

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

**Московский институт электроники и математики им. А.Н. Тихонова**

Домашняя работа

«Проектирование реляционной базы данных»

на тему:

**«БД сервиса по аренде автомобилей»**

по дисциплине «Базы данных»

Выполнил:

студент группы БИВ184

Милосердов Александр Вячеславович

Москва 2020 г.

## Оглавление

1.	Инфологическое проектирование .....	3
1.1	Особенности предметной области .....	3
1.2	Сущности и атрибуты предметной области.....	3
1.3	Определение круга пользователей и информационных задач ..	4
2.	Логическое проектирование .....	6
2.2	Составление реляционных отношений.....	7
2.3	Нормализация полученных отношений (до 4НФ).....	10
2.4	Определение дополнительных ограничений целостности .....	15
2.5	Описание групп пользователей и прав доступа.....	15
3.	Реализация проекта базы данных .....	16
3.1	Создание таблиц.....	16
3.2	Создание представлений .....	19
3.3	Назначение прав доступа .....	20
3.4	Создание индексов.....	21
3.5	Разработка стратегии резервного копирования .....	21
4.	Выводы.....	21

# 1. Инфологическое проектирование

## 1.1 Особенности предметной области

База данных создается для информационного обслуживания сервиса по прокату автомобилей, обеспечивает учет транспортных средств, управление договорами аренды и менеджмент клиентов.

В соответствии с предметной областью система строится с учетом следующих особенностей:

- Арендодатель является владельцем автопарка.
- Все автомобили относятся к одному автопарку.
- Клиент берет в аренду автомобиль у арендатора на основании договора, обе стороны являются участниками договора.
- Один клиент может взять в аренду несколько автомобилей одновременно.
- Один автомобиль одновременно может участвовать только в одном договоре.
- Юридический и почтовый адрес арендодателя совпадают.
- Номерной знак автомобиля остается неизменным (права собственности на автомобиль принадлежат арендодателю на протяжении всего срока эксплуатации автомобиля).
- Рейтинг (от 0 до 5) определяет надежность клиента, размер залога, возможность брать машины более высокого класса и получать льготные условия на аренду.

## 1.2 Сущности и атрибуты предметной области

- Автомобиль. Атрибуты: **номерной знак**, марка, модель, класс, цвет, год выпуска, пробег, тип страхования, *номер свидетельства о регистрации ТС*.

- Арендодатель. Атрибуты: **название организации**, ОГРН (основной государственный регистрационный номер), ИИН, КПП (код причины постановки на учет), *почтовый и юридический адрес*.
- Арендатор. Атрибуты: *паспортные данные*, ФИО, **ИНН**, *место жительства*, дата рождения, пол, дата получения первого водительского удостоверения, дата окончания срока действия водительского удостоверения, участие в ДТП, рейтинг, телефоны.
- Автопарк. Атрибуты: **название**, *адрес*, объем парка (количество парковочных мест), телефоны.
- Договор об аренде. Атрибуты: **номер договора**, начало срока аренды, окончание срока аренды, размер арендной платы, дисконт, размер залога.

Предметом каждого договора является автомобиль.

Условные обозначения:

- Полу жирное начертание применено к **идентифицирующим атрибутам**
- Курсивное начертание применено к *составным атрибутам*
- Подчеркивание текста применено к многозначным атрибутам

Исходя из выявленных сущностей и описания предметной области построим ER-диаграмму:

### 1.3 Определение круга пользователей и информационных задач

Определим группы пользователей, их основные задачи и запросы к БД:

#### 1. Руководитель сервиса:

- Получение полной информации о клиентах и договорах
- Получение полных сведений обо всех имеющихся автомобилях
- Определение классов автомобилей
- Возможность изменения данных арендодателя
- Определение размера почасовой оплаты для конкретного класса

2. Менеджер по работе с клиентами:

- Получение и редактирование списка клиентов
- Заключение договоров
- Получение сведений обо всех договорах
- Получение информации об автомобилях, доступных для проката и автомобилях, взятых в прокат
- Определение доступных клиенту классов
- Определение размера дисконта и залога
- Просмотр и редактирование данных квитанций об оплате

3. Сотрудник автопарка:

- Получение полной информации об автомобилях
- Получение сведений об автопарке
- Редактирование данных об автомобилях

4. Клиент (арендатор):

- Просмотр информации об автомобилях
- Получение списка доступных к прокату автомобилей
- Получение списка заключенных (действующих и завершившихся) договоров на свое имя
- Просмотр информации о договоре на свое имя
- Просмотр квитанций об оплате

