

Министерство науки и высшего образования Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО

ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ № 3
Создание таблиц базы данных PostgreSQL. Заполнение»
«таблиц рабочими данными
«по дисциплине «Проектирование и реализация баз данных

Обучающийся Саид Наваф

Факультет прикладной информатики

Группа K3241

Направление подготовки 09.03.03

Образовательная программа Мобильные и сетевые технологии

Преподаватель Говорова Марина Михайловна

Санкт-Петербург

2025-2026

1. Цель работы:

Овладеть практическими навыками создания таблиц базы данных PostgreSQL 1X, заполнения их рабочими данными, резервного копирования и восстановления БД.

Практическое задание:

1. Создать базу данных с использованием pgAdmin 4 (согласно индивидуальному заданию).
2. Создать схему в составе базы данных.
3. Создать таблицы базы данных.
4. Установить ограничения на данные: Primary Key, Unique, Check, Foreign Key.
5. Заполнить таблицы БД рабочими данными.
6. Создать резервную копию БД.

Указание:

Создать две резервные копии:

- с расширением CUSTOM для восстановления БД;
 - с расширением PLAIN для листинга (в отчете);
 - при создании резервных копий БД настроить параметры Dump options для Type of objects и Queries .
7. Восстановить БД.

3. Индивидуальное задание:

Вариант 12. «Прокат автомобилей»

Описание предметной области: Компания предоставляет прокат автомобилей. В офис обращаются клиенты, данные о которых регистрируют в базе. Цена проката зависит от марки автомобиля, технических характеристик и года выпуска.

Для проката авто с клиентом заключается договор, в котором фиксируется период проката, вид страховки, стоимость страховки, залоговая стоимость. Стоимость страховки и залоговая стоимость зависят от марки авто.

Залоговая стоимость возвращается полностью или частично клиенту, в зависимости от наличия аварий и штрафов в период действия договора. Если залоговая стоимость уже возвращена клиенту, но на авто в компанию пришел штраф, то он оплачивается компанией, а не клиентом.

При передаче авто клиенту составляется акт о передаче в аренду. При возвращении автомобиля также составляется акт о возврате авто из аренды.

Если клиент не вернул автомобиль в срок и не оформил продление, ему назначается штраф за каждый час просрочки.

Постоянным клиентам предоставляются скидки.

В системе необходимо хранить историю нарушений (со штрафами за вид нарушения ПДД) и аварий автомобилей. Нарушение может быть совершено и во время аварии. Необходимо хранить информацию, кто оплачивает штраф: компания или клиент.

Цены на прокат автомобилей могут меняться.

