Частное образовательное учреждение высшего образования

«Казанский инновационный университет имени В. Г. Тимирясова (ИЭУП)»

Факультет менеджмента и инженерного бизнеса

Кафедра информационных технологий и техносферной безопасности

**КурсовОЙ ПРОЕКТ**

**по дисциплине «Базы данных»**

**на тему: «База данных «Телефонные переговоры»**

Направление подготовки: 09.03.03 «Прикладная информатика»

**Выполнил:** \_\_\_\_\_\_\_\_\_\_\_\_\_\_

обучающийся гр. 1001у

заочной формы обучения

Поладов М. Ш. о.

**Руководитель**: \_\_\_\_\_\_\_\_\_\_\_

Кандидат экономических наук  
Григорян К. А.

Казань, 2021 г

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 2](#_Toc88577591)

[1. ИССЛЕДОВАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ 4](#_Toc88577592)

[2. КОНЦЕПТУАЛЬНОЕ ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ 5](#_Toc88577593)

[2.1. Определение сущностей 5](#_Toc88577594)

[2.2. Определение атрибутов 6](#_Toc88577595)

[3. ЛОГИЧЕСКОЕ ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ 7](#_Toc88577596)

[3.1. Нормализация отношений 7](#_Toc88577597)

[3.2. Классификация связей и определение ключей 9](#_Toc88577598)

[4. ФИЗИЧЕСКОЕ ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ 11](#_Toc88577599)

[5. РАЗРАБОТКА ЗАПРОСОВ, ПРЕДСТАВЛЕНИЙ, ФУНКЦИЙ И ТРИГГЕРОВ ДЛЯ БАЗЫ ДАННЫХ 17](#_Toc88577600)

[ЗАКЛЮЧЕНИЕ 31](#_Toc88577601)

[СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ 32](#_Toc88577602)

# ВВЕДЕНИЕ

На сегодняшнее время в компаниях, где общение с клиентом происходит непосредственно с помощью телефонного звонка, обычно задействована система записи телефонных звонков. Делается это с целью повышения качества обслуживания. Также запись телефонного разговора позволяет организациям решать проблемы других проблем в процессе общения с клиентом. Поэтому чаще всего средний и крупный бизнес прибегает к использованию такой системы [1]. Но перед подключением возникает вопрос о законности записи телефонных переговоров с клиентами [2].

В Конституции Российской Федерации нет ни одного закона прямой записи телефонного разговора клиентов и сотрудников организации, однако в законодательстве Российской Федерации имеется несколько правовых актов, которые регламентируют правомерность записи телефонных переговоров граждан. Конституцией Российской Федерации закреплено право каждого гражданина на тайну телефонных переговоров. В статье 24 говорится, что сбор, хранение, использование и распространение информации о частной жизни лица без его согласия не допускается.

В Федеральном законе «Об информации, информационных технологиях и о защите информации» говорится, что требование от граждан предоставления информации об их частной жизни и получение таких данных помимо воли граждан, помимо случаев, предусмотренных в других законах, находится под запретом.

Также в качестве регуляторов правомерной записи телефонных разговоров граждан выступают статьи 137 и 138 Уголовного кодекса Российской Федерации.

Поэтому перед тем, как проводить запись телефонных разговоров с клиентами, организации необходимо регламентировать способы записи и особенности записи.

Одним таким способ записи телефонных разговоров является некая система записи телефонного общения[3] с клиентами в одной организации, которая осуществляет звонки по межгороду в целях удовлетворения своих бизнес-интересов. Помимо записи телефонных разговоров система также вносит дополнительную информацию в базу данных [4], чтобы в дальнейшем специалисты компании могли провести анализ и оценить финансовые затраты на звонки.

Целью проводимого исследования является организация базы данных, которая будет хранить информацию проведенных телефонных переговорах.

Основные задачи проводимого исследования:

1. Исследование предметной области;
2. Концептуальное проектирование базы данных;
3. Логическое проектирование базы данных;
4. Физическое проектирование базы данных;
5. Разработка запросов, представлений, функций и триггеров для базы данных.
6. ИССЛЕДОВАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ

Юридическое лицо – это предприятие или организация, которая имеет имущественный комплекс для нормального производства и функционирования. Компания состоит из нескольких отделов, каждый из которых выполняет определенную роль.

Клиент – предприятие или организация, которое заинтересовано в партнерстве с нашей компанией. Основной способ коммуникации – телефонные звонки, так как клиенты-компании могут находиться в разных городах страны.

Провайдер телефонной связи – юридическое лицо или организация, деятельность которой заключается в предоставлении услуг связи для физических и юридических лиц.

Наша компания воспользовалась услугами провайдера и подключила себе телефонную связь на особых условиях для юридических лиц. В условиях подключенного тарифа указана поминутная ставка для каждого города в стране, а также предоставляемая для клиента скидка.

Дополнительно наша организация арендовала у провайдера оборудование для записи телефонных переговоров со своими партнерами. Данное оборудование проводит запись телефонных переговоров, а также позволяет записывать детальную информацию в базу данных, чтобы потом провести детализацию для дальнейшего анализа.

1. КОНЦЕПТУАЛЬНОЕ ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ

**Стейкхолдер –** это физическое либо юридическое лицо, которое прямо или косвенно воздействует на работу организации или располагает определёнными ожиданиями от результатов ее деятельности.

Концептуальное проектирование – это процесс создания модели используемой в организации информации, не зависящей от любых других физических аспектов ее представления [5]. Оно заключается в создании концептуальной модели данных для анализируемой части организации. Эта модель данных создается на основе информации, записанной в спецификациях требований пользователей. В отличие от физической реализации концептуальное проектирование не зависит от типа целевой СУБД, используемого языка программирования, типа вычислительной платформы и прочих особенностей. При разработке концептуальная модель данных постоянно подвергается тестированию и проверке на соответствие требованиям стейкхолдеров. Созданная концептуальная модель данных организации является источником информации для логического проектирования.

На текущем этапе формирование сущностей и атрибутов базы данных «Телефонные переговоры» будет осуществляться в два этапа:

* Определение сущностей;
* Определение атрибутов каждой сущности.
  1. Определение сущностей

Сущность представляет тип объектов, которые должны храниться в базе данных. Каждая таблица в базе данных представляет одну сущность. Как правило, сущности соответствуют объектам из исследуемой предметной области. Для правильного определения сущностей разработчику баз данных необходимо обратиться к требованиям или к бизнес-аналитику, который составлял эти требования в процессе переговоров с заказчиком.

Сущности, которые входят в базу данных «Телефонные переговоры»:

1. «Абонент» - юридические лица, с которыми сотрудничает наша компания;
2. «Номера Абонентов» - номер или номера партнеров нашей компании;
3. «Переговоры» - информация о каждом проведенном звонке;
4. «Тарифы» - информация по поминутной ставке для каждого города страны;
5. «Скидки» - информация о предоставляемых скидках для каждого города;
   1. Определение атрибутов

Атрибут представляет свойство, которое описывает некоторую характеристику сущности. Каждый столбец должен хранить один атрибут сущности.