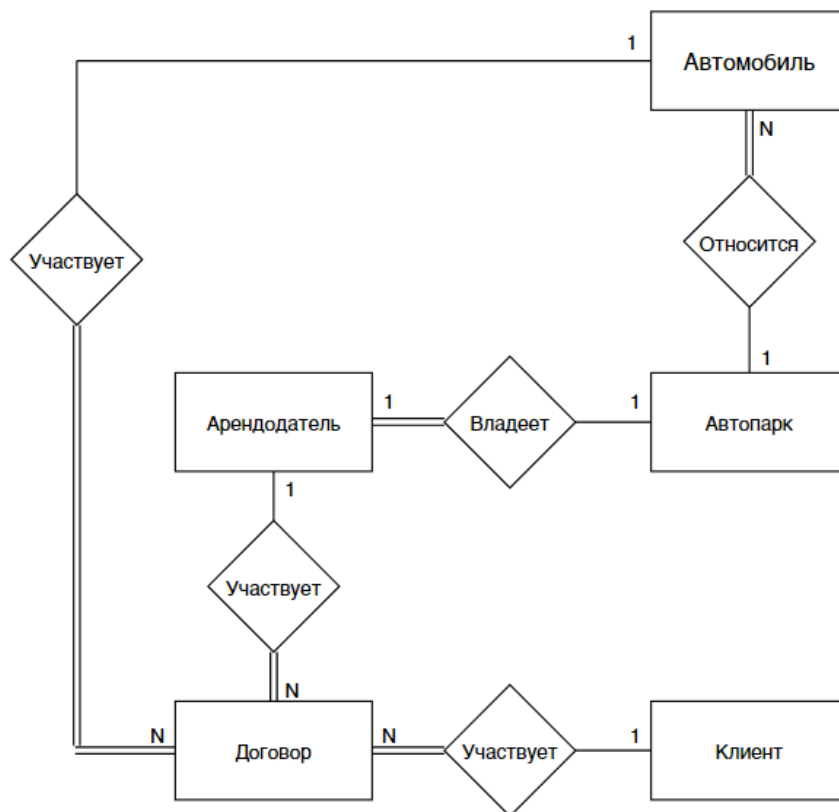


Рис. 1 1 ER-диаграмма предметной области

## 2. Логическое проектирование

### 2.1 Преобразование ER-диаграммы в схему базы данных



## 2.2 Составление реляционных отношений

Отношения приведены в табл. 2.2.1-2.2.5. Для каждого отношения указаны атрибуты с их внутренним названием, типом и длиной. Типы данных обозначаются так: N – числовой, C – символьный тип фиксированной длины, V – символьный тип переменной длины, D – дата (этот тип имеет стандартную длину, зависящую от СУБД, поэтому она не указывается).

Таблица 2.2.1 Отношение Автомобиль (car)

Содержание поля	Имя поля	Тип, длина	Примечания
Номерной знак	C_ID	C(9)	<b>первичный ключ</b>
Марка	C_FIRM	V(32)	обязательное поле
Модель	C_MODEL	V(32)	обязательное поле
Класс	C_CLASS	V(16)	обязательное поле, «эконом», «стандарт» или «премиум»
Цвет	C_COLOR	V(20)	обязательное поле
Год выпуска	C_YEAR	D	обязательное поле
Пробег	C_MILEAGE	N(12)	обязательное поле
Тип страхования	C_INSURANCE	C(5)	обязательное поле, «ОСАГО» или «КАСКО»
Номер свидетельства о регистрации ТС	C_TC_NUM	C(10)	обязательное поле
Автопарк	C_PARK	V(32)	вн. ключ (к autopark)

Таблица 2.2.2 Отношение Арендодатель (lessor)

Содержание поля	Имя поля	Тип, длина	Примечания
Название	L_NAME	V(64)	обязательное поле
ОГРН	L_ID	C(13)	<b>первичный ключ</b>
ИНН	L_INN	C(10)	обязательное поле
КПП	L_KPP	C(9)	обязательное поле
Адрес	L_ADDR	V(100)	обязательное поле

Таблица 2.2.3 Отношение Клиент (renter)

