

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ  
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»**

**МОСКОВСКИЙ ИНСТИТУТ ЭЛЕКТРОНИКИ И МАТЕМАТИКИ  
им. А.Н. ТИХОНОВА**

Отчет по домашней работе по дисциплине «Базы данных»

Тема работы: «База данных фитнес-клуба»

Выполнил: студент БИВ-184 Ковалев В.С.

Руководитель: доцент Карпова И.П.  
департамент компьютерной инженерии

## Оглавление

Инфологическое проектирование .....	3
Анализ предметной области: .....	3
Пользователи системы и их функции: .....	3
Требования к техническому обеспечению .....	4
Выбор СУБД .....	4
Логическое проектирование .....	4
Сущности: .....	4
ER-диаграмма: .....	5
Преобразование ER–диаграммы в схему базы данных .....	5
Нормализация отношений .....	7
Физическое проектирование .....	12
Листинг таблиц: .....	12
Создание готовых представлений: .....	15
Определение прав доступа: .....	16
Таблицы для визуализации доступа user'ов: .....	16
Создание индексов .....	17
Резервное копирование .....	17

## **Инфологическое проектирование**

### **Анализ предметной области:**

- Каждый клиент может тренироваться персонально или в группе
- Каждый тренер может вести и групповые, и персональные тренировки
- Тренеру не обязательно иметь высшего образования, но к проведению тренировок у клиентов младше 18 лет допускаются только тренеры с высшим образованием
- У одного тренера может быть несколько клиентов
- У клиента может быть несколько тренеров по разным видам тренировок (набор массы и бассейн или медитация и массаж)
- У каждой единицы инвентаря есть ответственный за нее человек.
- В спортивном зале основным видом инвентаря является фитнес-тренажер. Срок действия любого тренажера-5 лет с регулярным осмотром раз в 6 месяцев
- Персоналом фитнес-клуба являются инструкторы зала, тренеры, массажисты, подсобные рабочие, администраторы рецепции, менеджеры по продажам, доктора и управляющий

### **Пользователи системы и их функции:**

1. Директор: осуществляет организацию и контроль функционирования бизнес-процессов компании. Имеет право на получение всей необходимой для анализа информации.
2. Менеджер: осуществляет постоянное взаимодействие между клиентами и тренерами. Имеет право на получение всей необходимой информации о тренерах. Занимается регистрацией и удалением клиентов.
3. Тренер: основная функция- оказание услуг фитнес-наставника для клиентов. Имеет право получать список клиентов по направлениям, где он оказывает услуги.
4. Администратор: осуществляет все необходимые работы для поддержания и целостного функционирования базы данных.

## Требования к техническому обеспечению

Для проектируемой базы данных объем данных не столь велик, поэтому никаких определенных требований к памяти компьютера не предъявляется.

## Выбор СУБД

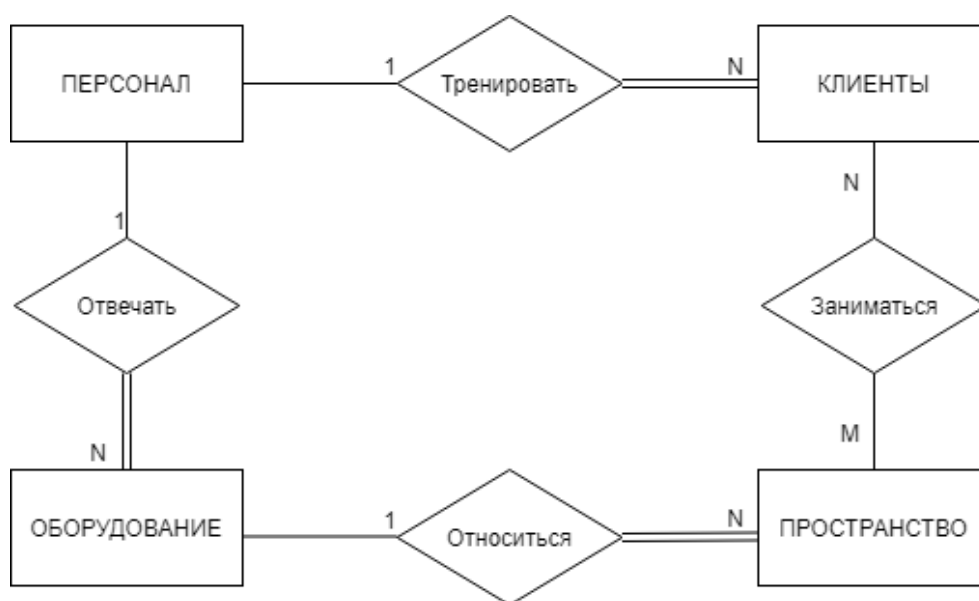
Для выдвигаемых задач подходит СУБД PostgreSQL, так как она поддерживает реляционную модель данных и является общедоступной на основе свободного лицензионного соглашения.

