Министерство образования и науки Российской Федерации

Федеральное государственное автономное образовательное учреждение

Высшего профессионального образования

«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт сервиса туризма и дизайна (филиал) в г. Пятигорске

Кафедра информационной безопасности, систем и технологий

**Контрольная работа**

По дисциплине: «Технологии обработки информации»

Курс: 4 Группа: П-ИСТ-б-з-121

Направление: 09.03.02 «Информационные системы и технологии»

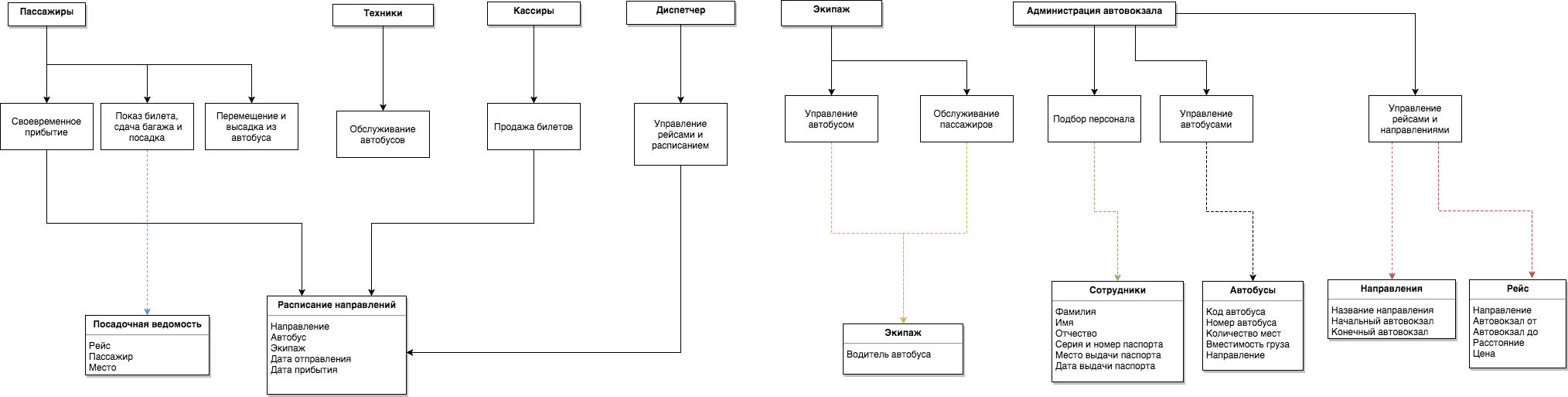
Выполнил: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Вариант № \_\_\_\_

