# Базы данных

**Практическая работа № 8 «Проектирование баз данных на основе ER-модели.»**

## **Цель:**

Установка и настройка свободно распространяемой СУБД PostgreSQL.

## **Задание.**

## Физическая реализация концептуальной схемы в виде логической схемы БД с помощью, установленной СУБД.

## **Ход выполнения.**

**PostgreSQL** — это объектно-реляционная система управления базами данных (СУБД) с открытым исходным кодом. Она поддерживает расширения, сложные запросы, транзакции, правила ссылочной целостности и пользовательские функции. PostgreSQL широко используется благодаря высокой надежности, расширяемости и поддержке множества типов данных.

**pgAdmin III** — это графический интерфейс для управления PostgreSQL. Он предоставляет удобные инструменты для создания, модификации и управления базами данных, выполнения SQL-запросов и анализа структуры БД.

**Создание базы данных "Прокат"**

CREATE DATABASE Прокат;

Определение БД:

* Имя базы данных: Прокат
* Кодировка: по умолчанию используется UTF-8.
* Владелец базы данных: текущий пользователь.

**Создание схемы "Прокат"**

CREATE SCHEMA Прокат;

Определение схемы:

* Имя схемы: Прокат
* Назначение: логическая структура для группировки объектов БД (таблиц, доменов и других).

**Создание доменов**

Пример запросов для создания доменов:

CREATE DOMAIN domain\_telephone AS VARCHAR(15) CHECK (VALUE ~ '^\+?[0-9\-]+$');

CREATE DOMAIN domain\_price AS NUMERIC(10, 2) CHECK (VALUE >= 0);

CREATE DOMAIN domain\_date AS DATE;

Определение доменов:

* domain\_telephone — домен для телефонных номеров (макс. 15 символов, только цифры, + и -).
* domain\_price — домен для цены (макс. 10 разрядов, 2 десятичных знака).
* domain\_date — домен для хранения даты.

**Создание таблиц**

Пример создания таблицы (без связей):

CREATE TABLE Прокат.Клиенты (

id SERIAL PRIMARY KEY,

имя VARCHAR(50) NOT NULL,

телефон domain\_telephone NOT NULL,

адрес VARCHAR(100)

);

Определение таблицы:

* id — первичный ключ (уникальный идентификатор клиента).
* имя — строка, обязательное поле.
* телефон — домен domain\_telephone.
* адрес — необязательное поле для хранения строки.

**Установка связей между таблицами**

Пример связи таблиц Клиенты и Заказы:

CREATE TABLE Прокат.Заказы (

id SERIAL PRIMARY KEY,

клиент\_id INT REFERENCES Прокат.Клиенты(id) ON DELETE CASCADE,

дата domain\_date NOT NULL,

сумма domain\_price NOT NULL

);

Объяснение связей:

* Поле клиент\_id связывает таблицу Заказы с таблицей Клиенты по id.
* При удалении клиента (ON DELETE CASCADE) автоматически удаляются все его заказы.

**Определение правил ссылочной целостности**

Пример настройки целостности для таблицы Автомобили и Аренда:

CREATE TABLE Прокат.Автомобили (

id SERIAL PRIMARY KEY,

модель VARCHAR(50) NOT NULL,

цена\_за\_день domain\_price NOT NULL

);

CREATE TABLE Прокат.Аренда (

id SERIAL PRIMARY KEY,

автомобиль\_id INT REFERENCES Прокат.Автомобили(id) ON DELETE RESTRICT,

клиент\_id INT REFERENCES Прокат.Клиенты(id),

дата\_начала domain\_date NOT NULL,

дата\_окончания domain\_date,

CHECK (дата\_окончания > дата\_начала)

);

Объяснение целостности:

* В Аренда проверка дат (CHECK), чтобы дата окончания была больше даты начала.
* Поле автомобиль\_id связано с Автомобили.id, при этом нельзя удалить автомобиль, если он используется в аренде (ON DELETE RESTRICT).

Создание и выполнение задания:

