

**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** Недиков Михаил Олегович

**Факультет** прикладной информатики

**Группа** K3239

**Направление подготовки** 09.03.03 Прикладная информатика

**Образовательная программа** Мобильные и сетевые технологии 2025

**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.

## 1. 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.

## 2. Индивидуальное задание:

Вариант 19. БД «Банк»

Описание предметной области: Система обеспечивает работу с вкладами и кредитами клиентов банка.

Клиенты банка имеют вклады и кредиты различных видов. Для вкладов и кредитов может использоваться различная валюта.

Сотрудники банка заключают договоры с клиентами. Фиксируется сотрудник, заключивший договор.

Ежемесячно начисляется процент по вкладу, и полученная сумма добавляется к сумме вклада заказчика. Вкладчик имеет право снимать проценты по вкладу или всю сумму вклада с процентами по истечении срока вклада. При снятии денег до истечения срока вклада процент за текущий месяц не начисляется.

Кредит выдается на определенный срок. Формируется график выплат, который получает клиент при заключении договора, в котором ежемесячно указывается сумма выплаты по вкладу и сумма выплаты по процентам банку. Хранится информация по своевременности ежемесячных выплат.

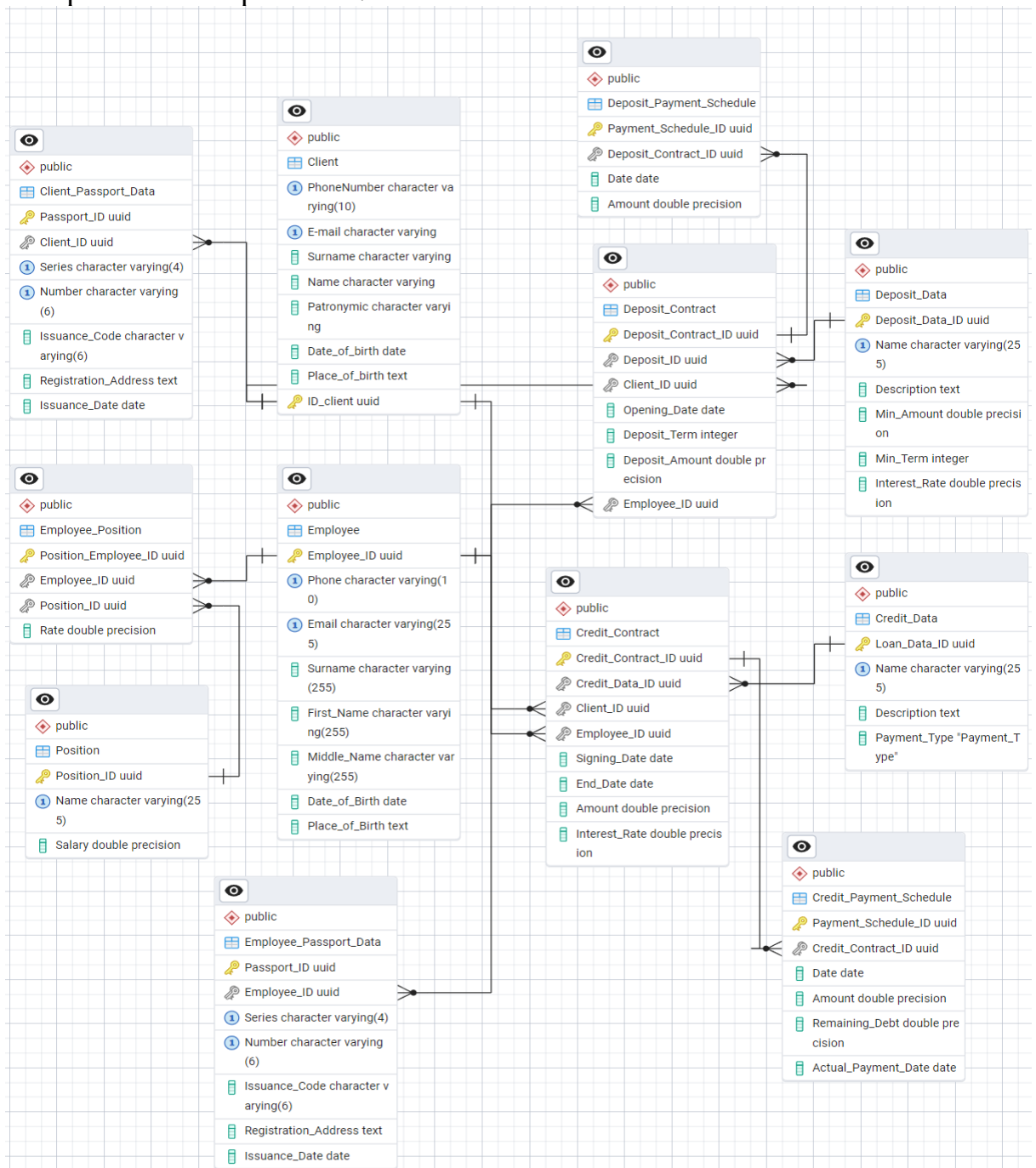
БД должна содержать следующий минимальный набор сведений: ФИО сотрудника. Возраст сотрудника. Адрес сотрудника. № телефона сотрудника. Паспортные данные сотрудника. Должность сотрудника. Оклад сотрудника (зависит от должности). Наименование вклада. Описание вклада. Минимальный срок вклада. Минимальная сумма вклада. Процент по вкладу. Срок вклада. Процентная ставка. Код валюты. Наименование валюты. ФИО вкладчика. Адрес вкладчика. Телефон

