максимальной стоимостью ремонта, выполненные мастерами из Москвы. Найти количество отремонтированных предметов, выполненных за безналичный расчет. Найти среднюю стоимость отремонтированных предметов, выполненных за наличный расчет. Найти общую стоимость отремонтированных предметов для мастеров со стажем работы более 5 лет. Продемонстрировать результаты работы.

Структура базы данных

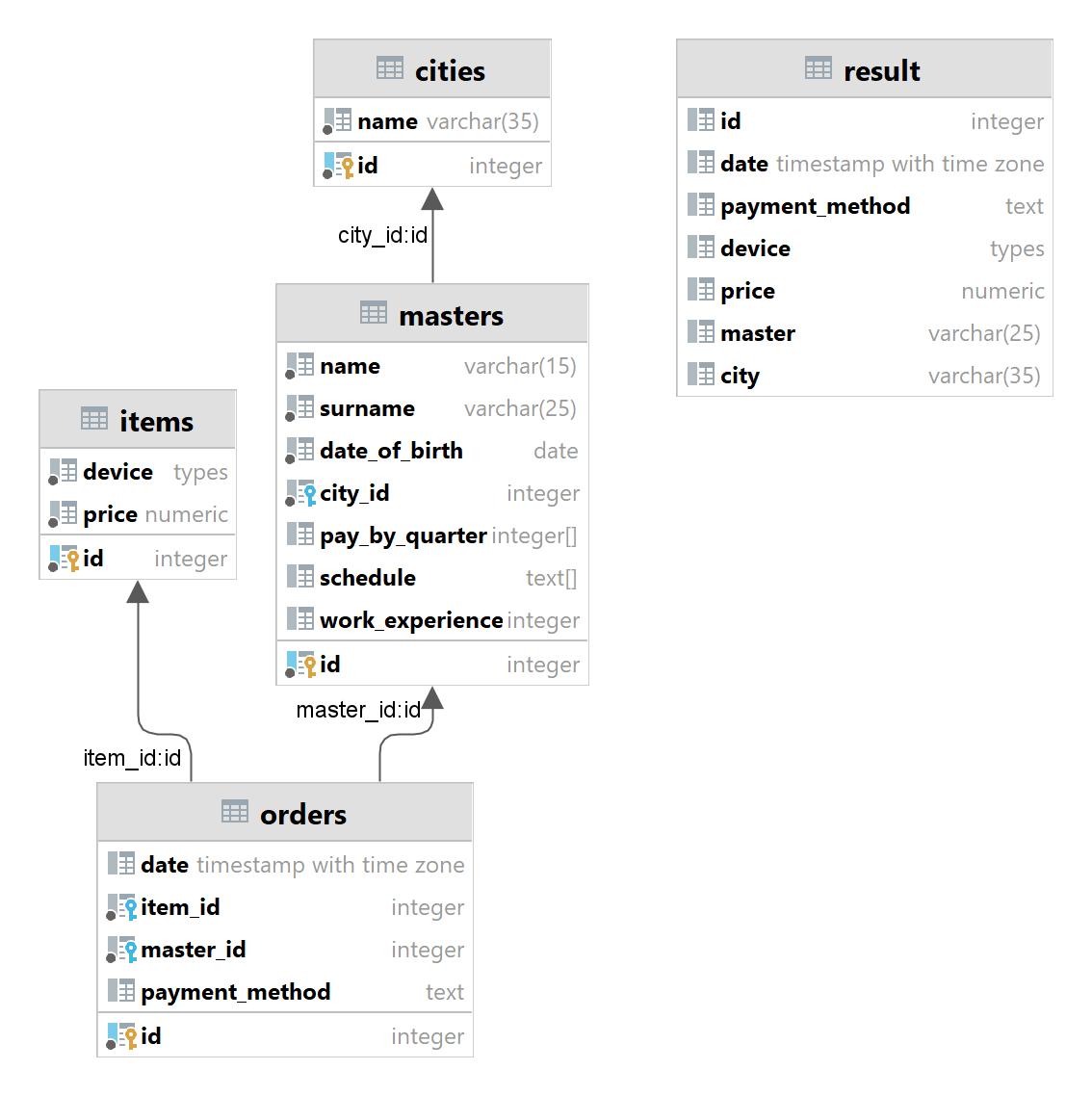


Рисунок 1. Структура базы данных

Ход работы

Поля данных PostgreSQL вместо отдельных величин могут содержать конструкции, называемые массивами.