Содержание поля	Имя поля	Тип, длина	Примечания
Паспортные данные	R_PASSP	C(10)	обязательное поле
ФИО	R_FIO	V(100)	обязательное поле
Номер водительского удостоверения	R_DRLNS	C(10)	обязательное поле
СНИЛС	R_SNILS	C(11)	<b>первичный ключ</b>
Место жительства	R_ADDR	V(100)	обязательное поле
Дата рождения	R_BORN	D	обязательное поле
Пол	R_GENDER	C(1)	обязательное поле, «М» или «Ж»
Дата получения первого водительского удостоверения	R_1ST_DL	D	обязательное поле
Дата окончания срока действия водительского удостоверения	R_END_DL	D	обязательное поле, >даты получения
Участие в ДТП	R_DTP	V(12)	обязательное поле
Рейтинг	R_RATE	N(2,1)	обязательное поле, от 1 до 5
Телефоны	R_PHONE	V(30)	обязательное многозначное поле



Таблица 2.2.4 Отношение Автопарк (autopark)

Содержание поля	Имя поля	Тип, длина	Примечания
Название	P_NAME	V(30)	<b>первичный ключ</b>
Адрес	P_ADDR	V(100)	обязательное поле
Объем	P_VOL	V(9)	обязательное поле
Телефоны	P_PHONE	V(30)	обязательное многозначное поле
Владелец	P_OWNER	C(13)	вн. ключ (к lessor)

Таблица 2.2.5. Отношение Договор (contract)

Содержание поля	Имя поля	Тип, длина	Примечания
Номер	CTR_ID	C(9)	<b>суррогатный первичный ключ</b>
Начало срока аренды	CTR_BEGIN	D	обязательное поле
Завершение срока аренды	CTR_END	D	обязательное поле, больше начала срока
Размер арендной платы	CTR_SIZE	N(12)	обязательное поле
Размер залога	CTR_DEP	N(12)	обязательное поле
Дисконт	CTR_BEN	N(2)	необязательное поле от 0 до 99
Арендодатель	CTR_LES	C(13)	вн. ключ (к lessor)
Клиент	CTR_RNTR	C(10)	вн. ключ (к renter)
Предмет договора	CTR_CAR	C(9)	вн. ключ (к car)

## 2.3 Нормализация полученных отношений (до 4НФ)

**1НФ.** Для приведения таблиц к 1НФ требуется составить прямоугольные таблицы (одно значение атрибута – одна ячейка таблицы) и разбить сложные (составные) атрибуты на простые.

Атрибут ФИО разделим на два атрибута: Фамилия и Имя, отчество, Паспортные данные на Серия и номер паспорта, Дата выдачи, Кем выдан, а Дисконт. Многозначный атрибут Телефоны в отношениях *Клиент* и *Автопарк* заменим на однозначный атрибут Контактный номер телефона, так как для связи с клиентом или автопарком достаточно знать один рабочий номер телефона.

**2НФ.** Все отношения разрабатываемой базы данных находятся в 1НФ и имеют первичные ключи, неключевые атрибуты этих отношений функционально полно зависят от составных первичных ключей.

**3НФ.** В отношении *Автомобиль* атрибут Класс зависит от Марки и Модели, а модель в свою очередь, зависит от Марки. Создадим отдельное отношение *Классы* с первичным ключом Название класса и полем Плата за час. Создадим справочную таблицу *Марки* с полем Класс – внешним ключом к отношению *Классы*. Информацию о модели автомобиля так же выделим в отдельное отношение *Модели*.

Атрибут Рейтинг в отношении *Клиент* не выносится в отдельную таблицу, поскольку не предполагается хранение данных о дате ДТП, размере ущерба и т.п.

Атрибут Размер арендной платы в отношении *Договор* является вычислимым полем, при этом зависимость 1:1, поэтому создавать отдельное отношение не требуется.

**4НФ.** Получившиеся отношения соответствуют 4-й нормальной форме, в таблицах отсутствуют нетривиальные многозначные зависимости.

Нормализованные отношения приведены в таблицах 2.3.1 – 2.3.7

Таблица 2.3.1 Отношение Классы (Classes)

Содержание поля	Имя поля	Тип, длина	Примечания
Название класса	CLASS	V(16)	<b>первичный ключ</b>
Плата за час	PLATA	N(6)	обязательное поле

Таблица 2.3.2 Отношение Модели (Models)

Содержание поля	Имя поля	Тип, длина	Примечания
ID Модели	M_ID	V(6)	<b>суррогатный первичный ключ</b>
Название модели	M_MODEL	V(32)	обязательное поле
Класс	M_CLASS	V(16)	вн. ключ к отношению (classes)

Таблица 2.3.3 Отношение Марки (Firms)