Определим атрибуты для сущностей базы данных «Телефонные переговоры»:

1. Абоненты (идентификатор\_клиента, имя\_юр\_лица, ИНН, адрес)
2. Переговоры (идентификатор\_звонка, код\_города, идентификатор\_клиента, продолжительность\_звонка, стоимость\_разговора, стоимость\_разговора\_со\_скидкой, идентификатор\_номера\_телефона, номер\_телефона\_получателя, дата\_начала, дата\_конца)
3. Тарифы (код\_города, имя\_города, поминутная ставка)
4. Номера абонентов (идентификатор\_номера\_телефона\_клиента, идентификатор\_клиента, номер\_телефона)
5. Скидки (идентификатор\_скидки, код\_города, продолжительность\_скидки, процент\_скидки)
6. ЛОГИЧЕСКОЕ ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ

Логическое проектирование базы данных – это процесс создания модели используемой в организации информации на основе выбранной модели организации данных, но без учета типа СУБД и других физических аспектов реализации.

Данный раздел содержит два этапа:

* Нормализация отношений
* Классификация связей и определение ключей

# 3.1. Нормализация отношений

Нормальная форма — свойство отношения в реляционной модели данных, характеризующее его с точки зрения избыточности, потенциально приводящей к логически ошибочным результатам выборки или изменения данных. Нормальная форма определяется как совокупность требований, которым должно удовлетворять отношение.

Процесс нормализации базы данных заключается в декомпозиции исходных отношений на более простые отношения. Для построения правильной логической модели базы данных необходимо привести все отношения к третьей нормальной форме – 3НФ.

Отобразим разработанные ранее сущности и их атрибуты базы данных «Телефонные переговоры»:

1. Абоненты (идентификатор\_клиента, имя\_юр\_лица, ИНН, адрес)
2. Переговоры (идентификатор\_звонка, код\_города, идентификатор\_клиента, продолжительность\_звонка, стоимость\_разговора, стоимость\_разговора\_со\_скидкой, идентификатор\_номера\_телефона, номер\_телефона\_получателя, дата\_начала, дата\_конца)
3. Тарифы (код\_города, имя\_города, поминутная\_ставка)
4. Номера абонентов (идентификатор\_номера\_телефона\_клиента, идентификатор\_клиента, номер\_телефона)
5. Скидки (идентификатор\_скидки, код\_города, продолжительность\_скидки, процент\_скидки)

Текущая база данных находится в третьей нормальной форме (3НФ), так как отвечает требованиям первой, второй и третьей нормальных форм.

Первая нормальная форма. В таблице отсутствуют повторяющиеся строки; в каждой ячейке таблицы хранится только одно типовое не составное значение, а также отсутствуют списки и массивы в любой форме.

Вторая нормальная форма. База данных находится в первой нормальной форме; все таблицы имеют ключ, по которым гарантировано можно отличить строки друг от друга. Например:

1. В таблице «Абоненты» ключом является столбец «идентификатор\_клиента»;
2. В таблице «Переговоры» ключом является столбец «идентификатор\_звонка»;
3. В таблице «Тарифы» ключом является столбец «Код\_города»;
4. В таблице «Номера Абонентов» ключом является столбец «идентификатор\_номера\_телефона\_клиента»;
5. В таблице «Скидки» ключом является столбец «идентификатор\_скидки»;

Третья нормальная форма. База данных находится в первой и второй нормальных формах, в таблицах отсутствует транзитивная зависимость, то есть каждый неключевой столбец зависит только от первичного ключа.

# 3.2. Классификация связей и определение ключей

Первичный ключ – это поле в таблице, которое однозначно идентифицирует каждую строку в таблице базы данных. Первичные ключи могут содержать только уникальные значения и быть явными. Поскольку все сущности сформированы, опишем все связи, существующие между отношениями.

Связь 1:М означает, что несколько строк из дочерней таблицы зависят от одной строки в родительской таблице.

Связь 1:1 означает, что объекту одной сущности можно сопоставить только один объект другой сущности.

Связь М:М означает, что одна строка из таблицы А может быть связана с множеством строк из таблицы Б.

В базе данных «Телефонные переговоры» определены следующие отношения между таблицами:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Родительская таблица | Дочерняя таблица | Первичный ключ | Внешний ключ | Вид связи |
| 1 | Абоненты | Номера абонентов | ClientId | fk\_clientid | 1:M |
| 2 | Абоненты | Переговоры | ClientId | fk\_clientId | 1:M |
| 3 | НомераАбонентов | Переговоры | IDSerialKey | fk\_phoneIdFrom | 1:1 |
| 4 | Тарифы | Переговоры | CityId | fk\_cityId | 1:M |
| 5 | Тарифы | Скидки | CityId | fk\_cityID | 1:M |

Графическое представление модели базы данных «Телефонные переговоры» показано на рис. 3.1

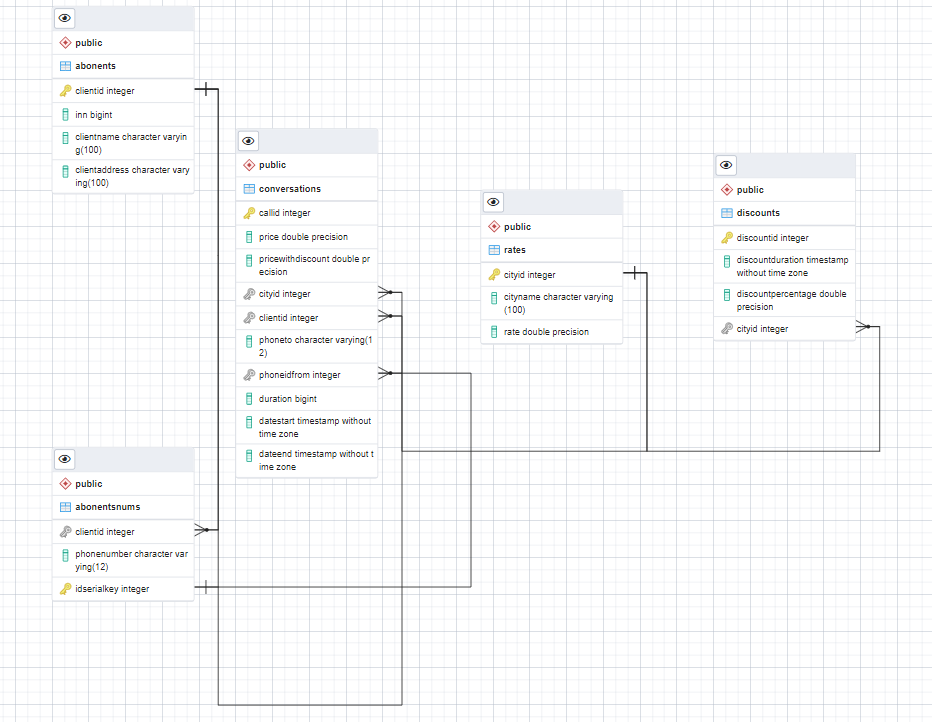


Рисунок 3.1 – Графическое представление базы данных «Телефонные переговоры»

1. ФИЗИЧЕСКОЕ ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ

Физическое проектирование баз данных – процесс подготовки описания реализации базы данных на вторичных запоминающих устройствах; на этом этапе рассматриваются основные отношения, организация файлов и индексов, предназначенных для обеспечения эффективного доступа к данным, а также все его связанные с этим ограничения целостности и средства защиты.

