**курсОВАЯ РАБОТА**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| по МДК | 04.01 | Технология разработки и защиты баз данных | | | | | | |
|  | | | | | | | | |
|  | | | 6 семестр | | | | | |
| Тема курсовой работы | | | Разработка БД | | | | | |
|  | | | «Продажа билетов автовокзала» | | | | | |
|  | | |  | | | | | |
| Аквт.09.02.07. КР45.22ПЗ | | | | | | | | |
| Листов: | | | | 18 |  | | | |
|  | |  | | | | | | |
|  | |  | | | | | | |
| Разработчик | |  | | | | Студент гр. | | ПБ – 31 |
|  | |  | | | | Баринов Н. В. | | |
|  | |  | | | |  |  | |
| Руководитель | |  | | | | Преподаватель | | |
|  | |  | | | | Храмцовский И.А | | |
|  | |  | | | |  | | |
|  | |  | | | |  | | |

|  |
| --- |
| УТВЕРЖДЕНО |
| Цикловой комиссией |
| специальности 09.02.07 |
| ПЦК \_\_\_\_\_\_\_\_\_ Ю.С. Андрианова |
| «\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2023 г. |
| **Специальность** | 09.02.07 Информационные системы и программирование | |
| **МДК 02.01** | Технология разработки и защита баз данных | |
| **Группа** | ПБ – 31 | |
| **ЗАДАНИЕ** | | | |
| **на курсовое проектирование** | | | |
| **Баринова Никиты Васильевича** | | | |
| (фамилия, имя, отчество студента) | | | |
| Тема курсового проекта | | **Разработка БД** | |
| **«Продажа билетов автовокзала»** | | | |

Содержание пояснительной записки курсового проекта

Аннотация

Содержание

Введение

1. Технический проект

1.1 Описание предметной области

1.2 Проектирование концептуальной модели

1.2.1 Разработка логической модели данных

1.2.2 Разработка диаграммы вариантов использования

1.3 Выбор СУБД

2 Реализация базы данных

2.1 Написание исходного кода БД

2.2 Написание запросов

Заключение

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Дата выдачи задания | « | 20 | » | 02. | |  | 20 | 23 | г. |
| Руководитель |  | | | | Храмцовский И.А | | | | |
| Студент |  | | | | Баринов Н.В. | | | | |

**Аннотация**

Представленная курсовая работа состоит из 18 страниц.

Объект исследования – общие сведения о деятельности автовокзалов.

Цель работы: создание базы данных «Продажа билетов автовокзалов».

Предметом исследования являются деятельности по продаже билетов автовокзалов.

Методами исследования предметной области являются количественный и качественный анализ полученной информации, необходимой для реализации базы данных.

Курсовая работа состоит из теоретического раздела, в котором представлены общие сведения о деятельности автовокзалов, и проектного, в котором непосредственно представлена разработка БД.

Ключевые слова: база данных (далее – БД), концептуальное проектирование, система управления базами данных (далее – СУБД), таблицы, предметная область, физическая модель, реляционная база данных (далее – РБД), схема данных, отчёты.

**Оглавление**

[1 Технический проект 6](#_Toc137848048)

[1.1 Анализ и описание предметной области информационной системы 6](#_Toc137848049)

[1.2 Проектирование концептуальной модели 6](#_Toc137848052)

[1.2.1 Построение логической модели, схемы базы данных 6](#_Toc137848055)

[1.2.2 Разработка диаграммы вариантов использования 8](#_Toc137848056)

[1.3 Выбор СУБД 10](#_Toc137848060)

[2 Реализация базы данных 12](#_Toc137848061)

[2.1 Написание исходного кода БД 12](#_Toc137848062)

[2.2 Написание запросов 14](#_Toc137848063)

[2.2.1 Представление «Отчёт о продажах билетов за последние 2 месяца» 14](#_Toc137848067)

[2.2.2 Процедура «Обновления маршрутов» 15](#_Toc137848068)

[2.2.3 Функция «Цена поездки» 15](#_Toc137848069)