Содержание поля	Имя поля	Тип, длина	Примечания
ID Марки	F_ID	V(6)	<b>суррогатный первичный ключ</b>
Название марки	F_NAME	V(32)	обязательное поле

Таблица 2.3.4 Отношение Автопарк (autopark)

Содержание поля	Имя поля	Тип, длина	Примечания
Название	P_NAME	V(30)	<b>первичный ключ</b>
Адрес	P_ADDR	V(100)	обязательное поле
Объем	P_VOL	N(6)	обязательное поле
Контактный номер телефона	P_PHONE	V(30)	обязательное поле
Владелец	P_OWNER	C(13)	вн. ключ (к lessor)

Таблица 2.3.5 Отношение Автомобиль (car)

Содержание поля	Имя поля	Тип, длина	Примечания
Номерной знак	C_ID	C(9)	<b>первичный ключ</b>
Модель	C_MOD	V(6)	вн. ключ (к models)
Марка	C_FIRM	V(6)	вн. ключ (к firms)
Цвет	C_COLOR	V(20)	обязательное поле
Год выпуска	C_YEAR	N(4)	обязательное поле
Пробег	C_MILEAGE	N(12)	обязательное поле
Тип страхования	C_INSURANCE	C(5)	обязательное поле, «ОСАГО» или «КАСКО»
Номер свидетельства о регистрации ТС	C_TC_NUM	C(10)	обязательное поле
Автопарк	C_PARK	V(30)	вн. ключ (к autopark)

Таблица 2.3.6 Отношение Арендодатель (lessor)

Содержание поля	Имя поля	Тип, длина	Примечания
ОГРН	L_ID	C(13)	<b>первичный ключ</b>
Название	L_NAME	V(10)	обязательное поле
ИНН	L_INN	C(10)	обязательное поле
КПП	L_KPP	C(9)	обязательное поле
Адрес	L_ADDR	V(100)	обязательное поле

Таблица 2.3.7 Схема отношения Клиент (renter)

Содержание поля	Имя поля	Тип, длина	Примечания
Серия и номер паспорта	R_PASP	C(10)	обязательное поле
Когда и кем выдан	R_KEM_KOGDA	V(100)	обязательное поле
Фамилия	R_FNAME	V(64)	обязательное поле
Имя, отчество	R_LNAME	V(100)	обязательное поле
Номер водительского удостоверения	R_DRLNS	C(10)	обязательное поле
СНИЛС	R_SNILS	C(11)	<b>первичный ключ</b>
Место жительства	R_ADDR	V(100)	обязательное поле
Дата рождения	R_BORN	D	обязательное поле
Пол	R_GENDER	C(1)	обязательное поле, «М» или «Ж»
Дата получения первого водительского удостоверения	R_1ST_DL	D	обязательное поле
Дата окончания срока действия водительского удостоверения	R_END_DL	D	обязательное поле
Контактный номер телефона	R_PHONE	V(30)	обязательное поле
Рейтинг клиента	R_RATE	N(2,1)	обязательное поле

Таблица 2.3.8 Отношение Договор (contract)

Содержание поля	Имя поля	Тип, длина	Примечания
Номер	CTR_ID	C(9)	<b>суррогатный первичный ключ</b>
Начало срока аренды	CTR_BEGIN	D	обязательное поле
Завершение срока аренды	CTR_END	D	обязательное поле, больше начала срока
Размер арендной платы	CTR_SIZE	N(12)	обязательное поле
Размер залога	CTR_DEP	N(12)	обязательное поле
Дисконт	CTR_BEN	N(2)	необязательное поле от 0 до 99
Арендодатель	CTR_LES	C(13)	вн. ключ (к lessor)
Клиент	CTR_RNTR	C(10)	вн. ключ (к renter)
Предмет договора	CTR_CAR	C(9)	вн. ключ (к car)

Схема базы данных после нормализации:



## **2.4 Определение дополнительных ограничений целостности**

1. Атрибут «Плата за час» неотрицателен
2. Атрибут «Год выпуска» не меньше 1980
3. Поле «Тип страхования» принимает значения «ОСАГО» или «КАСКО»
4. «Объем» автопарка неотрицателен
5. Атрибут «Дата получения первого водительского удостоверения» больше атрибута «Дата окончания срока действия водительского удостоверения»
6. Поле «Рейтинг клиента» принимает значения от 0 до 5
7. «Размер арендной платы» неотрицателен
8. Атрибут «Начало срока аренды» больше атрибута «Завершение срока аренды»
9. «Размер залога» неотрицателен
10. «Дисконт» принимает значения от 0 до 99

## **2.5 Описание групп пользователей и прав доступа**

Опишем для каждой группы пользователей права доступа к каждой таблице. Права приведены в табл. 2.5.1, используются следующие сокращения:

S – чтение данных (select);

I – добавление данных (insert);

U – модификация данных (update);

D – удаление данных (delete).