На текущем этапе производятся составление таблиц с подходящими свойствами атрибутов и их ограничений, создание первичных и внешних ключей и установка постоянных связей между сформированными таблицами.

Физическое проектирование производится на основе модели, полученной в результате логического проектирования.

В качестве реляционной системы управления базами данных будет применяться свободная объектно-реляционная система управления базами данных PostgreSQL [6].

В таблице 4.1 указаны атрибуты и их свойства для каждой сущности.

Таблица 4.1 – Сущности, атрибуты, выбранные типы данных и ограничения

|  |  |  |  |
| --- | --- | --- | --- |
| Сущность | Атрибут | Тип данных | Ограничения |
| Абоненты | ClientId | integer | NOT NULL, Primary Key, Identity Always |
| INN | integer | NOT NULL |
| ClientName | Character varying | NOT NULL, Length<=100 |
| ClientAddress | Character varying | NOT NULL, Length<=100 |
| Продолжение таблицы 4.1 | | | |
| НомераАбонентов | clientId | integer | NOT NULL |
| PhoneNumber | Character varying | NOT NULL |
| IDSerialKey | integer | NOT NULL, PrimaryKey |
| Переговоры | CallId | integer | NOT NULL, Primary key |
| Price | Double precision | As default 0 |
| PriceWithDiscount | Double precision | As default 0 |
| CityId | integer | NOT NULL |
| ClientId | integer | NOT NULL |
| PhoneTo | Character varying | NOT NULL, Length<=12 |
| PhoneIDFrom | integer | NOT NULL |
| Duration | bigint | As default 0 |
| DateStart | timestamp | NOT NULL |
| DateEnd | timestamp | >=DateStart |
| Скидки | DiscountID | integer | NOT NULL, Primary Key |
| DiscountDuration | timestamp | NOT NULL |
| DiscountPercentage | integer | NOT NULL, As default 0, Length >=0 OR Length <=100 |
| CityId | integer | NOT NULL |
| Тарифы | CityId | integer | NOT NULL, Primary Key |
| Rate | Double precision | NOT NULL |
|  | | | |
| Продолжение таблицы 4.1 | | | |
|  | CityName | Character varying | NOT NULL, Length <=100 |

Описание выбранных типов:

* Integer – целочисленный тип данных. Служит для представления целых чисел;
* Character varying – текст с ограничением по длине;
* Double precision – данный тип хранит значения с дробной частью после запятой;
* Bigint – целочисленный тип с увеличенным диапазоном;
* Timestamp – используется для величин с информацией только о дате в формате «YYYY-MM-DD HH:MM:SS».

Для многих столбцов выставлено ограничение NOT NULL, так как семантика предметной области требует явное значение в столбцах.

Для первичных ключей выставлено ограничение Primary Key, так как оно уникально идентифицирует строку в таблице, то есть каждая строка будет по своей сути уникальна.

Значения некоторых столбцов не должны превышать и/или быть меньше определенного значения. Поэтому для них будет выставлено ограничение CHECK.

Для некоторых столбцов выставлено ограничение DEFAULT, которое выставляет значение по умолчанию в случае, если при добавлении данных для столбца не будет предусмотрено значение.

Создание базы данных «Телефонные переговоры».

Создание новой базы производится с помощью команды CREATE DATABASE <Имя\_Базы\_Данных>. В качестве суперпользователя будет postgres, кодировка UTF-8, устанавливаем неограниченное количество одновременно подключенных к базе данных пользователей с помощью команды CONNECTION LIMIT = -1, устанавливаем табличные пространства с помощью команды TABLESPACE. Полный текст выглядит так:

CREATE DATABASE "Phone conversations"

WITH

OWNER = postgres

ENCODING = 'UTF8'

LC\_COLLATE = 'Russian\_Russia.1251'

LC\_CTYPE = 'Russian\_Russia.1251'

TABLESPACE = pg\_default

CONNECTION LIMIT = -1;

COMMENT ON DATABASE "Phone conversations"

IS 'Курсовая работа';

Теперь необходимо заполнить базу данных таблицами, чтобы полноценно быть базой данных.

Создание таблицы производится с помощью команды CREATE TABLE <имя\_таблицы>. Далее по очереди вводятся имена столбцов и их тип, потом с помощью команды CONSTRAINT устанавливаются ограничения, которые раннее были отображены в таблице 4.1.

Код таблицы «Абоненты» отображен на рисунке 4.1.

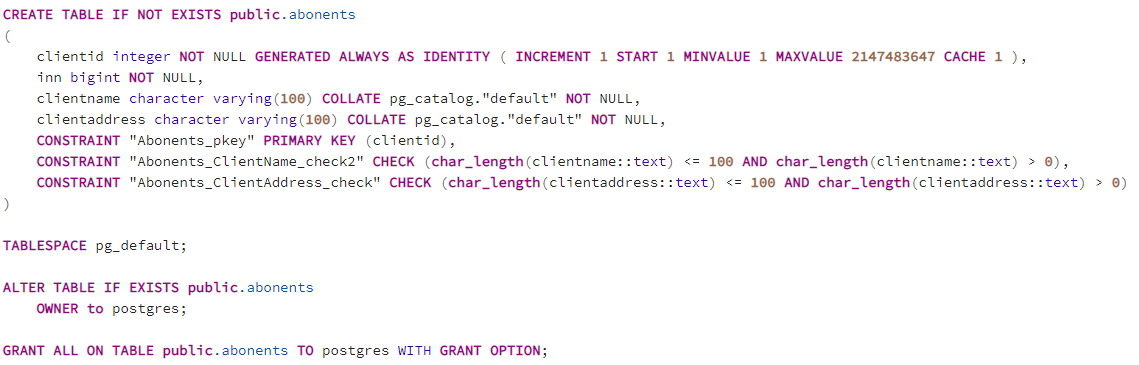


Рисунок 4.1 – Код таблицы «Абоненты»

Код таблицы «Номера абонентов» отображен на рисунке 4.2.