Чтобы создать простейшее поле-массив, включите в команду CREATE TABLE или ALTER TABLE пару квадратных скобок после имени поля. Квадратные скобки показывают, что вместо одного значения в поле может храниться массив указанного типа.

Например, команда для создания поля single\_array типа type выглядит так:

single\_array type[] -- Одномерный массив

Дополнительные квадратные скобки определяют многомерные массивы, то есть «массивы массивов». Пример:

multi\_array type[][] -- Многомерный массив

Теоретически в квадратных скобках можно указать целое число, чтобы созданный массив имел фиксированный размер (то есть всегда состоял из п элементов по указанному измерению и не более). Но в текущей реализации это не учитывается.

Создадим таблицу контактов с полем массивом и двумерным массивом.

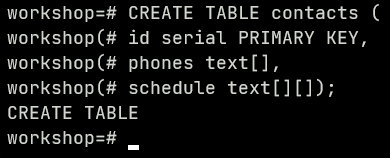


Рисунок 2. Создание таблицы с массивами

Приведем пример запроса для заполнения одномерного массива и двумерного:

INSERT INTO contacts VALUES (default, '{"(383)-123-45-67", "(383)-890-12-34", "(383)-567-89-01"}', '{{"09:00", "18:00"}, {"09:00", "18:00"}, {"09:00", "18:00"}, {"10:00", "18:00"}, {"10:00", "18:00"}}');

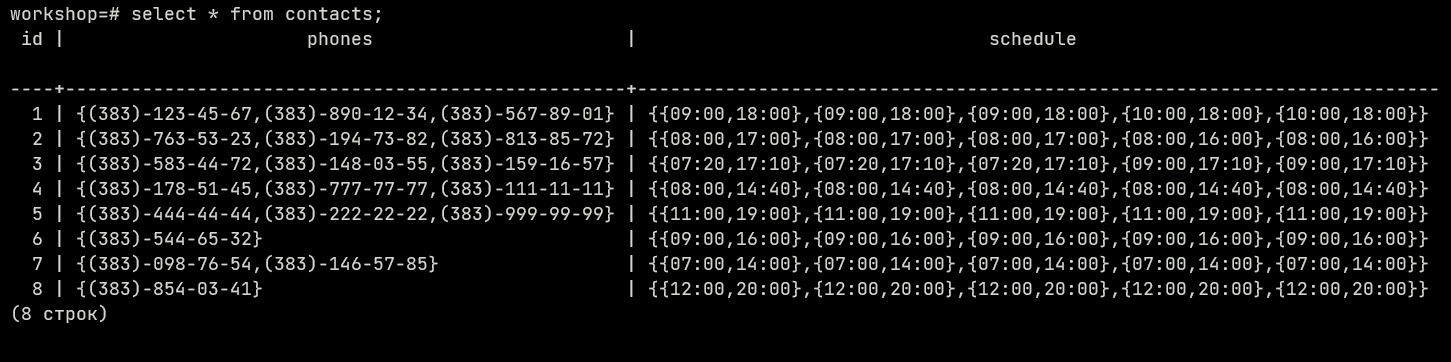


Рисунок 3. Заполнение массивов в таблице

Продемонстрируем результаты выборок полей массивов.

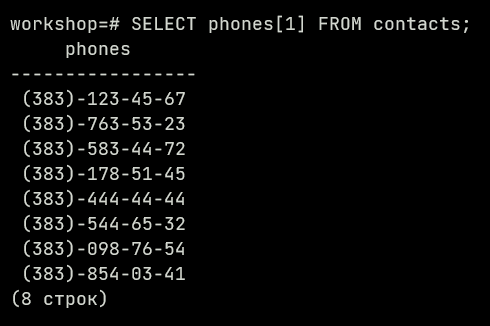


Рисунок 4. Выборка элементов одномерного массива

Если выйти за пределы массива, выборка будет состоять из NULL значений.