Проверил: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Пятигорск, 2016 г

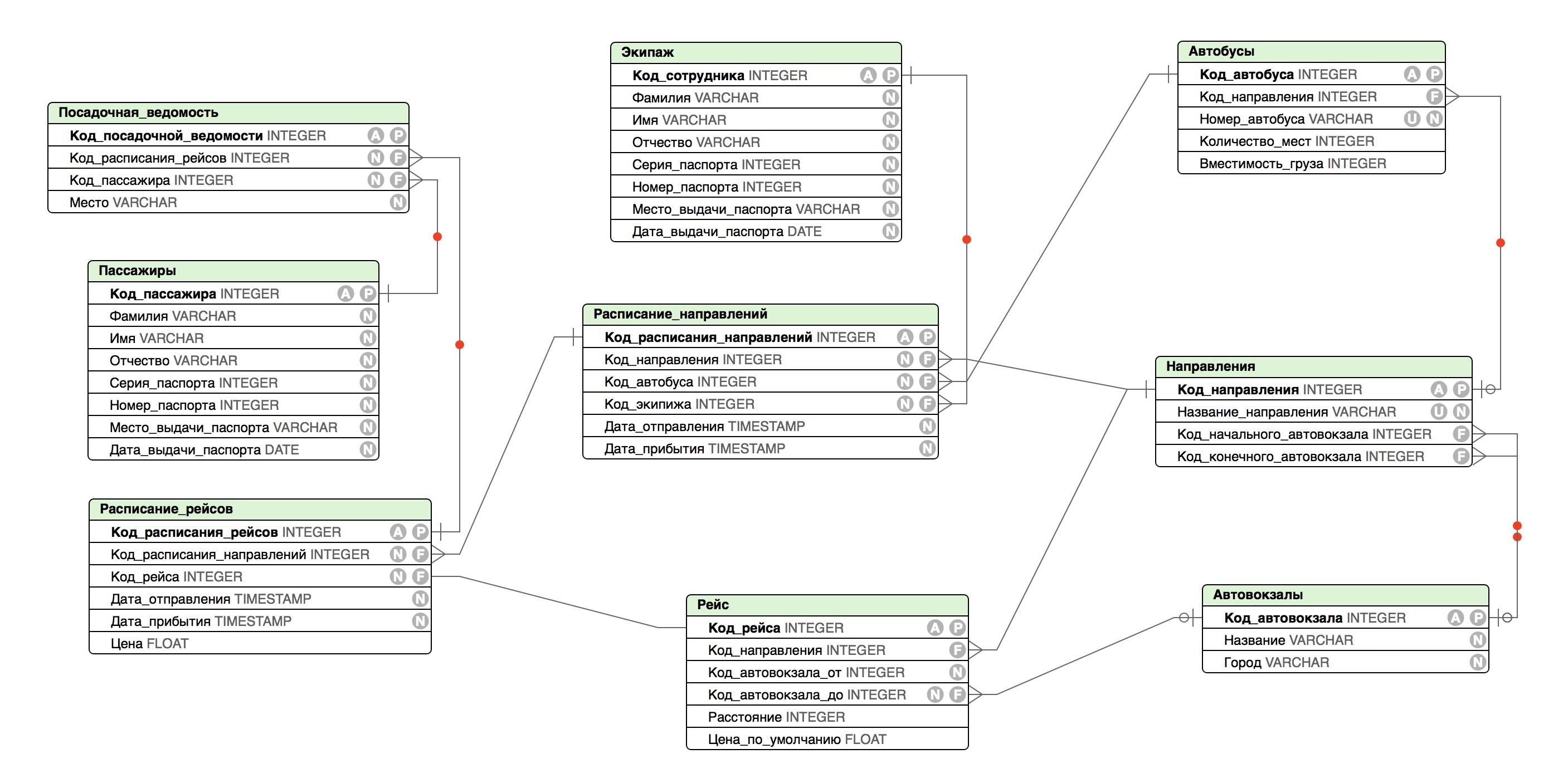
## Построение концептуальной модели данных

На следующей диаграмме изображены потенциальные пользователи системы, их задачи и необходимые данные. 

## 2. Построение логической модели данных

На следующей диаграмме наглядно показаны таблицы и их связи. Необходимо отметить следующее:

* + Система должна позволить пассажирам перемещаться по неполным маршрутам, поэтому решено разбить маршруты на рейсы
  + Рейс - маршрут соединяющий 2 города
  + Направление - основные маршруты компании, которые состоят из одного и более рейсов
  + Так как нам пришлось разбить маршруты на направления и рейсы, в расписание так же пришлось разбить на “Расписание направлений” и “Расписание рейсов”.
  + Стоимость билета для клиента рассчитывается автоматически из стоимости каждого рейса, который необходимо преодолеть. Цена рейса берется из расписания рейсов, но если она не указана, то берется цена по умолчанию рейса
  + Экипаж и автобус назначаются в расписании направления, так как эти данные могут варьироваться, особенно если несколько автобусов работают на одном направлении
  + В таблице “Посадочная ведомость” указываются все клиенты, которые приобрели билеты на данный рейс. Если клиент совершит возврат билета, то строка из данной таблицы будет удалена.
  + Так как ведется запись паспортных данных клиентов, есть возможность уникально идентифицировать пассажиров. Это может пригодиться для ведения определенной статистики.



## 3. Создание базы данных

CREATE TABLE "Автовокзалы"

(

"Код\_автовокзала" SERIAL,

"Название" VARCHAR NOT NULL,

"Город" VARCHAR NOT NULL,

CONSTRAINT "Автовокзалы\_pkey" PRIMARY KEY ("Код\_автовокзала")

);

CREATE TABLE "Направления"

(

"Код\_направления" SERIAL,

"Название\_направления" VARCHAR NOT NULL UNIQUE ,

"Код\_начального\_автовокзала" INTEGER,

"Код\_конечного\_автовокзала" INTEGER,

CONSTRAINT "Направления\_pkey" PRIMARY KEY ("Код\_направления")

);

CREATE TABLE "Пассажиры"

(

"Код\_пассажира" SERIAL,

"Фамилия" VARCHAR NOT NULL,

"Имя" VARCHAR NOT NULL,

"Отчество" VARCHAR NOT NULL,

"Серия\_паспорта" INTEGER NOT NULL,

"Номер\_паспорта" INTEGER NOT NULL,

"Место\_выдачи\_паспорта" VARCHAR NOT NULL,

"Дата\_выдачи\_паспорта" DATE NOT NULL,

CONSTRAINT "Пассажиры\_pkey" PRIMARY KEY ("Код\_пассажира")