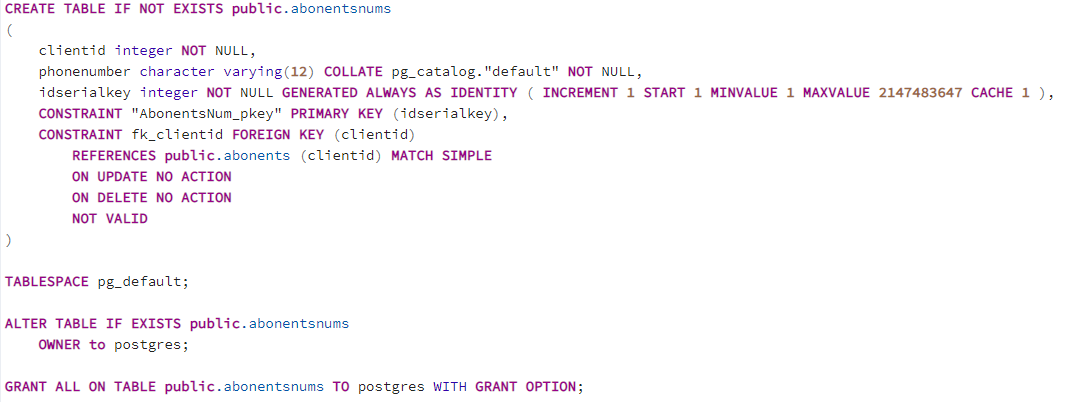


Рисунок 4.2 – Код таблицы «Номера абонентов»

Код таблицы «Тарифы» отображен на рисунке 4.3.

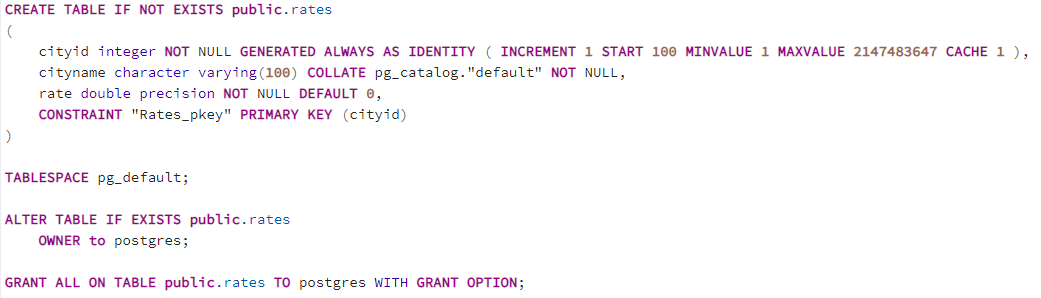


Рисунок 4.3 – Код таблицы «Тарифы»

Код таблицы «Переговоры» отображен на рисунке 4.4.

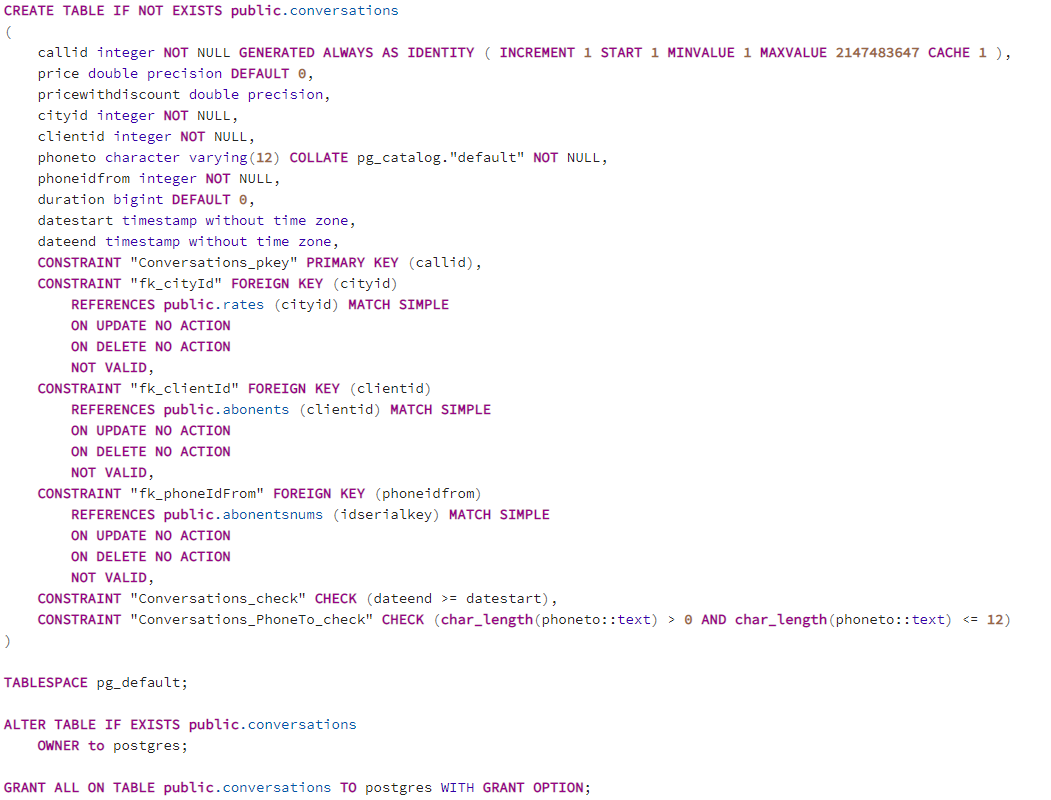


Рисунок 4.4 – Код таблицы «Переговоры»

Код таблицы «Скидки» отображен на рисунке 4.5.

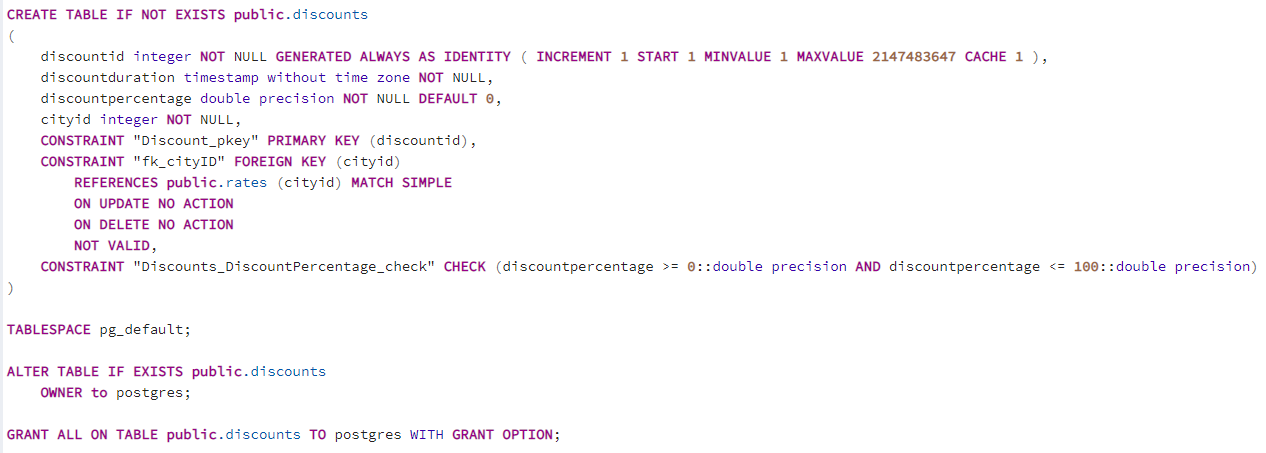


Рисунок 4.5 – Код таблицы «Скидки»

1. РАЗРАБОТКА ЗАПРОСОВ, ПРЕДСТАВЛЕНИЙ, ФУНКЦИЙ И ТРИГГЕРОВ ДЛЯ БАЗЫ ДАННЫХ

Запрос – это инструмент для извлечения и изменения информации из одной или нескольких таблиц базы данных. Запрос может использоваться для формирования отчетов и страниц доступа к данным.