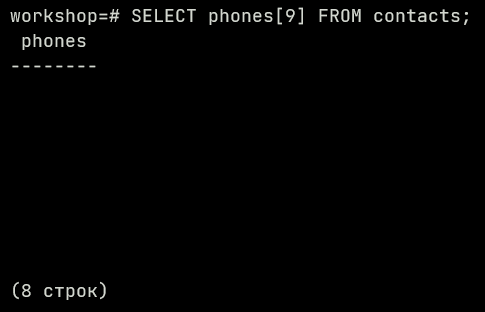


Рисунок 5. Выборка элементов за пределами массива

Это можно предотвратить, добавив условие IS NOT NULL.

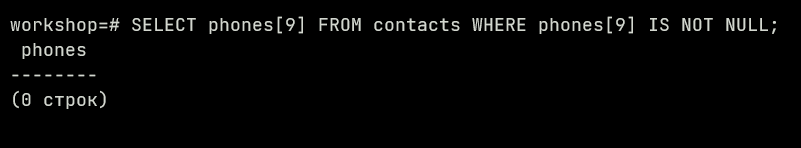


Рисунок 6. Предотвращение выборки NULL в массивах

Получим время начала работы первого контакта со вторника по четверг с помощью среза.

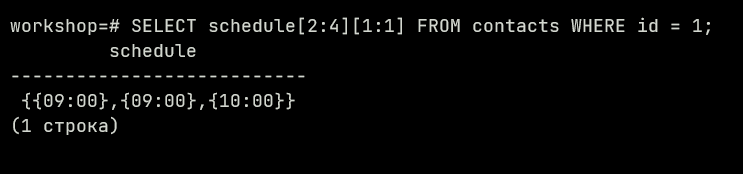


Рисунок 7. Срез двумерного массива

Теперь найдем третьи телефоны всех контактов.

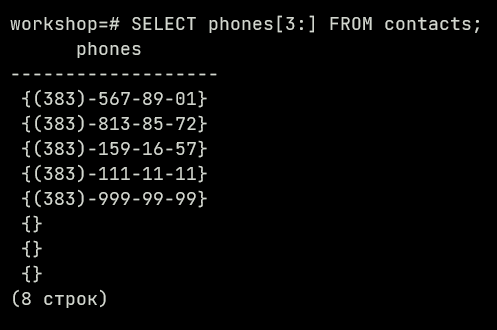


Рисунок 8. Срез одномерного массива

Функция array\_dims() возвращает текущие размеры значения массива.

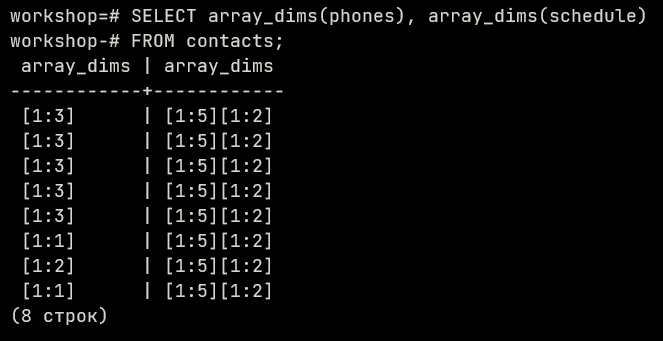


Рисунок 9. Пример работы функции array\_dims()

Обновим срез массива часов работы контактов. Установим начало рабочего дня в 12:00 в четверг и пятницу.