[2.2.4 Триггер «Обновление,добавление и удаление цены» 16](#_Toc137848070)

[3 Заключение 18](#_Toc137848071)

**Введение:**

В современном мире, где скорость и комфорт являются важными факторами в повседневной жизни, автобусные перевозки становятся все более популярным способом перемещения как внутри страны, так и за ее пределами. С увеличением спроса на автобусные билеты возникает необходимость в эффективной системе продажи, которая бы обеспечивала удобство для пассажиров и эффективность для автовокзалов. В связи с этим, разработка базы данных (БД) для продажи билетов автовокзалов становится актуальной и востребованной задачей.

Цель данной курсовой работы заключается в разработке БД, которая позволит автовокзалам автоматизировать процесс продажи билетов и управление данными о расписании рейсов, маршрутах, доступных местах, а также информацию о пассажирах. Разработка эффективной БД способствует повышению качества обслуживания пассажиров и оптимизации внутренних процессов автовокзалов.

Актуальность данной темы заключается в том, что эффективная система продажи билетов является неотъемлемой частью организации автобусных перевозок. Точность и скорость обработки заказов, учет доступных мест, предоставление актуальной информации о расписании и ценах — все это играет решающую роль в привлечении и удержании клиентов. Разработка БД специально для продажи билетов автовокзалов позволяет автоматизировать эти процессы, минимизировать ошибки и повысить качество обслуживания.

Одной из ключевых проблем, с которой сталкиваются автовокзалы, является неэффективное управление продажами билетов. Ручная обработка заказов и использование устаревших систем приводят к длительным очередям, ошибкам в выписке билетов и недовольству пассажиров. Разработка специализированной БД позволит решить эти проблемы, обеспечивая быстрое и точное оформление билетов, автоматическое учет доступных мест и удобный доступ к информации о расписании и ценах.

В рамках данной работы будет проведено проектирование и разработка БД для продажи билетов автовокзалов, а также реализация необходимых функциональностей для эффективного управления данными. Использование современных методов разработки баз данных и подходящих технологий позволит создать систему, которая повысит эффективность работы автовокзалов и улучшит опыт пассажиров при покупке билетов.

**1 Технический проект**

1. **Анализ и описание предметной области информационной системы**

Предметная область данной информационной системы связана с продажей билетов на автобусные рейсы автовокзалов. Она охватывает все процессы, связанные с организацией и управлением продажей билетов, включая учет расписания рейсов, доступных мест, информацию о пассажирах и ценах.

Важным аспектом предметной области является расписание рейсов автобусов. Оно включает информацию о времени отправления и прибытия, маршруте, остановках, типе автобуса и его вместимости. Учет и обновление расписания являются ключевыми задачами системы, чтобы предоставлять актуальную информацию о доступных рейсах и помогать пассажирам выбирать наиболее удобные варианты.

Другой важной составляющей предметной области является учет информации о пассажирах. Система должна иметь возможность хранить данные о пассажирах, такие как их имена, контактная информация и паспортные данные. Это позволяет эффективно обрабатывать бронирования, контролировать посадку на рейс и обеспечивать безопасность и удобство пассажиров.

Еще одним важным аспектом является информация о ценах на билеты. Система должна хранить актуальную информацию о ценах на различные маршруты и типы автобусов, а также предоставлять возможность автоматического расчета стоимости билета на основе выбранного рейса и дополнительных услуг.

Анализ предметной области позволяет определить основные сущности и связи между ними, которые должны быть учтены при разработке БД для продажи билетов автовокзалов. Это включает сущности, такие как рейсы, маршруты, места, пассажиры, цены и т.д. Анализ также помогает идентифицировать основные функциональности, которые должны быть реализованы в системе, чтобы обеспечить эффективное управление данными и удовлетворение потребностей пользователей.