После того как база данных была сформирована, необходимо заполнить созданные таблицы данными для их дальнейшего использования.

Но прежде чем заполнить таблицы данными, необходимо установить триггеры, которые будут вызывать процедуры для расчета некоторых столбцов на основе значений других столбцов [7].

Например, для расчета столбца «Длительность» из таблицы «Переговоры» необходимо найти разницу в секундах между столбцом «Дата Конца» и «Дата Начала», а для столбца «Стоимость» умножить «Длительность» на «Тариф» из таблицы «Тарифы».

Напишем триггер, который будет срабатывать каждый раз после внесения записи в таблицу «Переговоры» (Рис. 4.6):

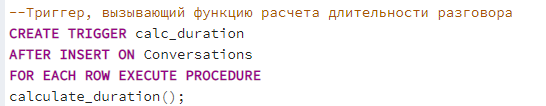


Рисунок 4.6 – Триггер для расчета длительности

Теперь напишем процедуру, которая будет вызываться этим триггером каждый раз, когда в таблицу «Переговоры» будет вноситься запись (Рис. 4.7):

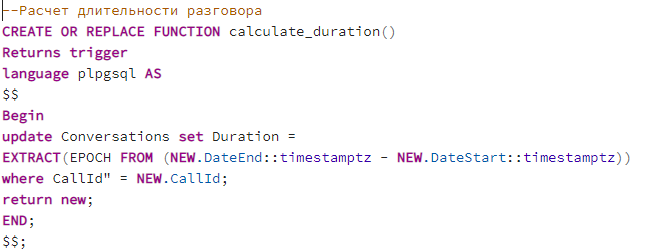


Рисунок 4.7 – Процедура расчета длительности разговора

Теперь напишем аналогичный триггер, который будет вызывать процедуру расчета стоимости телефонного разговора без скидки и со скидкой (Рис. 4.8):

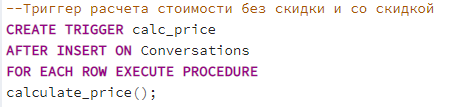


Рисунок 4.8 – Триггер для расчета стоимости разговоров

Процедура расчета стоимости телефонных разговоров будет выглядеть следующим способом (Рис. 4.9):

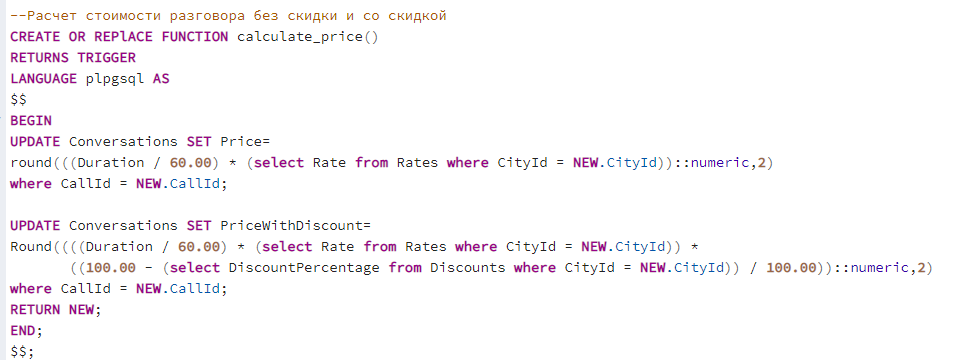


Рисунок 4.9 – Процедура для расчета стоимости телефонных разговоров

Теперь, когда установлены триггеры и процедуры для расчета значений некоторых столбцов, можно заполнить таблицы данными.

Следующий запрос заполнит все таблицы в базе данных «Телефонные переговоры» (Рис. 4.6):

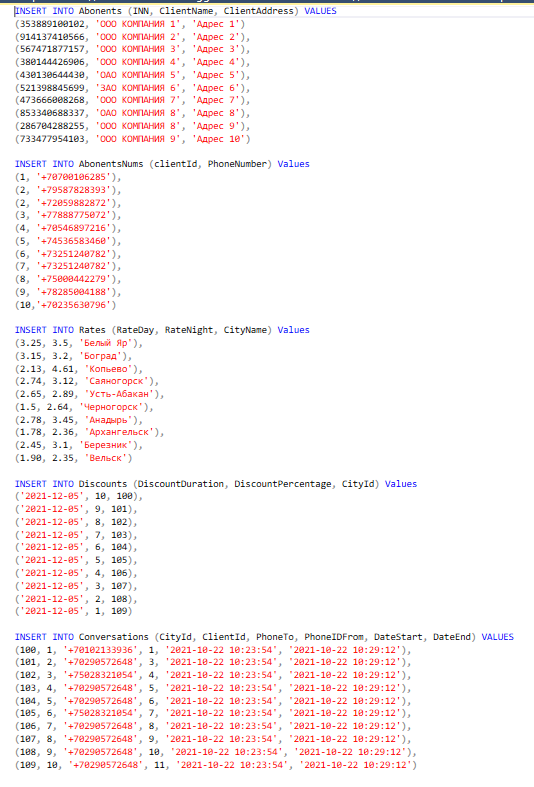


Рисунок 4.10 – Запрос на заполнение данными всех таблиц

Просмотреть заполненные данные в таблицах можно при помощи команды SELECT \* FROM <Имя\_Таблицы>.

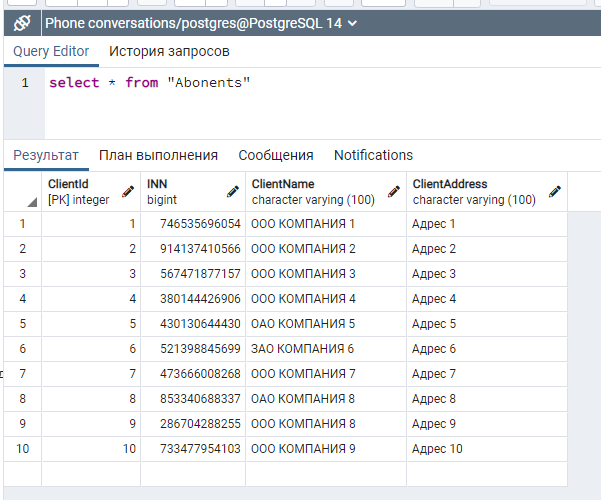


Рисунок 4.11 – Таблица «Абоненты»