## Логическое проектирование

### Сущности:

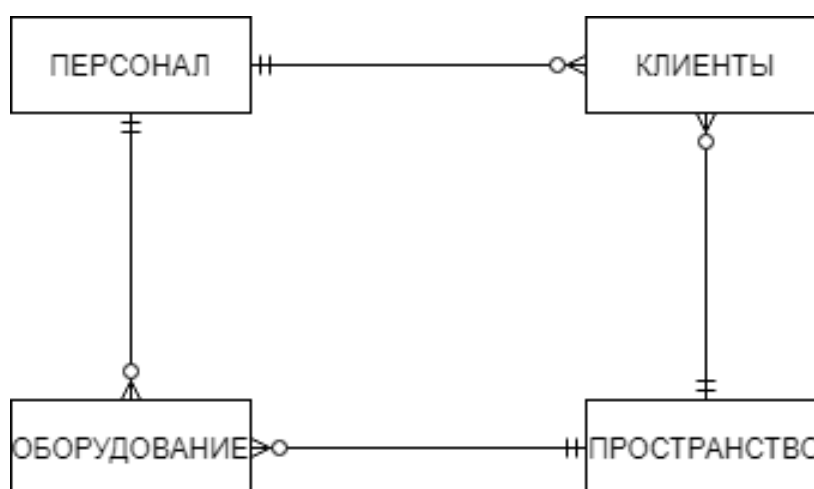
1. **Персонал.** Атрибуты: ФИО, дата рождения, паспорт, образование, телефон, почтовый ящик, занимаемая должность, заработная плата
2. **Клиент.** Атрибуты: ФИО, дата рождения, телефон, выбранные направления подготовки, дата покупки абонемента, срок действия абонемента, тип абонемента
3. **Оборудование.** Атрибуты: тип, модель, начало владения (дата поступления в эксплуатацию), дата последнего осмотра(ремонта),
4. **Пространство.** Атрибуты: название, тип, площадь, расходы на инфраструктуру (электричество, водоснабжение и т.д.)

## ER-диаграмма:



## Преобразование ER-диаграммы в схему базы данных

### Схема база данных:



Ниже представлены схемы отношений. Для каждого указаны атрибуты, первичные и внешние ключи. В схемах отношений используются следующие сокращения для типов данных: N-числовой, V-символьный, D-дата, I-временной интервал, Т-время.

### Схема отношения Персонал(Employee)

Содержание поля	Имя поля	Тип, длина	Примечание
Идентификатор	ID	N(3)	Суррогатный первичный ключ
ФИО	FIO	V(100)	Обязательное поле
Дата рождения	BIRTH	D	Обязательное поле
Пол	GENDER	C(1)	Обязательное поле, «м» или «ж»
Паспортные данные	PASSPORT	V(100)	Обязательное уникальное поле
ИНН	INN	V(100)	Обязательно уникальное поле
Образование	EDU	V(300)	Обязательное многозначное поле
Специальность	SPECIALITY	V(50)	Обязательное поле
Должность	POSITION	V(50)	Обязательное поле
Зарплата	SALARY	N(6)	Обязательное поле
Рабочий телефон	PHONE	V(15)	Обязательное поле
Рабочий e-mail	MAIL	V(30)	Обязательное поле