1. 2. **Проектирование концептуальной модели**
   4. 1. **Построение логической модели, схемы базы данных**

Представленная концептуальная модель базы данных для продажи билетов автовокзалов включает следующие связи и сущности:

1. Таблица "Address":
   * Атрибуты: id, country, city, street, house.
   * Связи: отсутствуют.
2. Таблица "Bus\_station":
   * Атрибуты: id, phone, address, description, name.
   * Связи: связь с таблицей "Address" по внешнему ключу address.
3. Таблица "Route":
   * Атрибуты: id, first, second.
   * Связи: связь с таблицей "Bus\_station" по внешним ключам first и second.
4. Таблица "Transport":
   * Атрибуты: id, car\_brand, car\_model, count\_place.
   * Связи: отсутствуют.
5. Таблица "State\_trip":
   * Атрибуты: id, name.
   * Связи: отсутствуют.
6. Таблица "State\_Purchase":
   * Атрибуты: id, name.
   * Связи: отсутствуют.
7. Таблица "Trip":
   * Атрибуты: id, route, state, transport, departure, arrival.
   * Связи: связь с таблицей "Route" по внешнему ключу route, связь с таблицей "Transport" по внешнему ключу transport, связь с таблицей "State\_Purchase" по внешнему ключу state.
8. Таблица "Price":
   * Атрибуты: id, trip, date\_begin, date\_end.
   * Связи: связь с таблицей "Trip" по внешнему ключу trip.
9. Таблица "Citizenship":
   * Атрибуты: id, citizenship.
   * Связи: отсутствуют.
10. Таблица "Documents":
    * Атрибуты: id, documents, number.
    * Связи: отсутствуют.
11. Таблица "User":
    * Атрибуты: id, name, firstname, lastname, phone, password, gender, birthdate, citizenship, documents.
    * Связи: связь с таблицей "Citizenship" по внешнему ключу citizenship, связь с таблицей "Documents" по внешнему ключу documents.
12. Таблица "Purchase":
    * Атрибуты: id, trip, date\_purchase, payment\_method, user\_id, place, state\_purchase.
    * Связи: связь с таблицей "Trip" по внешнему ключу trip, связь с таблицей "User" по внешнему ключу user\_id, связь с таблицей "State\_Purchase" по внешнему ключу state\_purchase.

Эта концептуальная модель предоставляет основу для дальнейшей разработки физической модели базы данных и реализации функциональностей информационной системы по продаже билетов автовокзалов.

Таким образом, вы можете построить логическую модель базы данных, используя представленные таблицы и связи между ними. Эта модель будет служить основой для создания схемы БД и дальнейшей реализации информационной системы по продаже билетов автовокзалов (Рисунок 1).