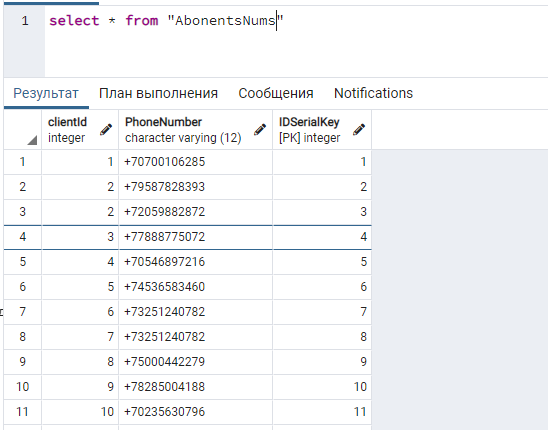


Рисунок 4.12 – Таблица «Номера абонентов»

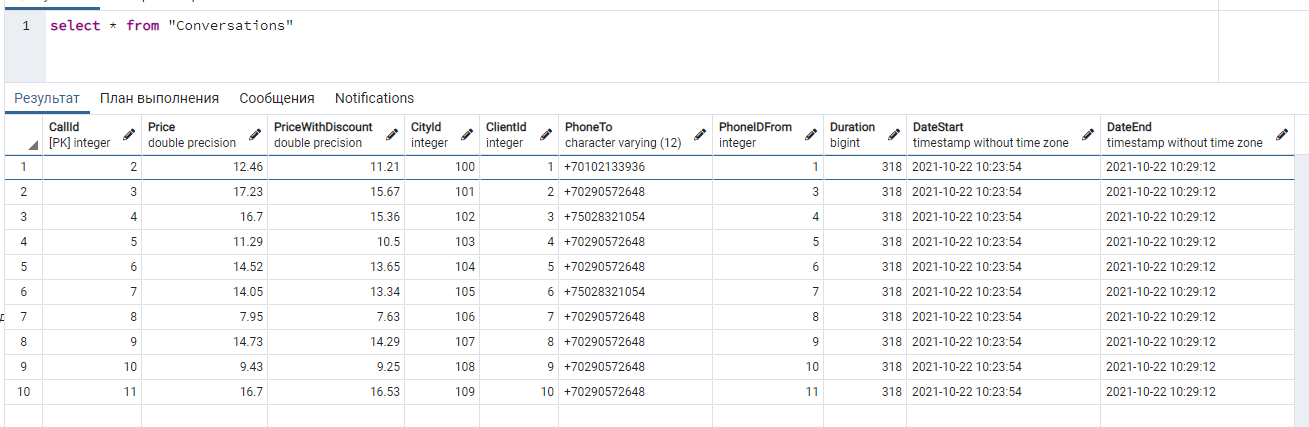


Рисунок 4.13 – Таблица «Переговоры»

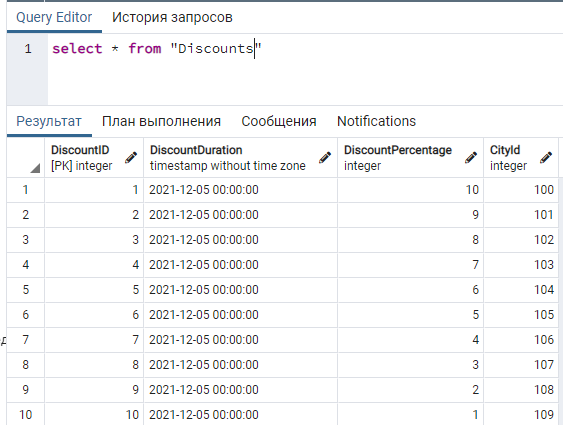


Рисунок 4.14 – Таблица «Скидки»

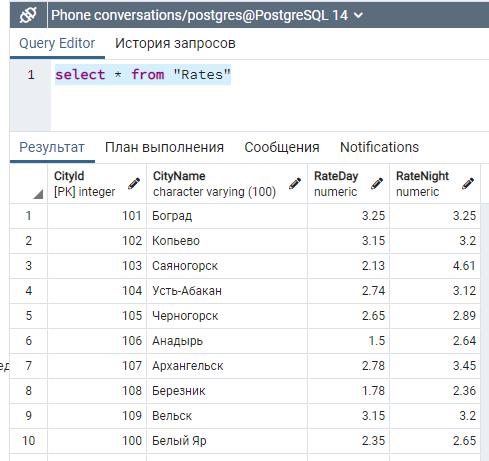


Рисунок 4.15 – Таблица «Тарифы»

Как видим, в таблице «Переговоры» столбцы «Длительность», «Стоимость» и «Стоимость со скидкой» были заполнены автоматически благодаря описанным раннее процедурам, которые были вызваны триггерами.

Теперь отобразим запросы, которые позволяют получить некоторую информацию как из одной, так и из нескольких таблиц.

Необходимо получить из базы данных «Телефонные переговоры» информацию по скидке для конкретного кода города. Для этого выполним следующий запрос (Рис. 4.16):

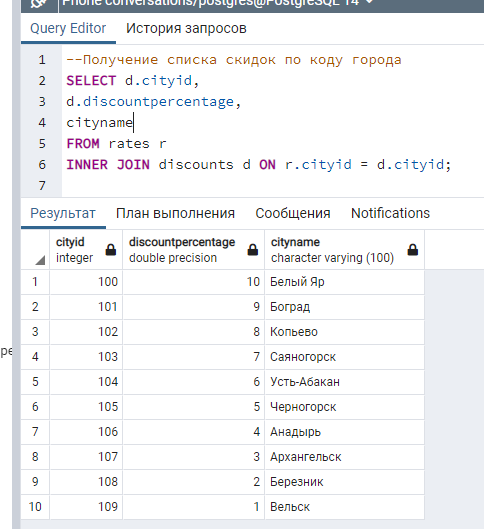


Рисунок 4.16 – Получение списка скидок по коду города

Необходимо получить информацию по клиентам и их затратам на телефонные разговоры и отсортировать по убыванию затрат. Для этого напишем следующий запрос (Рис. 4.17):

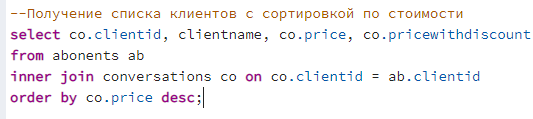


Рисунок 4.17 – Получение отсортированного списка клиентов с их затратами

Результатом данного запроса будет отсортированный список с клиентами и их затратами на телефонные переговоры (Рис. 4.18):

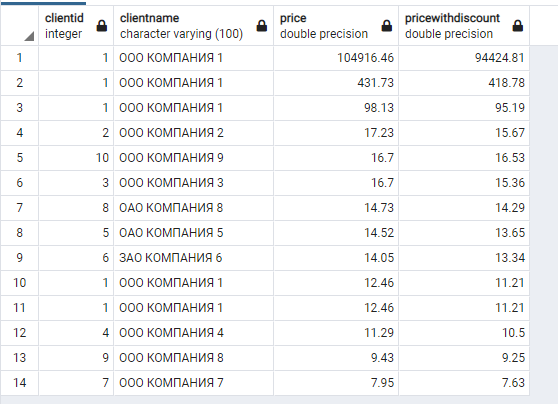


Рисунок 4.18 – Результат запроса получения списка клиентов, отсортированного по их затратам

