Московский авиационный институт

(Национальный исследовательский университет)

Институт № 8 «Компьютерные науки и прикладная математика»

Кафедра № 806 «Вычислительная математика и программирование»

**Лабораторная работа**

**по курсу «Базы данных»**

**по теме «Расписание поездов дальнего следования»**

Работу выполнили студенты группы М8О-305Б-20:

Черных С. Д., Чикунов И. В., Надоров И. С.

Преподаватель:

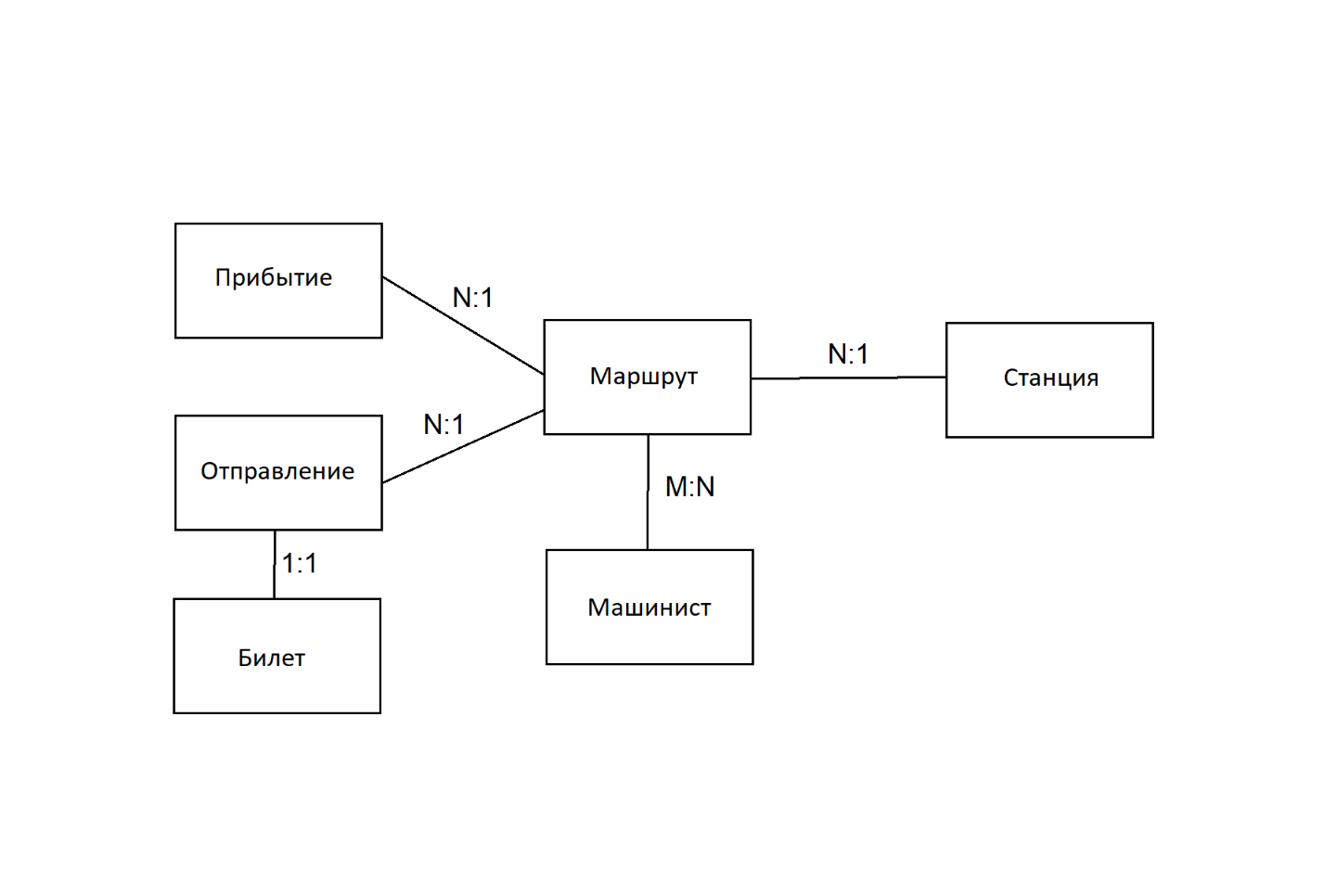
Киндинова В. В.