Таблица 2.5.1 Права доступа к таблицам для групп пользователей

Таблицы	Группы пользователей (роли)			
	Руководитель сервиса	Менеджер по работе с клиентами	Сотрудник автопарка	Клиент (арендатор)
Арендодатель	SUID	S	S	S
Автопарк	SUID	S	SU	S
Модели	S	S	SUID	S
Марки	S	S	SUID	S
Классы	SUID	S	S	S
Автомобиль	S	S	SUID	S
Клиент	S	SUID		S*
Договор	S	SUID		S*

\* возможен просмотр только своих данных и договоров

### 3. Реализация проекта базы данных

Описание логической схемы БД выполнено на SQL-92, далее приведена реализация схемы на Data Definition Language.

#### 3.1 Создание таблиц

##### 1) Отношение Lessor (Арендодатель)

```
CREATE TABLE lessor
(
    l_id    char(13) PRIMARY KEY,
    l_name  varchar(10) not null,
    l_inn   char(10)    not null,
    l_kpp   char(9)     not null,
    l_addr  varchar(100) not null
);
```



## 2) Отношение Autopark (Автопарк)

```
CREATE TABLE autopark
(
    p_name varchar(30) PRIMARY KEY,
    p_addr varchar(100) not null,
    p_vol numeric(6) not null check (p_vol >= 0),
    p_phone varchar(30) not null,
    p_owner char(13) not null
    constraint ref_lessor references lessor
);
```

## 3) Отношение Classes (Классы)

```
CREATE TABLE classes
(
    class varchar(16) PRIMARY KEY,
    plata numeric(6) not null check (plata >= 0)
);
```

## 4) Отношение Models (Модели)

```
CREATE TABLE models
(
    m_id varchar(6) primary key,
    m_name varchar(32) not null,
    m_class varchar(16)
    constraint ref_classes references classes
);
```

## 5) Отношение Firms (Марки)

```
CREATE TABLE firms
(
    f_id varchar(6) primary key ,
    f_name varchar(32) not null
);
```

## 6) Отношение Car (Автомобиль)

```
CREATE TABLE car
(
    c_id char(9) primary key,
    c_firm varchar(6)
    constraint ref_firms references firms,
    c_mod varchar(6)
    constraint ref_models references models,
    c_color varchar(20),
    c_year numeric(4) not null check (c_year >= 1980),
    c_mileage numeric(12) not null,
    c_insurance char(5) not null check (c_insurance in ('ОСАГО',
```

```

'КАСКО'))),
    c_tc_num    char(10)    not null,
    c_park      varchar(30)
        constraint ref_park references autopark
);

```

## 7) Отношение Renter (Клиент)

```

CREATE TABLE renter
(
    r_snils char(10) primary key,
    r_pasp   char(10)    not null,
    r_kem_kogda varchar(100) not null,
    r_fname  varchar(64)  not null,
    r_lname  varchar(100) not null,
    r_drlns  char(10) not null ,
    r_addr   varchar(100) not null,
    r_born   date         not null,
    r_gender char(1) check ( r_gender in ('M', 'Ж')),
    r_1st_dl date         not null,
    r_end_dl date         not null check (r_end_dl - r_1st_dl >
0),
    r_phone  varchar(30)  not null,
    r_rate   numeric(2, 1) not null check (r_rate >= 0 and 5 >=
r_rate)
);

```

## 8) Отношение Contract (Договор)

```

CREATE TABLE contract
(
    ctr_id    numeric(9) primary key,
    ctr_begin date         not null,
    ctr_end   date         not null check (ctr_end - ctr_begin > 0),
    ctr_size  numeric(12) not null check (ctr_size >= 0),
    ctr_dep   numeric(12) not null check (ctr_dep >= 0),
    ctr_sale  numeric(2) check (ctr_sale >= 0 and 99 >= ctr_sale),
    ctr_les   char(13)
        constraint ref_lessor references lessor,
    ctr_rntr  char(10)
        constraint ref_renter references renter,
    ctr_car   char(9)
        constraint ref_car references car
);

```

## 3.2 Создание представлений

Приведем примеры нескольких готовых запросов (представлений)