#### Схема отношения Клиент(Client)

Содержание поля	Имя поля	Тип, длина	Примечание
Идентификатор	ID	N(3)	Суррогатный первичный ключ
ФИО	FIO	V(100)	Обязательное поле
Дата рождения	BIRTH	D	Обязательное поле
Телефон	PHONE	V(30)	Обязательное поле
Направления подготовки	TRAIN	V(50)	Обязательное многозначное поле
Тренер	EMP	N(3)	Внешний ключ к Employee
Тип тренировки	TYPE	V(20)	Обязательное поле («индивидуальная», «групповая»)
Стоимость абонемента	COST	N(5)	Обязательное поле
Дата покупки абонемента	DATE_SUBS	D	Обязательное поле
Срок действия абонемента	TIME_SUBS	I	Обязательное поле
Пространство	SPACE_ID	V(20)	Внешний ключ к Space

#### Схема отношения Оборудование (Equipment)

Содержание поля	Имя поля	Тип, длина	Примечание
Тип	TYPE	V(20)	Обязательное поле
Модель	MODEL	V(20)	Первичный ключ
Начало владения	START_OWN	D	Обязательное поле, не более 5 лет
Дата последнего осмотра(ремонта)	LAST_REPAIR	D	Обязательное поле, не более 6 месяцев
Место нахождения	SPACE	V(20)	Внешний ключ к Space
Ответственный	RESP_ID	N(3)	Внешний ключ к Employee

#### Схема отношения Пространство (Space)

Содержание поля	Имя поля	Тип, длина	Примечание
Название	NAME	V(20)	Первичный ключ
Тип	TYPE	V(20)	Обязательное поле
Площадь	AREA	N(4)	Обязательное поле
Расходы	EXPENCES	N(6)	Обязательное поле

### Нормализация отношений

1НФ: В каждой ячейке таблиц должно находиться единственное значение. Декомпозируем многозначные поля. Разделяем многозначный атрибут «Паспортные данные» на несколько однозначных: «Номер паспорта», «Кем выдан паспорт», «Когда выдан паспорт». Избавляемся от дублирования данных, поэтому создаются 2 новых отношения: «Направления подготовки» (Exercises) и «Типы тренировок» (Types\_of\_Train), для того, чтобы ссылаться на записи этих отношений с помощью внешних ключей. Групповая и индивидуальные тренировки представляют собой разные типы связи. Создадим отдельно отношение «Индивидуальные тренировки». Для групповых тренировок с типом связи «многие-ко-многим» создадим 2 отношения: «Групповые тренировки» и «Участие в групповых тренировках», где будут храниться наборы первичных ключей отношений «Клиенты» и «Групповые тренировки».

2НФ: Необходимо, чтобы отношение находилось в 1НФ, а также каждый не ключевой атрибут функционально полно зависел от составного первичного ключа. Во всех отношениях простой первичный ключ, значит они уже приведены к 2НФ.

3НФ: Атрибут «Зарплата» зависит от атрибута «Должность» в отношении Персонал (Employee), а не от первичного ключа, поэтому выносим в отдельное отношение Должности(Jobs). Атрибут «специальность» зависит от атрибута «Образование» в отношении Персонал (Employee), поэтому выносим в новое отношение Образование(Education). Атрибут «стоимость абонемента» в отношении Клиенты зависит от атрибутов «Направление подготовки» и «Тип тренировки», а не от первичного ключа, поэтому выносим в отдельное отношение Прейскурант(Prices).