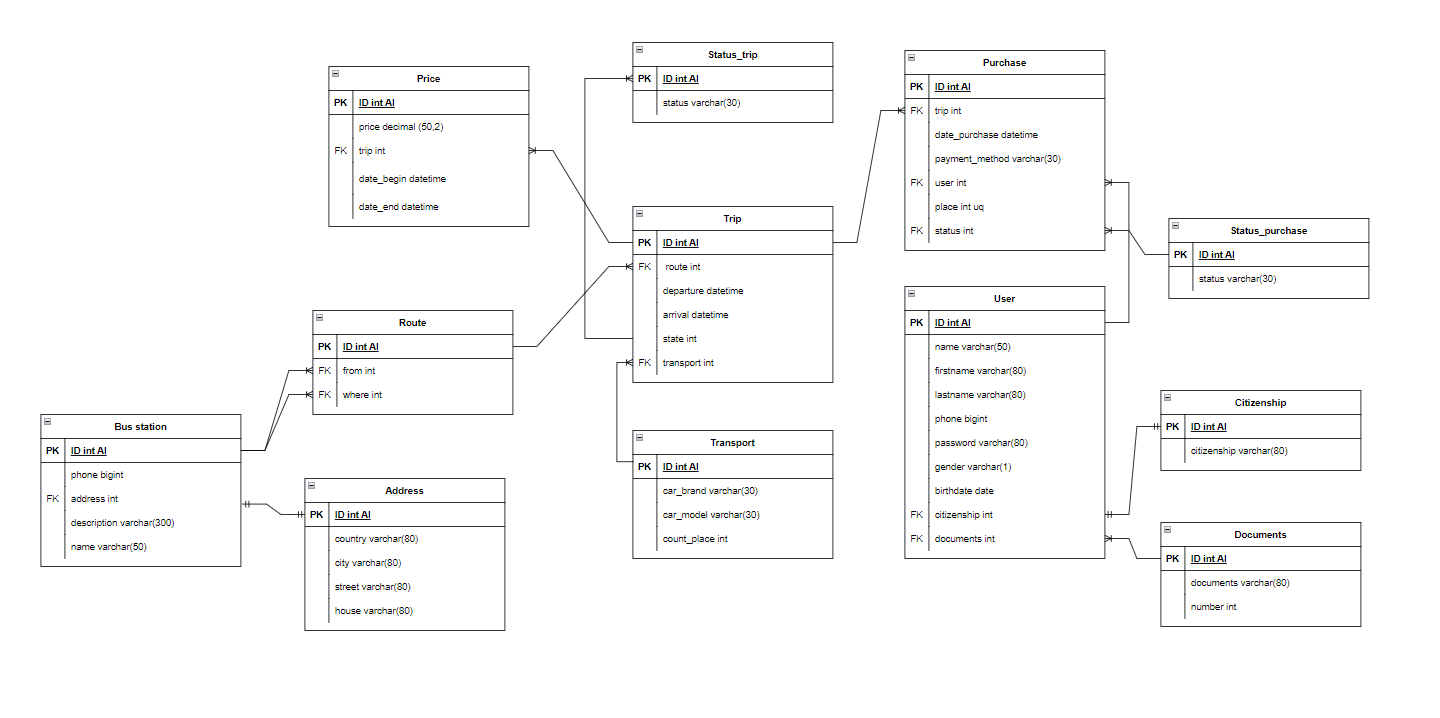


Рисунок 1 – Схема БД

* + 1. **Разработка диаграммы вариантов использования**

Диаграмма вариантов использования представляет собой графическое представление функциональных возможностей системы, описывающее взаимодействие между актерами (пользователями и администратором) и системой. В контексте курсовой работы на тему "Разработка БД по продаже билетов автовокзалов" можно определить следующие основные варианты использования:

* Покупка билета: Пользователь может осуществлять покупку билета через систему. Он выбирает маршрут, указывает дату и время, выбирает количество мест и производит оплату.
* Просмотр расписания: Пользователь и администратор может просматривать расписание автобусных рейсов для выбора удобного времени и маршрута.
* Управление маршрутами: Администратор системы (актер) имеет возможность добавлять, изменять и удалять маршруты, включая информацию о начальном и конечном пункте, времени отправления и прибытия.
* Управление автовокзалами: Администратор может добавлять, изменять и удалять информацию об автовокзалах, включая их адрес, контактные данные и описание.
* Управление транспортными средствами: Администратор может добавлять, изменять и удалять информацию о транспортных средствах, таких как марка, модель и количество мест.
* Генерация отчетов: Администратор может генерировать отчеты о продажах билетов, посещаемости и другой статистической информации для анализа и мониторинга работы системы.

Каждый вариант использования представляет собой отдельную функциональность системы, которая будет реализована через соответствующие функции и процессы в базе данных. Диаграмма вариантов использования помогает визуализировать основные сценарии взаимодействия пользователей и системы и служит основой для разработки функциональности базы данных.