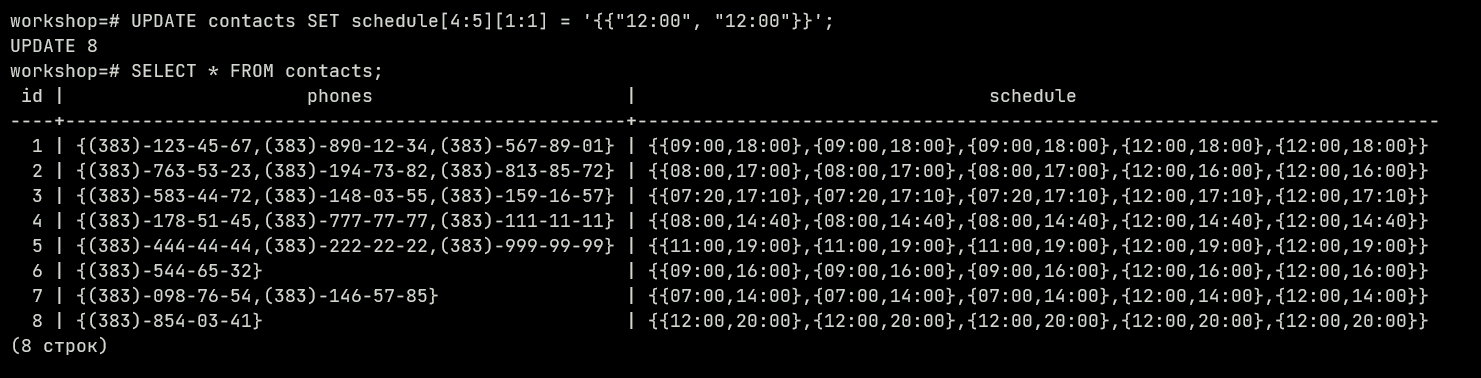


Рисунок 10. Обновление среза массива

Изменим второй телефон у пятого контакта.

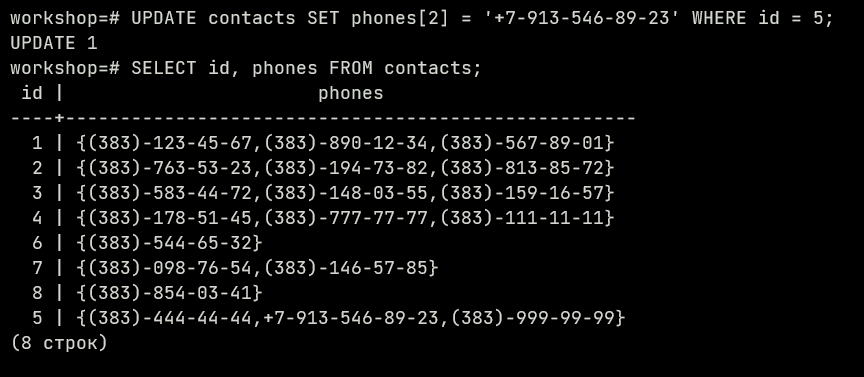


Рисунок 11. Обновление элемента массива

Поработав с массивами, можно сделать вывод, что одномерные массивы допустимы в схеме таблиц. Что касается многомерных массивов, как мне кажется, лучше выносить их в отдельные таблицы, чтобы пользователь мог легче обрабатывать информацию.

Перейдем к агрегатным функциям.

В SQL существует ряд специальных стандартных функций (SQL-функций). Кроме специального случая *COUNT(\*)* каждая из этих функций 10

оперирует совокупностью значений столбца некоторой таблицы и создает единственное значение.

Найдем предметы с минимальной стоимостью.

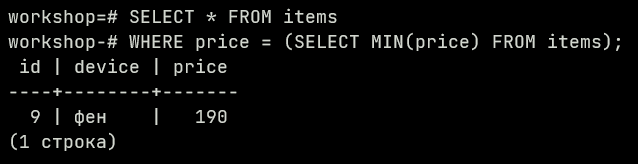


Рисунок 12. Предметы с минимальной стоимостью

Найдем предметы с максимальной стоимостью ремонта, выполненные мастерами из Москвы.

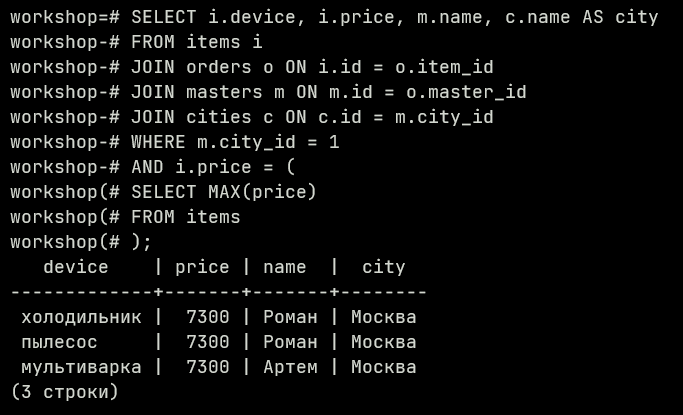


Рисунок 13. Предметы с максимальной стоимостью, выполненные мастерами из Москвы

Найдем количество предметов, выполненных за безналичный расчет.