Таким образом, после нормализации:

Схема отношения Персонал (Employee)

Содержание поля	Имя поля	Тип, длина	Примечание
Идентификатор	ID	N(3)	<b>Суррогатный первичный ключ</b>
Фамилия	F_NAME	V(20)	Обязательное поле
Имя, отчество	S_NAME	V(50)	Обязательное поле
Дата рождения	BIRTH	D	Обязательное поле
Пол	GENDER	C(1)	Обязательное поле
Номер паспорта	N_PASSPORT	N(10)	Обязательное уникальное поле
Кем выдан паспорт	DEP_PASSPORT	V(100)	Обязательное поле
Когда выдан паспорт	D_PASSPORT	D	Обязательное поле
ИНН	INN	V(100)	Обязательно уникальное поле
Должность	POSITION	V(50)	Внешний ключ к Jobs
Рабочий телефон	PHONE	V(15)	Обязательное поле
Рабочий e-mail	MAIL	V(30)	Обязательное поле

Схема отношения Должности(Jobs)

Содержание поля	Имя поля	Тип, длина	Примечание
Должность	POSITION	V(50)	<b>Первичный ключ</b>
Зарплата	SALARY	N(6)	Обязательное поле



Схема отношения Образование(Education)

Содержание поля	Имя поля	Тип, длина	Примечание
Идентификатор	ID	N(3)	Внешний ключ к Employee
Вид образования	TYPE	V(15)	Обязательное поле
Номер диплома	NUM_DEGREE	N(12)	Обязательное поле
Специальность	SPECIALITY	V(50)	
Выдавшее диплом учреждение	ORG_DEGREE	V(50)	Обязательное поле

Схема отношения Клиенты(Clients)

Содержание поля	Имя поля	Тип, длина	Примечание
Идентификатор	ID	N(3)	<b>Суррогатный первичный ключ</b>
Фамилия	F_NAME	V(20)	Обязательное поле
Имя, отчество	S_NAME	V(50)	Обязательное поле
Дата рождения	BIRTH	D	Обязательное поле
Телефон	PHONE	V(30)	Обязательное поле
Дата покупки абонемент	DATE_SUBS	D	Обязательное поле
Срок действия абонемент	TIME_SUBS	I	Обязательное поле

Схема отношения «Направления подготовки» (Exercises)

Содержание поля	Имя поля	Тип, длина	Примечание
Идентификатор	ID	N(3)	<b>Суррогатный первичный ключ</b>
Название	NAME	V(30)	Обязательное поле

Схема отношения «Типы тренировок» (Types\_of\_Train)

Содержание поля	Имя поля	Тип, длина	Примечание
-----------------	----------	---------------	------------

Идентификатор	ID	N(3)	<b>Суррогатный первичный ключ</b>
Название	NAME	V(30)	Обязательно поле

Схема отношения Прейскурант(Prices)

Содержание поля	Имя поля	Тип, длина	Примечание
Идентификатор программы подготовки	ID	N(3)	<b>Суррогатный первичный ключ</b>
Направление подготовки	TRAIN	N(3)	Внешний ключ к Exercises
Тип тренировки	TYPE	N(3)	Внешний ключ к Types_of_Train
Место тренировки	SPACE_ID	N(2)	Внешний ключ к Space
Срок	TIME_SUBS	I	Обязательное поле
Стоимость	COST	N(5)	Обязательное поле

Схема отношения «Индивидуальные Тренировки» (Ind\_train)

Содержание поля	Имя поля	Тип, длина	Примечание
Тренер	EMP_ID	N(3)	Внешний ключ к Employee
Клиент	CLIENT_ID	N(3)	Внешний ключ к Clients
Программа тренировок	PROGRAM	N(3)	Внешний ключ к Prices
Дата тренировки	DATE	D	Обязательное поле
Время тренировки	TIME	T	Обязательное поле

Схема отношения «Групповые тренировки» (Group\_train)

Содержание поля	Имя поля	Тип, длина	Примечание
Идентификатор	ID	N(3)	<b>Суррогатный первичный ключ</b>
Тренер	EMP_ID	N(3)	Внешний ключ к Employee
Программа тренировок	PROGRAM	N(3)	Внешний ключ к Prices
Дата тренировки	DATE	D	Обязательное поле

Время тренировки	TIME	T	Обязательное поле
------------------	------	---	-------------------

Схема отношения «Участие в групповых тренировках» (Participation)