Необходимо написать запрос на получение номеров клиентов. Запрос будет выглядеть следующим образом (Рис. 4.19):

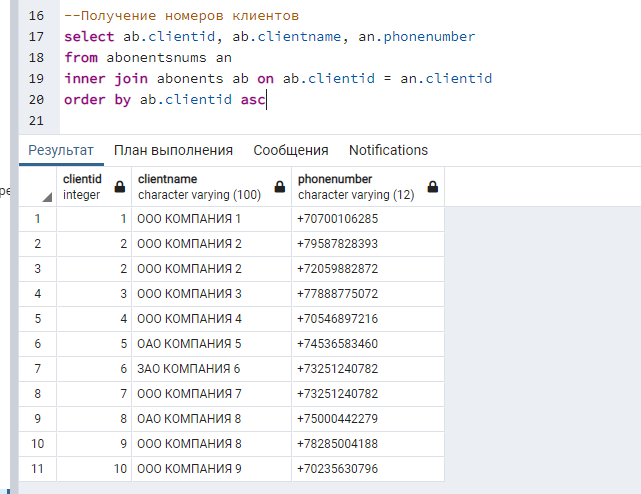


Рисунок 4.19 – Запрос на получение номеров клиентов

Необходимо выполнить запрос на получение тарифа и скидки для каждого города, отсортированный по коду города. Запрос будет выглядеть следующим образом (Рис. 4.20):

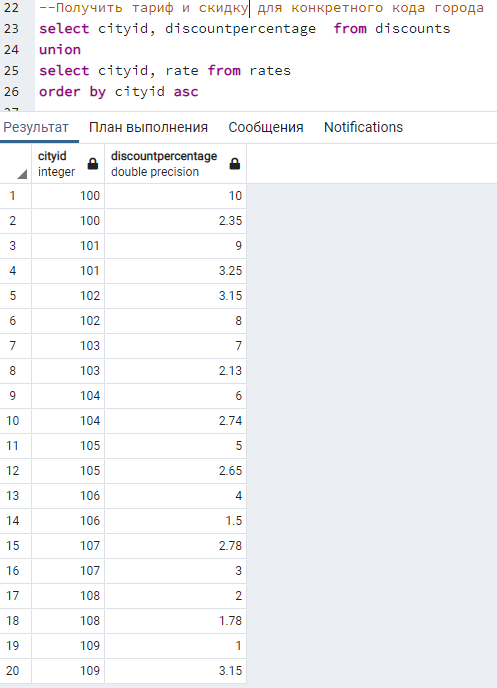


Рисунок 4.20 – Получение тарифа и скидки для каждого города

Необходимо написать запрос, при выполнении которого будет выводиться список затрат на каждый город, сгруппированный по коду города. Запрос будет выглядеть следующим образом (Рис. 4.21):

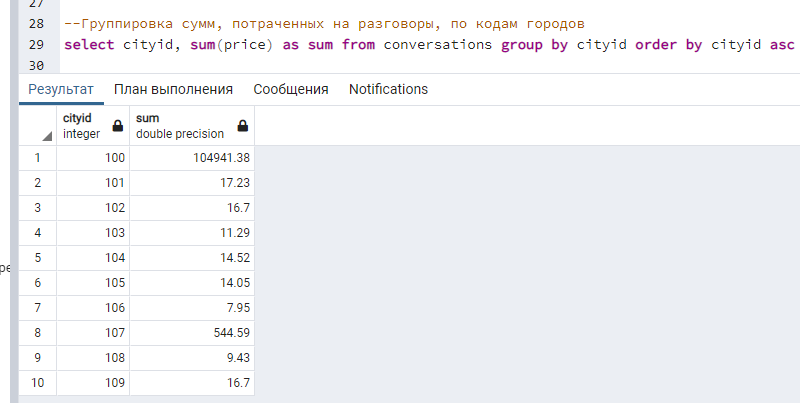


Рисунок 4.21 – Группировка затрат по коду города

Необходимо написать запрос на получение списка затрат по каждому городу, превышающих определенную сумму. Запрос будет выглядеть следующим образом (Рис. 4.22):

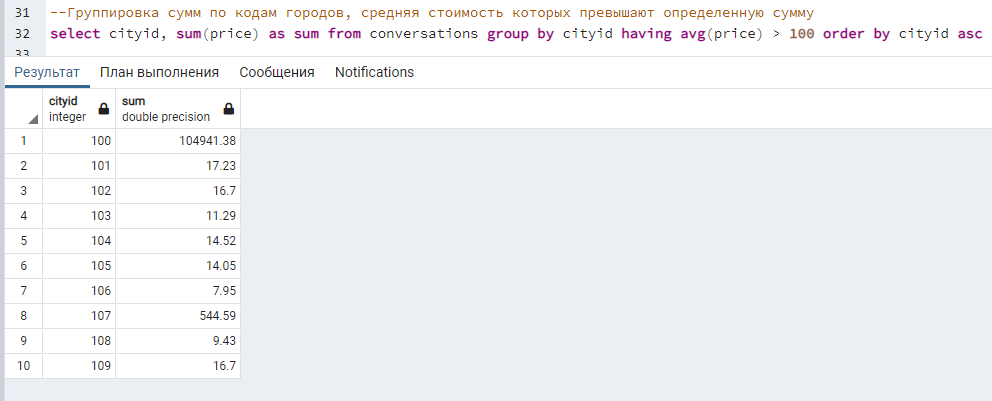


Рисунок 4.22 – Группировка затрат, превышающих определенную сумму, по коду города

Необходимо написать запрос на получение информации по звонкам, в которой должна быть информация по клиенту, его номера, номера, которому звонили и продолжительности разговора. Запрос будет выглядеть следующим образом (Рис. 4.23):

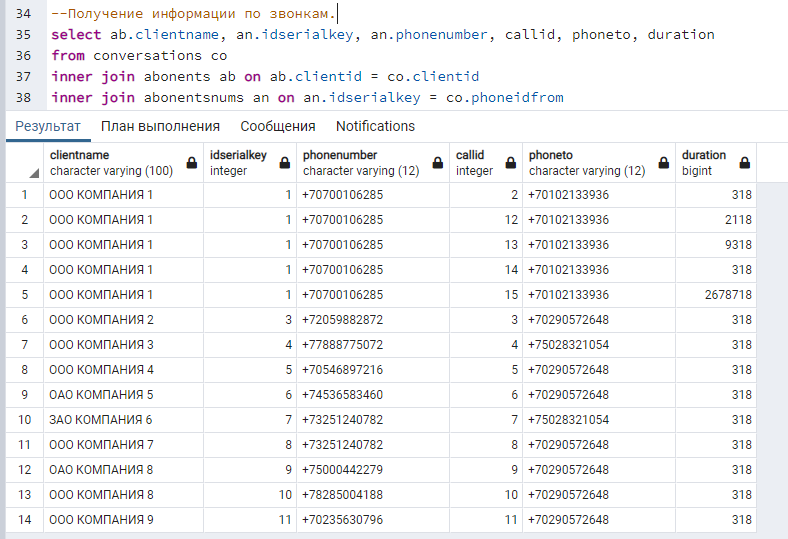


Рисунок 4.23 – Запрос на получение информации по звонкам

Необходимо написать запрос на получение среднего времени звонков в минутах. Для этого выполним следующий запрос (Рис. 4.24):