);

CREATE TABLE "Автобусы"

(

"Код\_автобуса" SERIAL,

"Код\_направления" INTEGER,

"Номер\_автобуса" VARCHAR NOT NULL UNIQUE ,

"Количество\_мест" INTEGER,

"Вместимость\_груза" INTEGER,

CONSTRAINT "Автобусы\_pkey" PRIMARY KEY ("Код\_автобуса")

);

CREATE TABLE "Посадочная\_ведомость"

(

"Код\_посадочной\_ведомости" SERIAL,

"Код\_расписания\_рейсов" INTEGER NOT NULL,

"Код\_пассажира" INTEGER NOT NULL,

"Место" VARCHAR NOT NULL,

CONSTRAINT "Посадочная\_ведомость\_pkey" PRIMARY KEY ("Код\_посадочной\_ведомости")

);

CREATE TABLE "Рейс"

(

"Код\_рейса" SERIAL,

"Код\_направления" INTEGER,

"Код\_автовокзала\_от" INTEGER NOT NULL,

"Код\_автовокзала\_до" INTEGER NOT NULL,

"Расстояние" INTEGER,

"Цена\_по\_умолчанию" FLOAT,

CONSTRAINT "Рейс\_pkey" PRIMARY KEY ("Код\_рейса")

);

CREATE TABLE "Экипаж"

(

"Код\_сотрудника" SERIAL,

"Фамилия" VARCHAR NOT NULL,

"Имя" VARCHAR NOT NULL,

"Отчество" VARCHAR NOT NULL,

"Серия\_паспорта" INTEGER NOT NULL,

"Номер\_паспорта" INTEGER NOT NULL,

"Место\_выдачи\_паспорта" VARCHAR NOT NULL,

"Дата\_выдачи\_паспорта" DATE NOT NULL,

CONSTRAINT "Экипаж\_pkey" PRIMARY KEY ("Код\_сотрудника")

);

CREATE TABLE "Расписание\_рейсов"

(

"Код\_расписания\_рейсов" SERIAL,

"Код\_расписания\_направлений" INTEGER NOT NULL,

"Код\_рейса" INTEGER NOT NULL,

"Дата\_отправления" TIMESTAMP NOT NULL,

"Дата\_прибытия" TIMESTAMP NOT NULL,

"Цена" FLOAT,

CONSTRAINT "Расписание\_рейсов\_pkey" PRIMARY KEY ("Код\_расписания\_рейсов")

);

CREATE TABLE "Расписание\_направлений"

(

"Код\_расписания\_направлений" SERIAL,

"Код\_направления" INTEGER NOT NULL,

"Код\_автобуса" INTEGER NOT NULL,

"Код\_экипижа" INTEGER NOT NULL,

"Дата\_отправления" TIMESTAMP NOT NULL,

"Дата\_прибытия" TIMESTAMP NOT NULL,

CONSTRAINT "Расписание\_направлений\_pkey" PRIMARY KEY ("Код\_расписания\_направлений")

);

ALTER TABLE "Направления" ADD FOREIGN KEY ("Код\_начального\_автовокзала") REFERENCES "Автовокзалы" ("Код\_автовокзала");

ALTER TABLE "Направления" ADD FOREIGN KEY ("Код\_конечного\_автовокзала") REFERENCES "Автовокзалы" ("Код\_автовокзала");

ALTER TABLE "Автобусы" ADD FOREIGN KEY ("Код\_направления") REFERENCES "Направления" ("Код\_направления");

ALTER TABLE "Посадочная\_ведомость" ADD FOREIGN KEY ("Код\_посадочной\_ведомости") REFERENCES "Расписание\_рейсов" ("Код\_расписания\_рейсов");

ALTER TABLE "Посадочная\_ведомость" ADD FOREIGN KEY ("Код\_пассажира") REFERENCES "Пассажиры" ("Код\_пассажира");

ALTER TABLE "Рейс" ADD FOREIGN KEY ("Код\_направления") REFERENCES "Направления" ("Код\_направления");

ALTER TABLE "Рейс" ADD FOREIGN KEY ("Код\_автовокзала\_до") REFERENCES "Автовокзалы" ("Код\_автовокзала");