Содержание поля	Имя поля	Тип, длина	Примечание	
Идентификатор тренировки	GROUP_ID	N(3)	Внешний ключ к Group_train	<b>Составной первичный ключ</b>
Клиент	CLIENT_ID	N(3)	Внешний ключ к Clients	

Схема отношения Пространство (Space)

Содержание поля	Имя поля	Тип, длина	Примечание
Идентификатор	ID	N(2)	<b>Суррогатный первичный ключ</b>
Название	NAME	V(20)	Обязательное поле
Тип	TYPE	V(20)	Обязательное поле
Площадь	AREA	N(4)	Обязательное поле
Расходы	EXPENCES	N(6)	Обязательное поле

Схема отношения Оборудование (Equipment)

Содержание поля	Имя поля	Тип, длина	Примечание
Идентификатор	ID	N(5)	<b>Суррогатный первичный ключ</b>
Тип оборудования	TYPE	V(20)	Обязательное поле
Модель	MODEL	N(10)	Обязательное поле
Место нахождения	SPACE	N(2)	Внешний ключ к Space
Ответственный за инвентарь	RESP_ID	N(3)	Внешний ключ к Employee
Начало владения	START_OWN	D	Обязательное поле, не более 5 лет
Дата последнего осмотра (ремонта)	LAST_REPAIR	D	Обязательное поле, не более 6 месяцев

## Схема базы данных после нормализации



## Физическое проектирование

### Листинг таблиц:

```

create table Employee
(
    ID                NUMERIC(3) NOT NULL,
    F_NAME            VARCHAR(20) NOT NULL,
    S_NAME            VARCHAR(50) NOT NULL,
    BIRTH             DATE NOT NULL,
    GENDER            CHAR(1) NOT NULL,
    N_PASSPORT        NUMERIC(10) NOT NULL UNIQUE,
    DEP_PASSPORT      VARCHAR(100) NOT NULL,
    D_PASSPORT        DATE NOT NULL,
    INN               VARCHAR(100) NOT NULL UNIQUE,
    POSITION           VARCHAR(50) NOT NULL,
    PHONE             VARCHAR(15) NOT NULL,
    MAIL              VARCHAR(30) NOT NULL

    CONSTRAINT Check_pk
    CHECK(GENDER in ('Ж', 'М')),

    CONSTRAINT Employee_pk
    PRIMARY KEY (ID),
    FOREIGN KEY (POSITION) REFERENCES Jobs (POSITION)
);

CREATE TABLE Jobs
(

```

```

    POSITION    VARCHAR(50) NOT NULL,
    SALARY     NUMERIC(6) NOT NULL,

    CONSTRAINT Jobs_pk
    PRIMARY KEY (POSITION)
);

CREATE TABLE Education
(
    E_ID        NUMERIC(3) NOT NULL,
    TYPE        VARCHAR(15) NOT NULL,
    NUM_DEGREE  NUMERIC(12) NOT NULL,
    SPECIALITY   VARCHAR(50),
    ORG_DEGREE  VARCHAR(50) NOT NULL,

    CONSTRAINT Education_pk
    FOREIGN KEY (E_ID) REFERENCES Employee (ID)
);

CREATE TABLE Clients
(
    ID          NUMERIC(3) NOT NULL,
    F_NAME      VARCHAR(20) NOT NULL,
    S_NAME      VARCHAR(50) NOT NULL,
    BIRTH DATE  NOT NULL,
    PHONE       VARCHAR(30) NOT NULL,
    DATE_SUBS   DATE NOT NULL,
    TIME_SUBS   INT NOT NULL,

    CONSTRAINT CHECK_PK
    CHECK ( TIME_SUBS > 0 ),

    CONSTRAINT Clients_pk
    PRIMARY KEY (ID),
    FOREIGN KEY (ID) REFERENCES Employee (ID),
);

CREATE TABLE Exercises
(
    ID NUMERIC(3) NOT NULL,
    NAME VARCHAR(30) NOT NULL,

    PRIMARY KEY (ID)
);

CREATE TABLE Types_of_Train
(
    ID NUMERIC(3) NOT NULL,
    NAME VARCHAR(30) NOT NULL,

    CONSTRAINT Types_of_Train_pk
    PRIMARY KEY (ID)
);

CREATE TABLE Prices
(
    ID NUMERIC(3) NOT NULL,
    TRAIN NUMERIC(3) NOT NULL,
    TYPE NUMERIC(3) NOT NULL,