1. Список всех действующих договоров

```
create view crr_contracts as
select *
from contract
where current_date <= ctr_end
order by ctr_end;
```

2. 15 клиентов с лучшим рейтингом

```
create or replace view renter_top (r_fname, r_lname, r_rate) as
select r_fname, r_lname, r_rate
from renter
order by r_rate desc
limit 15;
```

(Для вручения премиальных карт руководителем)

3. Список автомобилей, которые были взяты в прокат клиентами с рейтингом ниже 3.5

```
create view low_rate (c_id, c_firm, c_mod) as
select c_id,
       c_firm,
       c_mod
from car,
     renter,
     contract
where c_id = ctr_car
     and r_snils = ctr_rntr
     and r_rate <= 3.5
group by c_id, c_firm, c_mod;
```

4. Список контактных номеров клиентов, которые вернули автомобиль из проката не более 7 дней назад

```
create view renter_phones (r_fname, r_lname, r_phone) as
select r_fname, r_lname, r_phone
from renter,
     contract
where r_snils = ctr_rntr
     and current_date - ctr_end <= 7
group by (r_fname, r_lname, r_phone);
```

## 5. Кол-во завершённых договоров у клиента

```
create view num_of_ctr_for_mnf (First_name, Last_Name,
Number_of_Contracts) as
select r_fname, r_lname, r_phone, count(ctr_id)
from renter
      join contract on r_snils = ctr_rntr
where ctr_end < current_date
      group by r_fname, r_lname, r_phone;
```

Представление	Группы пользователей (роли)			
	Руководитель сервиса	Менеджер по работе с клиентами	Сотрудник автопарка	Клиент (арендатор)
crr_contracts	S	SUID		S*
renter_top	SUID	SU	S	S
low_rate	S	SUID	SU	
renter_phones	S	SUID	S	
num_of_ctr_for_mnf	S	SUID		S*

\* ВОЗМОЖЕН ПРОСМОТР ТОЛЬКО СВОИХ ДОГОВОРОВ

## 3.3 Назначение прав доступа

```
grant select, update, insert, delete on lessor, autopark, classes to head;
grant select on models, firms, car, renter, contract to head;
```

```
grant select on lessor, autopark, models, firms, classes, car to
customer_service_manager;
grant select, update, insert, delete on renter, contract to
customer_service_manager;
```

```
grant select, update, insert, delete on models, firms, car to fleet_employee;
grant select on lessor, classes to fleet_employee;
grant select, update on autopark to fleet_employee;
```

```
grant select on lessor, autopark, models, firms, classes, car, renter,
contract to renter;
```

```
grant select on crr_contracts, low_rate, renter_phones, num_of_ctr_for_mnf to
head;
grant select, update, insert, delete on renter_top to head;
```

```
grant select, update, insert, delete on crr_contracts, low_rate,
renter_phones, num_of_ctr_for_mnf to customer_service_manager;
```

```
grant select, update on renter_top to customer_service_manager;  
  
grant select on renter_top, renter_phones to fleet_employee;  
grant select, update on low_rate to fleet_employee;  
  
grant select on crr_contracts, num_of_ctr_for_mnf, renter_top to renter;
```

### 3.4 Создание индексов

Создадим индексы внешних ключей для повышения эффективности работы с данными:

```
create index ind_lessor on lessor(l_id, l_name);  
create index ind_autopark on autopark(p_name, p_addr);  
create index ind_model on models (m_id, m_name);  
create index ind_firm on firms (f_id, f_name);  
create index ind_renter on renter (r_snils, r_fname, r_lname, r_rate);  
create index ind_contract on contract (ctr_id, ctr_begin, ctr_end, ctr_size);
```

### 3.5 Разработка стратегии резервного копирования

Предполагается высокая интенсивность обновления базы данных в следствие большего числа клиентов, автомобилей и множества операций с договорами каждый день, так же высока важность самих данных, поэтому резервное копирование следует проводить не реже, чем раз в 1 час. Сервис по прокату автомобилей работает с 9:00 до 22:00 вечера, заключение договоров и работа с данными происходит в рабочие часы, в это время необходима непрерывная работа сервиса, а с 22:30 по 8:30 сервис может быть выключен.

## 4. Выводы

Таким образом, были выполнены этапы инфологического, логического, физического проектирования и реализации базы данных по прокату автомобилей.

Закреплены теоретические знания о проектировании реляционных баз данных, получены практические навыки реализации автоматизированных информационных систем, основанных на базах данных.