ALTER TABLE "Расписание\_рейсов" ADD FOREIGN KEY ("Код\_расписания\_направлений") REFERENCES "Расписание\_направлений" ("Код\_расписания\_направлений");

ALTER TABLE "Расписание\_рейсов" ADD FOREIGN KEY ("Код\_рейса") REFERENCES "Рейс" ("Код\_рейса");

ALTER TABLE "Расписание\_направлений" ADD FOREIGN KEY ("Код\_направления") REFERENCES "Направления" ("Код\_направления");

ALTER TABLE "Расписание\_направлений" ADD FOREIGN KEY ("Код\_автобуса") REFERENCES "Автобусы" ("Код\_автобуса");

ALTER TABLE "Расписание\_направлений" ADD FOREIGN KEY ("Код\_экипижа") REFERENCES "Экипаж" ("Код\_сотрудника");

INSERT INTO "public"."Автовокзалы"("Код\_автовокзала","Название","Город")

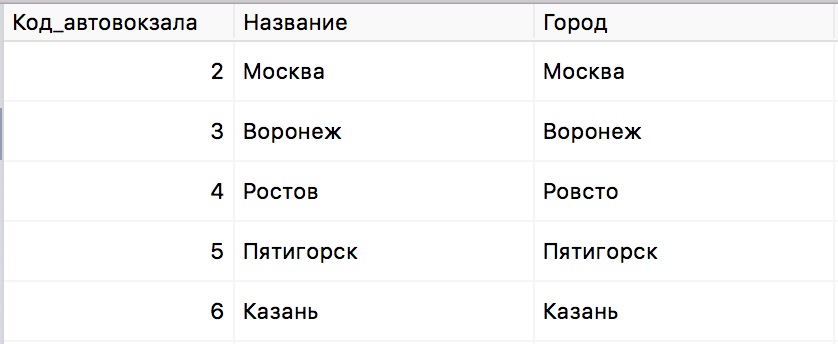
VALUES

(2,E'Москва',E'Москва'),

(3,E'Воронеж',E'Воронеж'),

(4,E'Ростов',E'Ровсто'),

(5,E'Пятигорск',E'Пятигорск'),

(6,E’Казань',E'Казань');

INSERT INTO "public"."Направления"("Код\_направления","Название\_направления","Код\_начального\_автовокзала","Код\_конечного\_автовокзала")

VALUES

(1,E'Москва-Ростов',2,4),

(2,E'Москва-Казань',2,6);

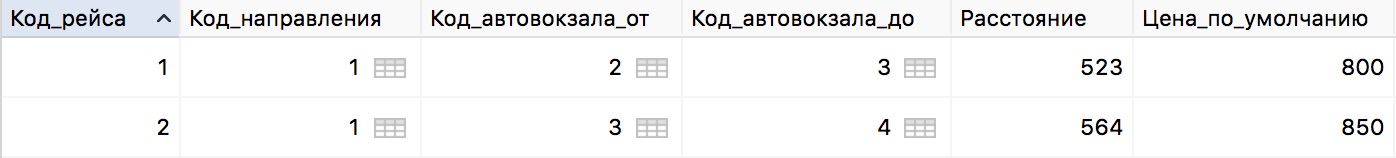
INSERT INTO "public"."Рейс"("Код\_рейса","Код\_направления","Код\_автовокзала\_от","Код\_автовокзала\_до","Расстояние","Цена\_по\_умолчанию")

VALUES

(1,1,2,3,523,800),

(2,1,3,4,564,850),

(3,2,2,6,1200,1400);



INSERT INTO "public"."Экипаж"("Код\_сотрудника","Фамилия","Имя","Отчество","Серия\_паспорта","Номер\_паспорта","Место\_выдачи\_паспорта","Дата\_выдачи\_паспорта")

VALUES

(1,E'Петров',E'Игорь',E'Сергеевич',1234,767574,E'УФМС101',E'2006-11-11'),

(2,E'Титов',E'Владимир',E'Иванович',1232,646432,E'УФМС202',E'2012-12-12');

INSERT INTO "public"."Автобусы"("Код\_автобуса","Код\_направления","Номер\_автобуса","Количество\_мест","Вместимость\_груза")

VALUES

(2,1,E'A360AA',25,2500),

(3,2,E'К201ББ',32,3000);

INSERT INTO "public"."Пассажиры"("Код\_пассажира","Фамилия","Имя","Отчество","Серия\_паспорта","Номер\_паспорта","Место\_выдачи\_паспорта","Дата\_выдачи\_паспорта")