```

```

SPACE_ID NUMERIC(2) NOT NULL,
TIME_SUBS INT NOT NULL,
COST NUMERIC(5) NOT NULL,

CONSTRAINT Prices_FK
FOREIGN KEY (TYPE) REFERENCES Types_of_Train(ID),
FOREIGN KEY (TRAIN) REFERENCES Exercises(ID),
FOREIGN KEY (SPACE_ID) REFERENCES Space(ID),

CONSTRAINT Prices_PK
PRIMARY KEY (TRAIN, TYPE, TIME_SUBS)
);

CREATE TABLE Space
(
    ID numeric(2) NOT NULL,
    NAME VARCHAR(20) NOT NULL,
    TYPE VARCHAR(20) NOT NULL,
    AREA NUMERIC(4) NOT NULL,
    EXPERIENCES NUMERIC(6) NOT NULL,

    CONSTRAINT Space_pk
    PRIMARY KEY (ID)
);

CREATE TABLE Equipment
(
    ID NUMERIC(5) NOT NULL,
    TYPE VARCHAR(20) NOT NULL,
    MODEL NUMERIC(10) NOT NULL,
    SPACE NUMERIC(2) NOT NULL,
    START_OWN DATE NOT NULL,
    LAST_REPAIR DATE NOT NULL,
    RESP_ID NUMERIC(3) NOT NULL,

    CONSTRAINT Equipment_PK
    PRIMARY KEY (ID),

    CONSTRAINT Equipment_CHECK
    CHECK ( DATE_PART('month', NOW()) - DATE_PART('month', LAST_REPAIR) < 6),
    CHECK ( DATE_PART('year', NOW()) - DATE_PART('year', START_OWN) < 5),

    CONSTRAINT Equipment_FK
    FOREIGN KEY (SPACE) REFERENCES Space (ID),
    FOREIGN KEY (RESP_ID) REFERENCES Employee (ID)
);

CREATE TABLE Ind_train
(
    EMP_ID NUMERIC(3) NOT NULL,
    CLIENT_ID NUMERIC(3) NOT NULL,
    PROGRAM NUMERIC(3) NOT NULL,
    DATE DATE NOT NULL,
    TIME TIME NOT NULL,

    CONSTRAINT Equipment_FK
    FOREIGN KEY (PROGRAM) REFERENCES Prices (ID),
    FOREIGN KEY (EMP_ID) REFERENCES Employee (ID),
    FOREIGN KEY (CLIENT_ID) REFERENCES Clients (ID)
)

```

```

CREATE TABLE Group_train
(
    ID          NUMERIC(3) NOT NULL,
    EMP_ID      NUMERIC(3) NOT NULL,
    PROGRAM     NUMERIC(3) NOT NULL,
    DATE        DATE NOT NULL,
    TIME        TIME NOT NULL,

    CONSTRAINT Group_train_FK
    FOREIGN KEY (PROGRAM) REFERENCES Prices (ID),
    FOREIGN KEY (EMP_ID) REFERENCES Employee (ID),

    CONSTRAINT Group_train_PK
    PRIMARY KEY (ID)
)