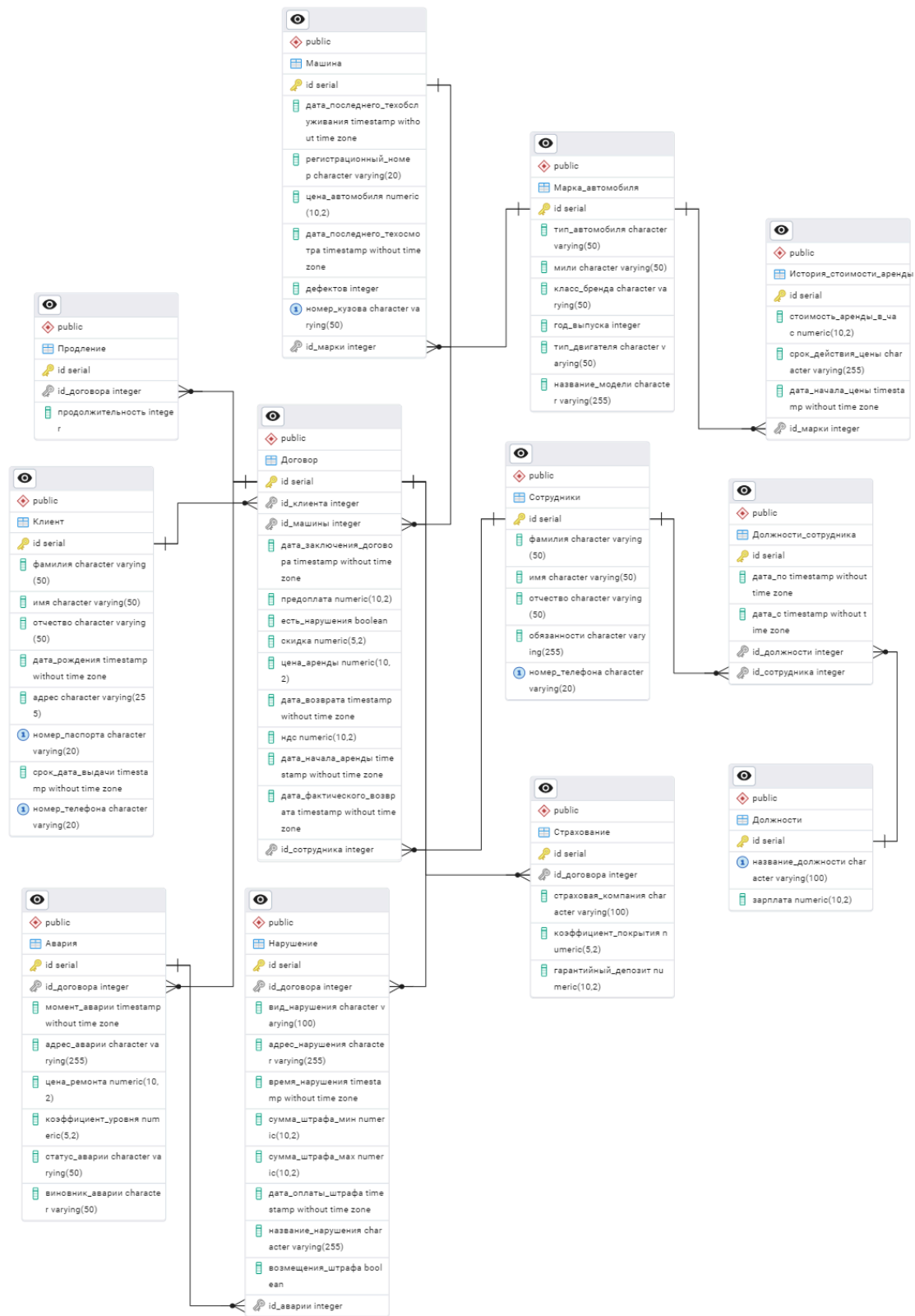
БД должна содержать следующий минимальный набор сведений: ФИО клиента. Паспортные данные. ФИО менеджера. Код должности. Наименование должности. Оклад. Обязанности. Код марки. Наименование. Технические характеристики. Описание. Код автомобиля. Регистрационный номер. Номер кузова. Номер двигателя. Год выпуска. Пробег. Цена автомобиля. Цена проката. Дата последнего ТО. Специальные отметки. Отметка о возврате. Код клиента. ФИО. Адрес. Телефон. Паспортные данные. Дата и время выдачи автомобиля. На сколько часов. Дата и время возврата автомобиля. Данные о нарушениях. Данные об авариях. Дата продления. Часов продления.

Дополните состав атрибутов на основе анализа предметной области.

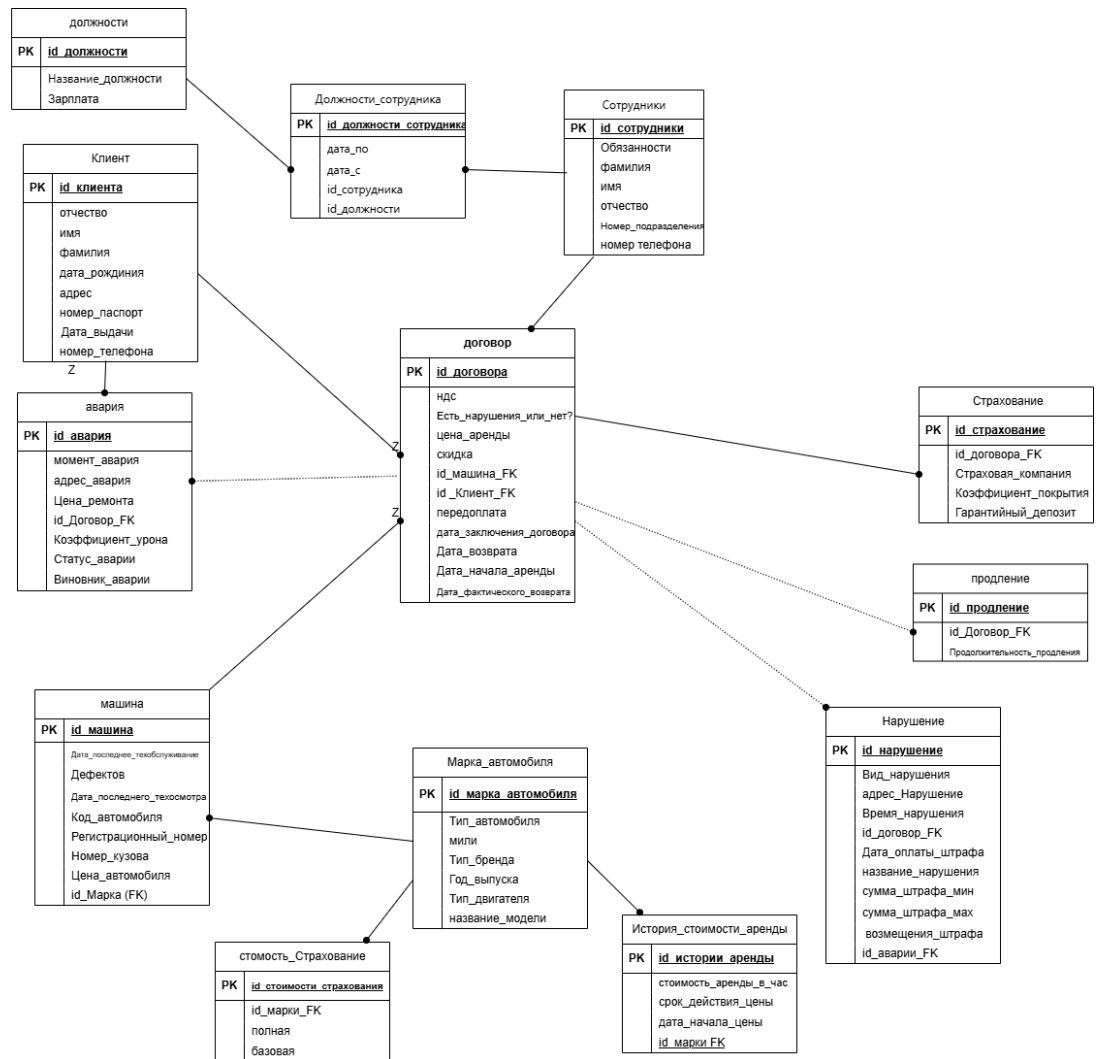
4. Выполнение:

1. Наименование БД: car rental

2. Схема логической модели базы данных, сгенерированная в Generate ERD изображена на картинке 1



Картинка 1 - Схема ИЛМ, сгенерированная в Generate ERD



Картинка 2 - Схема ИЛМ в нотации IDEF1X

3. Работа с резервными копиями БД.

- Создание резервной копии с расширением .custom:

-U postgres → подключение под пользователем postgres

-F c → формат дампа "custom" (сжатый, двоичный)

-f backup_custom.dump → имя выходного файла дампа

--create → включает команду CREATE DATABASE в дамп

pg_dump -U postgres -F c -f backup_custom.dump car_rental --create

- Создание дампа базы данных car_rental в формате PLAIN (обычный SQL)

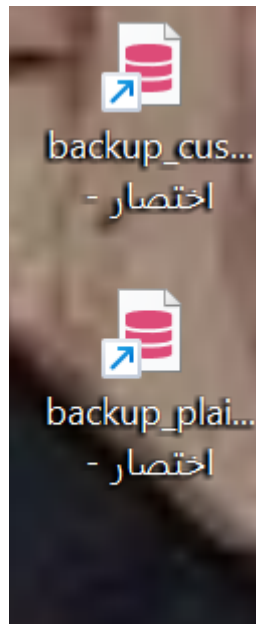
-F p → формат plain (текстовый .sql)

-f backup_plain.sql → имя выходного файла

--create → включает команду CREATE DATABASE

```
pg_dump -U postgres -F p -f backup_plain.sqlc car_rental --create
```

Итоговые файлы представлены на картинке 3:



Картинка 3 – Резервные копии баз данных

- Восстановление из .custom:

Создание новой базы данных, в которую будет загружен дамп

```
createdb -U postgres restored_db
```

Восстановление базы данных из дампа формата CUSTOM

-d restored_db → целевая база данных

-v → подробный (verbose) режим

```
pg_restore -U postgres -d restored_db -v backup_custom.dump
```

- Восстановление из .sql:

-d restored_db → база, в которую будет загружен SQL

```
psql -U postgres -d restored_db -f backup_plain.sql
```

4. Вывод

В ходе выполнения лабораторной работы были достигнуты следующие цели:

- Созданы таблицы, включающих:
 - Определение первичных и внешних ключей;
 - Ограничения целостности данных (NOT NULL, CHECK, FOREIGN KEY).
- Таблицы были успешно заполнены тестовыми рабочими данными при помощи SQL-запросов INSERT INTO.
- Проверена корректность структуры базы данных и целостность данных с помощью запросов SELECT.

В результате выполнения лабораторной работы были закреплены практические навыки проектирования и создания таблиц в PostgreSQL, определения связей между таблицами, а также первичной и тестовой загрузки данных.