VALUES

(1,E'Иванов',E'Петр',E'Ильич',1432,873452,E'УФМС202',E'2011-12-06'),

(2,E'Озмитель',E'Андрей',E'Васильевич',4311,341312,E'УФМС303',E'2011-12-06');

INSERT INTO "public"."Расписание\_направлений"("Код\_расписания\_направлений","Код\_направления","Код\_автобуса","Код\_экипижа","Дата\_отправления","Дата\_прибытия")

VALUES

(1,1,2,1,E'2017-05-04 10:00:00',E'2017-05-05 10:00:00'),

(2,2,3,2,E'2017-05-04 11:00:00',E'2017-05-05 12:00:00');

INSERT INTO "public"."Расписание\_рейсов"("Код\_расписания\_рейсов","Код\_расписания\_направлений","Код\_рейса","Дата\_отправления","Дата\_прибытия","Цена")

VALUES

(1,1,1,E'2017-05-04 10:00:00',E'2017-05-04 18:00:00',900),

(2,1,2,E'2017-05-04 18:05:00',E'2017-05-05 10:00:00',1000),

(3,2,3,E'2017-05-04 11:00:00',E'2017-05-05 12:00:00',2200);

INSERT INTO "public"."Посадочная\_ведомость"("Код\_посадочной\_ведомости","Код\_расписания\_рейсов","Код\_пассажира","Место")

VALUES

(1,1,1,E'22A'),

(2,2,1,E'22A'),

(3,3,2,E'11');

## 3.1 Запросы

1. Следующий запрос выводит номера автобусов, которые прибудут в Ростов 5 мая 2017 года:

SELECT Город, Номер\_автобуса, рр.Дата\_прибытия

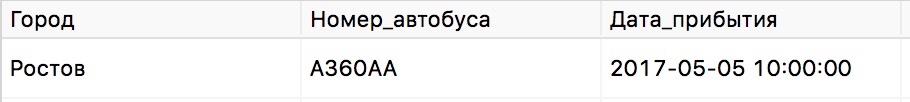
FROM Расписание\_рейсов рр

INNER JOIN Рейс р ON р.Код\_рейса = рр.Код\_рейса

INNER JOIN Автовокзалы а ON р.Код\_автовокзала\_до = а.Код\_автовокзала

INNER JOIN Расписание\_направлений рн ON рн.Код\_расписания\_направлений = рр.Код\_расписания\_направлений

INNER JOIN Автобусы аа ON рн.Код\_автобуса = аа.Код\_автобуса

WHERE а.Город = 'Ростов' and DATE(рр.Дата\_прибытия) = '2017-05-05'

2. Стоимость билетов на рейсы прибывающие в место назначения 5 мая 2017 года, сгруппированная по направлениям и отсортированная по убыванию:

SELECT нн.Название\_направления, sum(рр.Цена)

FROM Расписание\_направлений н

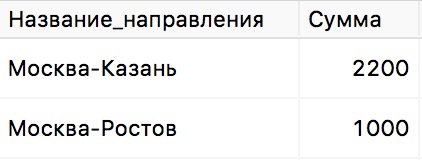
INNER JOIN Направления нн ON нн.Код\_направления = н.Код\_направления

INNER JOIN Расписание\_рейсов рр ON н.Код\_расписания\_направлений = рр.Код\_расписания\_направлений

INNER JOIN Посадочная\_ведомость пв ON рр.Код\_расписания\_рейсов = пв.Код\_расписания\_рейсов

WHERE DATE(рр.Дата\_прибытия) = '2017-05-05'

GROUP BY н.Код\_расписания\_направлений,нн.Код\_направления

ORDER BY sum DESC

3. Километраж предстоящий водителям за неделю от 1 мая до 7 мая. Водители отсортированы по имени.

SELECT э.Фамилия || ' ' || э.Имя || ' ' || э.Отчество AS "Ф.И.О.", sum(Расстояние) as Расстояние

FROM Экипаж э

INNER JOIN Расписание\_направлений р ON р.Код\_экипижа = э.Код\_сотрудника

INNER JOIN Расписание\_рейсов рр ON р.Код\_расписания\_направлений = рр.Код\_расписания\_направлений

INNER JOIN Рейс ре ON ре.Код\_рейса = рр.Код\_рейса

WHERE DATE(рр.Дата\_отправления) >= '2017-05-01' AND DATE(рр.Дата\_отправления) <= '2017-05-07'

GROUP BY э.Код\_сотрудника

ORDER BY "Ф.И.О." ASC