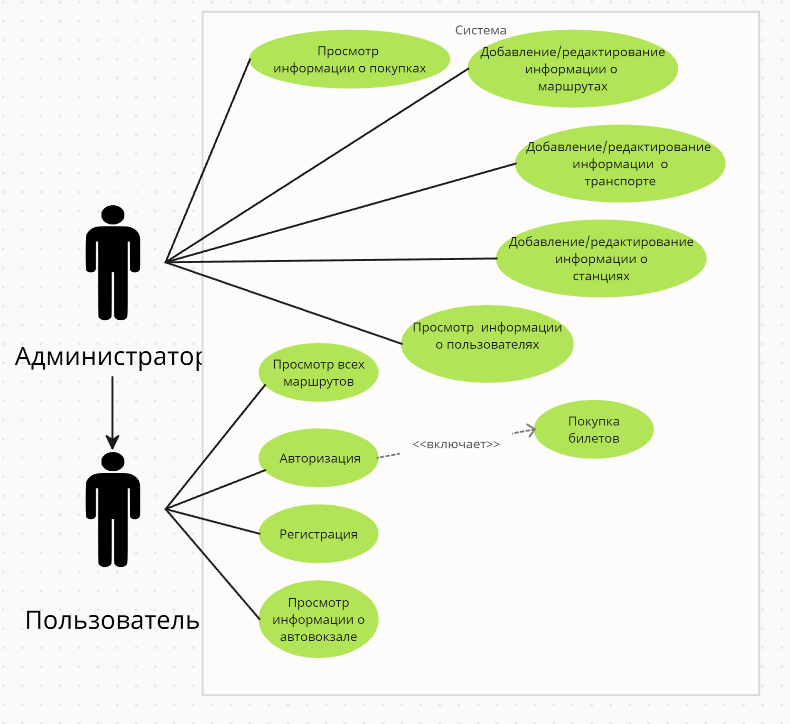
****

Рисунок 2 – Диаграмма вариантов использования

1. 3. **Выбор СУБД**

В контексте разработки базы данных для продажи билетов на автовокзалах, PostgreSQL имеет несколько преимуществ по сравнению с другими СУБД:

* Расширяемость и гибкость: PostgreSQL обладает широким спектром функций и возможностей для моделирования сложных данных. Он поддерживает различные типы данных, включая географические данные и массивы, а также позволяет создавать пользовательские типы данных и функции. Это позволяет более точно отразить структуру данных, связанных с автовокзалами, например, адреса, маршруты, транспортные средства и т.д.
* Надежность и целостность данных: PostgreSQL обеспечивает механизмы транзакций, которые гарантируют целостность данных и предотвращают ошибки при одновременных операциях записи. Это критически важно для системы продажи билетов, где точность данных и их согласованность являются ключевыми факторами.
* Производительность при обработке больших объемов данных: PostgreSQL хорошо справляется с обработкой больших объемов данных и высокими нагрузками. Он поддерживает многопоточность и оптимизирован для работы с параллельными запросами, что обеспечивает эффективное выполнение запросов и быстрый доступ к данным. В случае системы продажи билетов, где может быть большое количество билетов и пользователей, производительность играет важную роль для обеспечения плавной работы системы.
* Поддержка географических данных: PostgreSQL имеет встроенную поддержку географических данных и предоставляет богатый набор функций для работы с ними. Это позволяет легко реализовать функциональности, связанные с маршрутами, геолокацией автовокзалов и другими аспектами, связанными с географией.
* Большое сообщество и экосистема: PostgreSQL имеет активное сообщество пользователей и разработчиков, которые предоставляют поддержку, документацию и решения проблем. Кроме того, существует множество расширений и инструментов, разработанных сообществом, которые могут значительно упростить и ускорить разработку базы данных.

Учитывая эти преимущества, PostgreSQL является хорошим выбором для разработки базы данных для продажи билетов на автовокзалах, обеспечивая надежность, производительность и гибкость для решения задач.

1. **Реализация базы данных**
2. **Написание исходного кода БД**

Ниже приведен пример исходного кода для создания таблиц в базе данных на основе предоставленных схем:

sql

Copy code

-- Создание таблицы Адресы

CREATE TABLE Address (

id SERIAL PRIMARY KEY,

country VARCHAR(80) NOT NULL,

city VARCHAR(80) NOT NULL,

street VARCHAR(80) NOT NULL,

house VARCHAR(80) NOT NULL

);

-- Создание таблицы Автобусные станции

CREATE TABLE Bus\_station (

id SERIAL PRIMARY KEY,

phone BIGINT NOT NULL,

address INT,

description TEXT,

name VARCHAR(80) NOT NULL,

FOREIGN KEY (address) REFERENCES Address (id)

);

-- Создание таблицы Маршрут

CREATE TABLE Route (

id SERIAL PRIMARY KEY,

first INT,

second INT,

FOREIGN KEY (first) REFERENCES Bus\_station (id),

FOREIGN KEY (second) REFERENCES Bus\_station (id)

);

-- Создание таблицы Транспорт

CREATE TABLE Transport (

id SERIAL PRIMARY KEY,

car\_brand VARCHAR(80) NOT NULL,

car\_model VARCHAR(80) NOT NULL,

count\_place INT

);

-- Создание таблицы Статус поездки

CREATE TABLE State\_trip (

id SERIAL PRIMARY KEY,

name VARCHAR(80)

);

-- Создание таблицы Статус покупки

CREATE TABLE State\_Purchse (

id SERIAL PRIMARY KEY,

name VARCHAR(80)

);

-- Создание таблицы Поездка

CREATE TABLE Trip (

id SERIAL PRIMARY KEY,

route INT,

state INT,

transport INT,

departure TIMESTAMP,

arrival TIMESTAMP,

FOREIGN KEY (route) REFERENCES Route (id),

FOREIGN KEY (transport) REFERENCES Transport (id),

FOREIGN KEY (state) REFERENCES State\_Purchse (id)

);

-- Создание таблицы Цены

CREATE TABLE Price (

id SERIAL PRIMARY KEY,

trip INT,

date\_begin TIMESTAMP,

date\_end TIMESTAMP,

FOREIGN KEY (trip) REFERENCES Trip (id)

);

-- Создание таблицы Гражданство

CREATE TABLE Сitizenship (

id SERIAL PRIMARY KEY,

citizenship VARCHAR(80)

);

-- Создание таблицы Документы

CREATE TABLE Documents (

id SERIAL PRIMARY KEY,

documents VARCHAR(80),

number INT

);

-- Создание таблицы Пользователь

CREATE TABLE "user" (

id SERIAL PRIMARY KEY,

name VARCHAR(50),

firstname VARCHAR(80),

lastname VARCHAR(80),

phone BIGINT,

password VARCHAR(80),

gender VARCHAR(1),

birthdate DATE,

citizenship INT,

documents INT,

FOREIGN KEY (citizenship) REFERENCES Сitizenship (id),

FOREIGN KEY (documents) REFERENCES Documents (id)

);

-- Создание таблицы Покупки

CREATE TABLE Purchase (

id SERIAL PRIMARY KEY,

trip INT,

date\_purchase TIMESTAMP,

payment\_method VARCHAR(30),

user\_id INT,

place INT,

state\_purchse INT,

FOREIGN KEY (trip) REFERENCES Trip (id),

FOREIGN KEY (user\_id) REFERENCES "user" (id),

FOREIGN KEY (state\_purchse) REFERENCES State\_Purchse (id)

);

Этот код может быть использован для создания таблиц в базе данных, и они будут готовы к использованию для информационной системы по продаже билетов автовокзалов.

1. **Написание запросов**
2. 2. 1. **Представление «Отчёт о продажах билетов за последние 2 месяца»**

Данное представление составляет таблицу продаж маршрутов за последние 2 месяца.

Запрос:

SELECT x.\* FROM public."Month\_analytics" x

Тело запроса:

select \* from crosstab(

$$select p.trip ,to\_char( date\_trunc('month',p.date\_purchase),'TMMonth'),sum( price(p.trip,'2023-10-06 11:27:00.000'))

from trip t

right join purchase p on t.id = p.id

where p.date\_purchase >= date\_trunc('month',current\_date - interval '3 month' )

group by p.trip,date\_trunc('month',p.date\_purchase)

order by p.trip,date\_trunc('month',p.date\_purchase)$$

) as ct(trip int,"1" decimal,"2" decimal)

* + 1. **Процедура «Обновления маршрутов»**

Данная процедура обновляет маршрут по его «id» (route), заменяя начало маршрута (first) на новую переменную «a», конец маршрута (second) на новую переменную «b».