```

## Создание готовых представлений:

```

//клиенты с абонеентами которые истекнут в течение трех месяцев или истекли//
CREATE VIEW THREE_MONTH_LEFT AS
    SELECT S_NAME, PHONE, DATE_SUBS, TIME_SUBS
    FROM Clients
    where (NOW()::date - DATE_SUBS::date)/30-time_subs+3 < 3;

//клиенты у которых сегодня день рождения//
CREATE VIEW BIRTHDAY_LIST as
    SELECT F_NAME, S_NAME, BIRTH, PHONE
    FROM Clients
    WHERE BIRTH = NOW();

//Работники без специализации//
CREATE VIEW WITHOUT_SPECIALISATION AS
    SELECT *
        from Education
        join Employee on Employee.ID = Education.E_ID
    WHERE SPECIALITY IS NULL;

//оборудование, которое нужно заменить в текущем месяце //
create or replace view OLD_EQUIPMENT as
select *
from Equipment
where current_date-START_OW<30

// тренеры с высшим образованием //
create or replace view HIGH_EDU as
select F_NAME, S_NAME
from Employee as e
join Education as ed on ed.id=e.id
where ed.TYPE='высшее' and e.POSITION='тренер'

//клиенты, абонементы которых истекают на ближайшей(текущей) неделе//
create or replace view END_CLIENT as
select F_NAME, S_NAME, PHONE
from Clients
where current_date-DATE_SUBS<7

```

## Определение прав доступа:

```
CREATE USER trainer WITH PASSWORD 'trainer';
CREATE USER manager WITH PASSWORD 'manager';
CREATE USER admin1 WITH PASSWORD 'admin';
CREATE USER director WITH PASSWORD 'director';

CREATE ROLE "adm" NOSUPERUSER INHERIT NOCREATEDB NOCREATEROLE NOREPLICATION;
GRANT "adm" TO admin1;
GRANT ALL PRIVILEGES ON ALL TABLES IN SCHEMA public TO GROUP "adm";

CREATE ROLE "trainers" NOSUPERUSER INHERIT NOCREATEDB NOCREATEROLE NOREPLICATION;
GRANT "trainers" TO trainer;
GRANT SELECT, INSERT, UPDATE, DELETE ON Train, Types_of_Train, Clients, Ind_train,
Group_train TO GROUP trainers;

CREATE ROLE "managers" NOSUPERUSER INHERIT NOCREATEDB NOCREATEROLE NOREPLICATION;
GRANT "managers" TO manager;
GRANT SELECT, INSERT, UPDATE, DELETE ON clients, accounting, space, train, prices,
types_of_train, Ind_train, Group_train to manager;
GRANT SELECT ON employee, education to manager;

CREATE ROLE "director" NOSUPERUSER INHERIT NOCREATEDB NOCREATEROLE NOREPLICATION;
GRANT "director" TO director;
GRANT SELECT ON ALL TABLES IN SCHEMA public TO GROUP "director";
```

Таблицы для визуализации доступа user'ов:

Представления	Администрат ор	Трене р	Менеджер	Директо р
Список клиентов с днем рождения сегодня (BIRTHDAY_LIST)	SUID		SUID	S
Специалисты без специализации(WITHOUT_SPECIALISATION)	SUID		S	S
Список старого оборудования (OLD_EQUIPMENT)	SUID			S
Клиенты с заканчивающимися абонеменами (THREE_MONTH_LEFT)	SUID		SUID	S
Тренеры с высшим образованием (HIGH_EDU)	SUID		S	S
Клиенты, абонементы которых истекают на ближайшей(текущей) неделе (END_CLIENT)	SUID	SUID	SUID	S



Таблицы	Администратор	Тренер	Менеджер	Директор
Employee	SUID		S	S
Clients	SUID	SUID	SUID	S
Education	SUID		S	S
Equipment	SUID			S
Space	SUID		SUID	S
Train	SUID	SUID	SUID	S
Prices	SUID		SUID	S
Types_of_train	SUID	SUID	SUID	S
Exirsises	SUID			S
Jobs	SUID			S
Participation	SUID			S
Ind_train	SUID	SUID	SUID	S
Group_train	SUID	SUID	SUID	S

### Создание индексов

Индексы, конечно, ускоряют доступ к данным, но также требуют затрат на поддержание. Конкретно в нашей БД объем данных не такой большой, поэтому было решено не применять индексирование.

### Резервное копирование

Резервное копирование данных БД будет происходить раз в сутки.