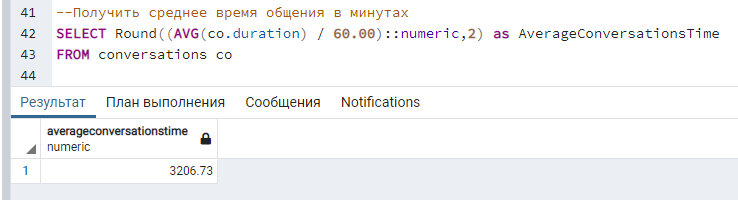


Рисунок 4.24 – Запрос на получение среднего времени звонков в минутах

Необходимо написать запрос на получение списка клиентов и городов, в которые они звонят. Выполним запрос следующего вида (Рис. 4.25):

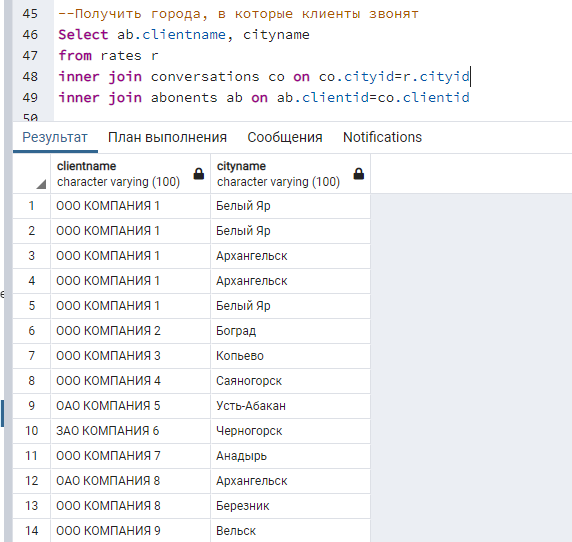


Рисунок 4.25 – Запрос на получение списка клиентов и городов, в которые они звонят

Теперь напишем виртуальные таблицы, содержимое которых определяется запросом.

Следующее представление будет отображать список клиентов и их номеров (Рис. 4.26):

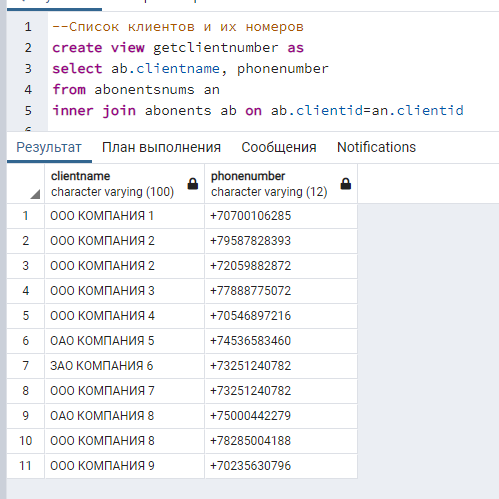


Рисунок 4.26 – Получение списка клиентов и их номеров

Следующий запрос создаст представление, которое будет выводить список городов и соответствующую каждому городу скидку (Рис. 4.27):

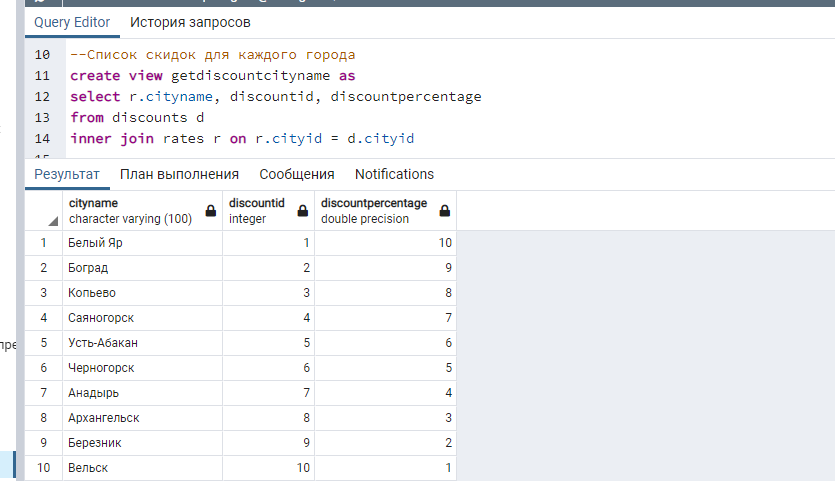


Рисунок 4.27 – Получение списка скидок для каждого города

Следующее представление будет отображать таблицу со списком телефонных переговоров и номеров, с которых звонили клиенты, а также номеров, куда звонили клиентов (Рис. 4.28):



Рисунок 4.28 – Получение списка клиентов с их номерами и номерами, куда звонили клиенты

ЗАКЛЮЧЕНИЕ

В ходе курсового исследования были достигнуты поставленные цели закрепления полученных практических навыков и знаний в разработке архитектуры реляционных баз данных, использования концептуального, логического и физического подходов к проектированию. Также в процессе проектирования базы данных «Телефонные переговоры» были использованы различные инструменты, предоставляемых реляционной системой управления базами данных PostgreSQL, таких, как триггеры, хранимые процедуры и представления.

По итогу проведенного исследования была сформирована база данных «Телефонные переговоры» с таблицами, связями между ними и вычисляемыми столбцами.

# СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Кэмпбелл Лейн, Мейджорс Черити. Базы данных. Инжиниринг надежности. — СПб.: Питер, 2020. — 304 с.: ил. — (Серия «Бестселлеры O’Reilly»);
2. Обеспечение законности записи телефонных разговоров с клиентами [Электронный ресурс]. – URL: <https://www.radium-it.ru/brief/pd/phone-record/> ;
3. Системы записи разговоров [Электронный ресурс]. – URL: <https://astrom-nw.ru/article/sistemy-zapisi-razgovorov> ;
4. Ёcy М. Т., Вальдуриес П. Принципы организации распределенных баз данных / пер. с англ. А. А. Слинкина. – М.: ДМК Пресс, 2021. – 672 с.: ил.;
5. Проектирование баз данных [Электронный ресурс]. – URL: <http://www.mstu.edu.ru/study/materials/zelenkov/ch_5_1.html> ;
6. Моргунов, Е. П. PostgreSQL. Основы языка SQL: учеб. пособие / Е. П. Моргунов; под ред. Е. В. Рогова, П. В. Лузанова. — СПб.: БХВ-Петербург, 2018. — 336 с.: ил.;
7. Домбровская Г., Новиков Б., Бейликова А. Оптимизация запросов в PostgreSQL / пер. с англ. Д. А. Беликова. – М.: ДМК Пресс, 2022. – 278 с.: ил.