клиента. Е-mail вкладчика. Паспортные данные. Номер договора. Дата вклада. Дата возврата. Сумма вклада. Сумма возврата. Данные по кредиту. Число выплаты ежемесячно (нельзя указывать 29, 30 и 31). Должность сотрудника. Количество ставок (по штатному расписанию).

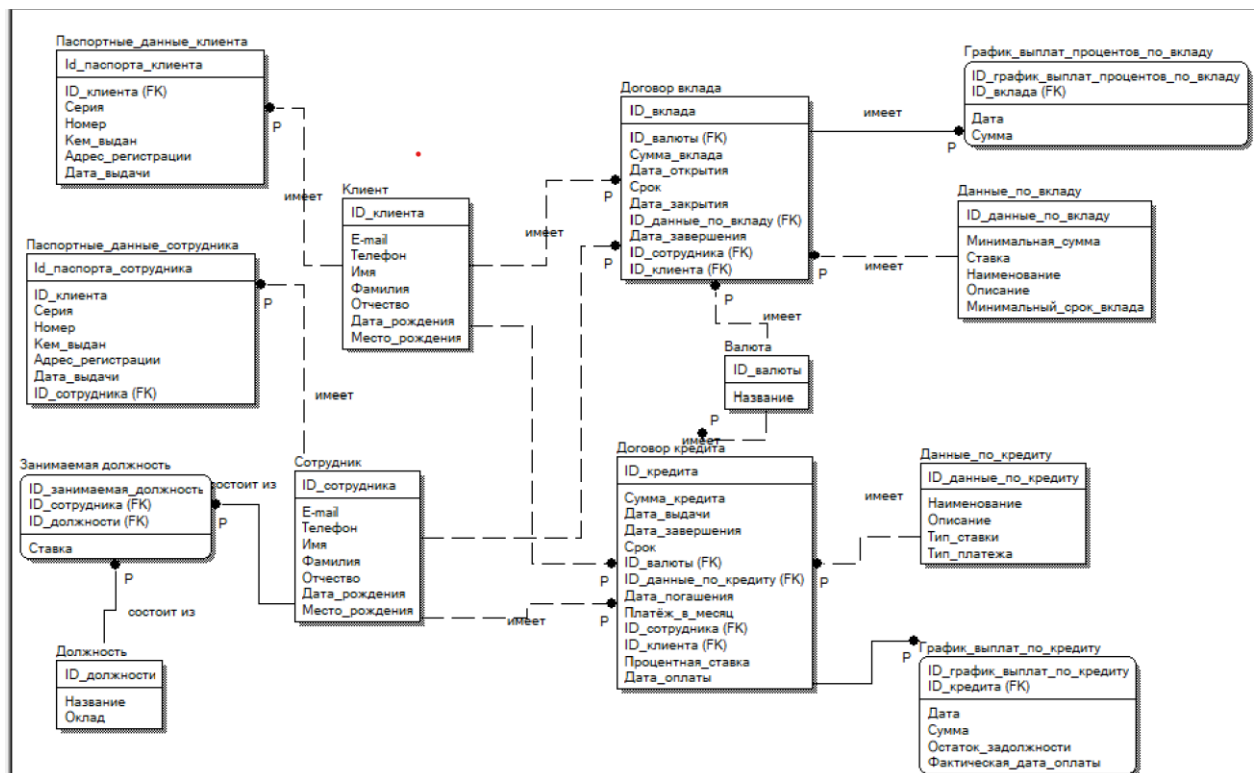
Дополните состав атрибутов на основе анализа предметной области.

### 3. Выполнение:

1. Наименование БД: bank\_sytem
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.

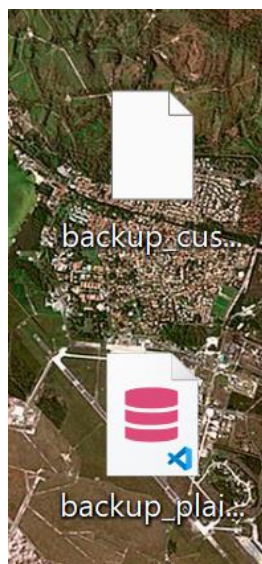
- Создание резервной копии с расширением .custom:

```
# -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 bank_system --create
```

- Создание дампа базы данных bank\_system 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.sql bank_system --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.