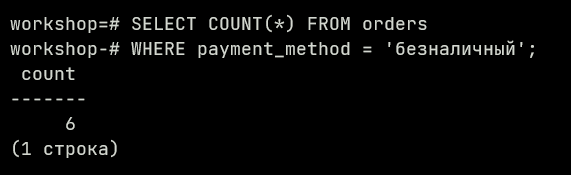


Рисунок 14. Количество предметов, выполненных за безналичный расчет

Найдем среднюю стоимость отремонтированных предметов, выполненных за наличный расчет.

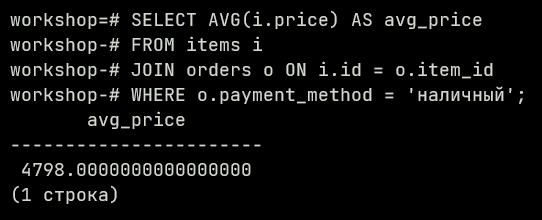


Рисунок 15. Средняя стоимость отремонтированных предметов, выполненных за наличный расчет

Добавим в таблицу мастеров поле «стаж работы».



Найдем общую стоимость отремонтированных предметов для мастеров со стажем работы более 5 лет.

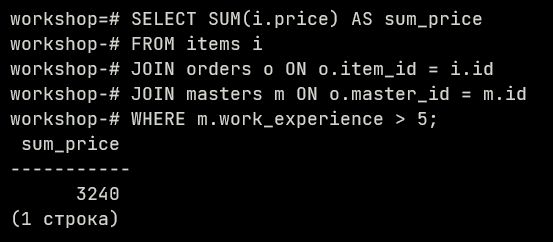


Рисунок 16. Общая стоимость отремонтированных предметов для мастеров со стажем работы более 5 лет

Вывод

В результате выполнения лабораторной работы были изучены базовые операции по работе с массивами. Также были приобретены навыки работы с агрегатными функциями в PostgreSQL.

Приложение

CREATE TABLE contacts (

id serial PRIMARY KEY,

phones text[],

schedule text[][]);

INSERT INTO contacts VALUES (default, '{"(383)-123-45-67", "(383)-890-12-34", "(383)-567-89-01"}', '{{"09:00", "18:00"}, {"09:00", "18:00"}, {"09:00", "18:00"}, {"10:00", "18:00"}, {"10:00", "18:00"}}');

SELECT phones[1] FROM contacts;

SELECT schedule[2:4][1:1] FROM contacts WHERE id = 1;

SELECT array\_dims(phones), array\_dims(schedule) FROM contacts;

UPDATE contacts SET schedule[4:5][1:1] = '{{"12:00", "12:00"}}';

UPDATE contacts SET phones[2] = '+7-913-546-89-23' WHERE id = 5;

SELECT \* FROM items

WHERE price = (SELECT MIN(price) FROM items);

SELECT i.device, i.price, m.name, c.name AS city

FROM items i

JOIN orders o ON i.id = o.item\_id

JOIN masters m ON m.id = o.master\_id

JOIN cities c ON c.id = m.city\_id

WHERE m.city\_id = 1

AND i.price = (

SELECT MAX(price)

FROM items

);

SELECT COUNT(\*) FROM orders

WHERE payment\_method = 'безналичный';

SELECT AVG(i.price) AS avg\_price

FROM items i

JOIN orders o ON i.id = o.item\_id

WHERE o.payment\_method = 'наличный';

SELECT SUM(i.price) AS sum\_price

FROM items i

JOIN orders o ON o.item\_id = i.id

JOIN masters m ON o.master\_id = m.id

WHERE m.work\_experience > 5;