Вызов процедуры:

call update\_route(1,1,100)

Тело процедуры:

CREATE PROCEDURE Update\_Route (route integer, a integer, b integer)

LANGUAGE SQL

AS $$

update public.route

set first = a,

second = b

where id=$1;

$$;

* + 1. **Функция «Цена поездки»**

Данная функция возвращает по «id» поездки и даты начала этой цены(price.date\_begine) саму цену (price.price).

Вызов функции:

select price(1,'2023-10-06 11:27:00.000')

Тело процедуры:

CREATE OR REPLACE FUNCTION public.price(integer, timestamp without time zone)

RETURNS numeric

LANGUAGE sql

AS $function$select price.price from price

where price.trip = $1 and price.date\_begine = $2;

$function$

;

* + 1. **Триггер «Обновление,добавление и удаление цены»**

Данный триггер добавляет в таблицу price\_log новую и старую цену после обновление, добавление и удаление цены в таблице price.

Запрос:

CREATE TABLE price\_log(

operation char(1) NOT NULL,

stamp timestamp NOT NULL,

userid text NOT NULL,

empname int,

new\_price integer,

old\_price integer

);

CREATE OR REPLACE FUNCTION process\_price\_log() RETURNS TRIGGER AS $price\_log$

BEGIN

IF (TG\_OP = 'DELETE') THEN

INSERT INTO price\_log SELECT 'D', now(),user,old.id,null,old.price ;

RETURN OLD;

ELSIF (TG\_OP = 'UPDATE') THEN

INSERT INTO price\_log SELECT 'U', now(),user,old.id,new.price,old.price;

RETURN NEW;

ELSIF (TG\_OP = 'INSERT') THEN

INSERT INTO price\_log SELECT 'I', now(),user,old.id,new.price,null;

RETURN NEW;

END IF;

RETURN NULL;

END;

$price\_log$ LANGUAGE plpgsql;

CREATE TRIGGER price\_log

AFTER INSERT OR UPDATE OR DELETE ON price

FOR EACH ROW EXECUTE PROCEDURE process\_price\_log()

1. **Заключение**

В заключении курсовой работы можно подвести итоги проведенного исследования и работы над проектом, а также описать основные результаты и выводы.

В ходе выполнения курсовой работы были достигнуты следующие цели:

* Изучение предметной области: Была проведена аналитическая работа по изучению процесса продажи билетов на автовокзалах. Были выявлены основные требования и функциональные возможности системы, а также определены связи и сущности в базе данных.
* Проектирование базы данных: Была разработана концептуальная и логическая модели базы данных с учетом специфики предметной области. Были определены таблицы, их структура и связи между ними.
* Разработка и реализация базы данных: На основе проектирования была создана база данных, используя СУБД PostgreSQL. Были созданы необходимые таблицы и настроены соответствующие связи и ограничения.
* Тестирование и проверка работоспособности: Были проведены тесты и проверка функциональности разработанной базы данных. Результаты тестирования подтвердили корректность работы системы и соответствие требованиям.

Выводы и рекомендации: В результате выполнения курсовой работы были сделаны следующие выводы:

* Разработка базы данных позволяет эффективно организовать процесс продажи билетов на автовокзалах, обеспечивая точное хранение информации о маршрутах, расписании, транспортных средствах и пользователях.
* Использование СУБД PostgreSQL обеспечивает надежность, производительность и гибкость базы данных для удовлетворения требований предметной области.
* Разработанная база данных может быть дальше расширена и оптимизирована для учета дополнительных функциональностей и повышения удобства использования.

В целом, выполнение курсовой работы по разработке базы данных для продажи билетов автовокзалов является успешным шагом в решении задач данной предметной области. Разработанная база данных предоставляет удобный и надежный инструмент для автоматизации процесса продажи билетов и управления информацией о маршрутах и транспортных средствах.