Оценка:

Дата

Москва, 2022

Схема базы данных:



Описание связей:

Маршрут соединяет две станции, причём для двух станций может быть множество маршрутов, отличающихся временем отправки и прибытия.

На каждый маршрут может быть поставлено несколько машинистов, так же, как и каждый машинист может работать по нескольким маршрутам.

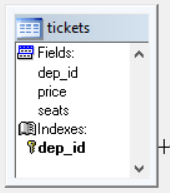
Каждый маршрут определён во временном интервале, который может повторяться изо дня в день, так и связаны между собой маршрут и прибытие с отправлением, которые существуют в определённые дни, в отличие от маршрута.

На каждое отправление есть свой билет.

Таблицы:

int – 4 байта, date – 8 байта, character – 30 символов

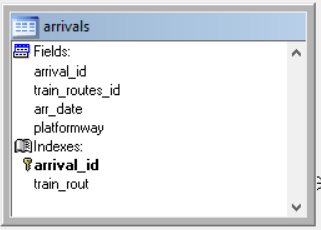
Билет:

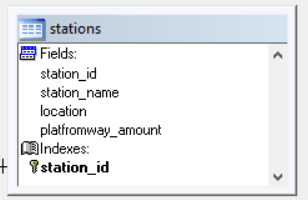
* dep\_id – номер отправления поезда (int), первичный ключ, связь с отправлениями
* price – цена билета (int)
* seats – количество свободных мест (int)

Отправление:

* departure\_id – номер отправления поезда, первичный ключ, связь с билетами (int)
* train\_routes\_id – внешний ключ, первичный для маршрута (int)
* dep\_date – день отправления (date)
* platformway – номер пути, с которого отправляется поезд (int)

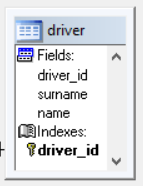
Прибытие:

* arrival\_id – номер прибытия, первичный ключ (int)
* train\_routes\_id – внешний ключ, первичный для маршрута (int)
* arr\_date – день прибытия (date)
* platformway – номер путь, на который прибывает поезд (int)

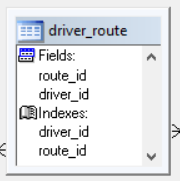
Станция:

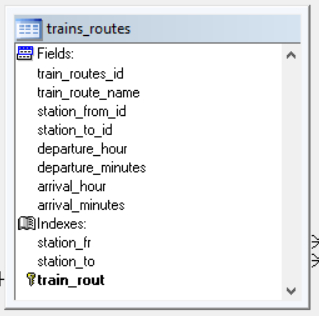
* station\_id – номер станции, первичный ключ, связь с маршрутами (int)
* station\_name – название станции/вокзала (character)
* location – город, в котором находится станция (character)
* platform\_amount – количество путей станции (int)

Машинист:

* driver\_id – номер машиниста, первичный ключ (int)
* surname – фамилия машиниста (character)
* name – имя машиниста (character)

Таблица, связывающая машинистов и маршруты:

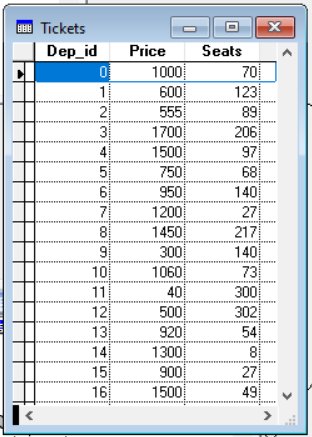
* route\_id – внешний ключ, первичный для маршрута (int)
* driver\_id – внешний ключ, первичный для машиниста (int)

Маршрут:

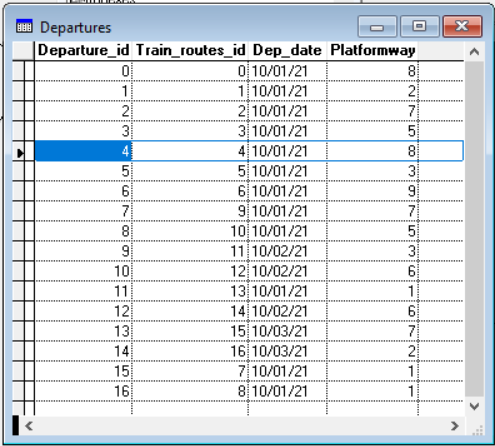
* train\_routes\_id – номер маршрута, первичный ключ (int)
* train\_route\_name – название маршрута (character)
* station\_from\_id – станция отправления, внешний ключ (int)
* station\_to\_id – станция прибытия, внешний ключ (int)
* departure\_hour – час отправления (int)
* departure\_minutes – минуты отправления (int)
* arrival\_hour – час прибытия (int)
* arrival\_minutes – минуты прибытия (int)

Таблицы с данными:

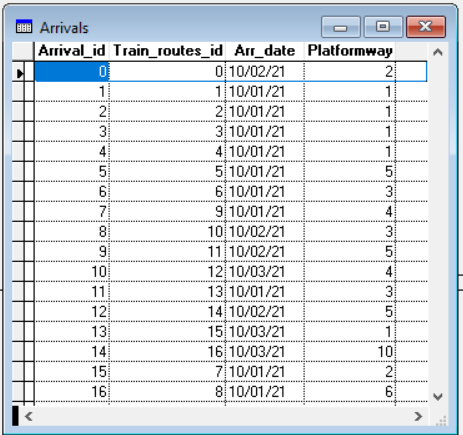
Билет:



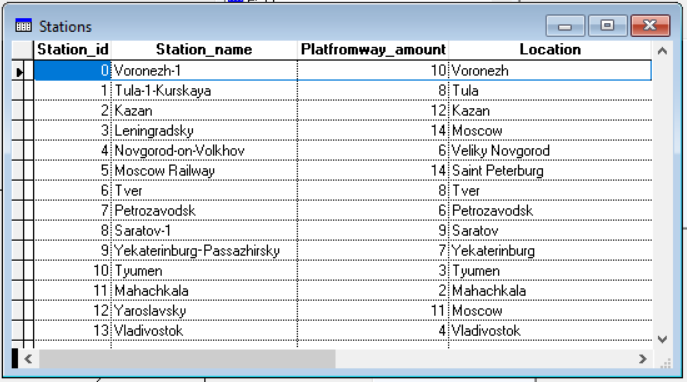
Отправление:



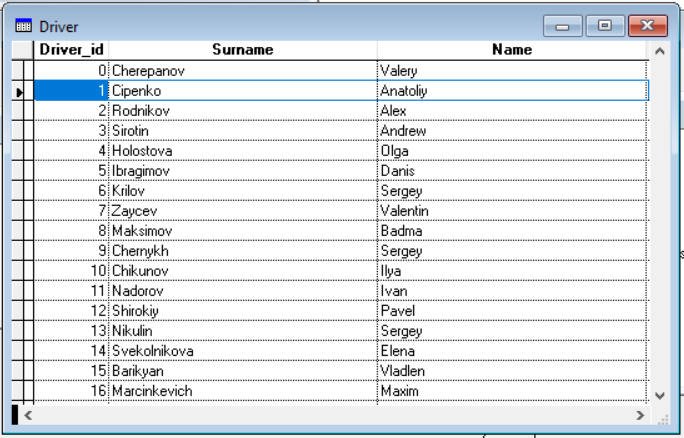
Прибытие:



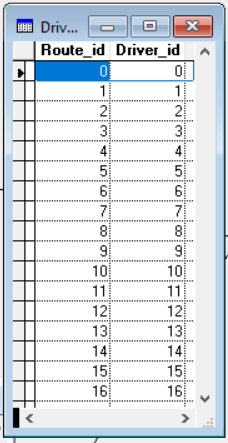
Станция:



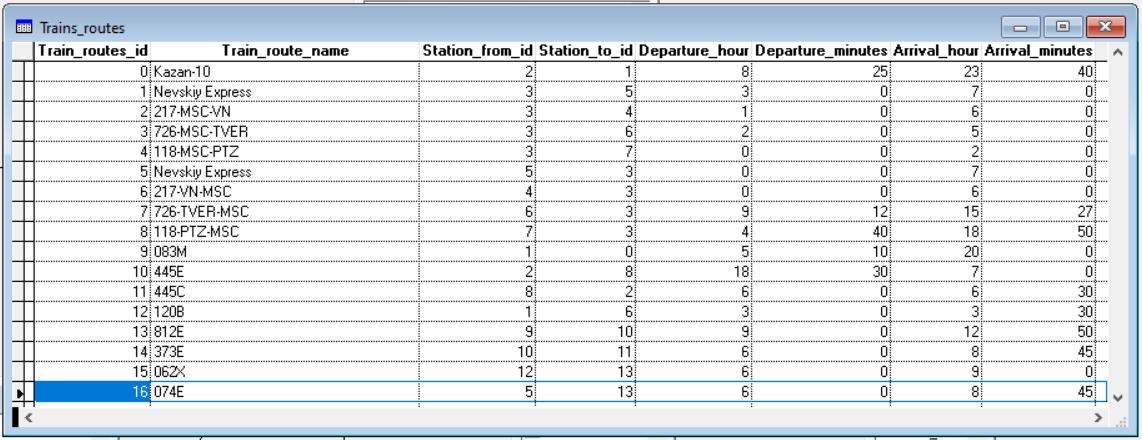
Машинист:



Таблица, связывающая машинистов и маршруты:



Маршрут:



Простые запросы:

1. Выбрать самое раннее время и самую раннюю дату для всех станций, на которые прибывают поезда, а также название станций и их расположение:

SELECT Stations.station\_name, Stations.location,;

MIN(Arrivals.arr\_date) AS earliest\_date,;

MIN(Trains\_routes.arrival\_hour) AS earliest\_hour,;

MIN(Trains\_routes.arrival\_minutes) AS earliest\_minutes;

FROM ;

data1!stations,;

data1!trains\_routes,;

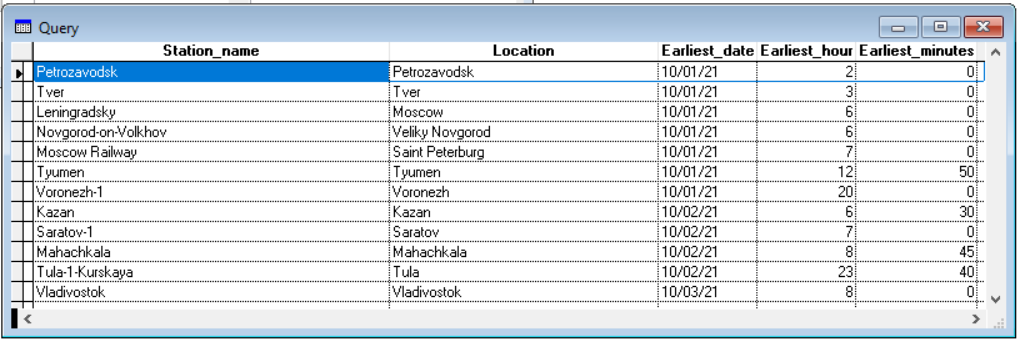
data1!arrivals;

WHERE ( station\_id ) = ( station\_to\_id );

AND Arrivals.train\_routes\_id = Trains\_routes.train\_routes\_id;

GROUP BY Stations.station\_name, Stations.location;

ORDER BY 3, 4



1. Показать таблицу отправлений в порядке убывания даты и времени отправки:

SELECT Departures.\*, Trains\_routes.departure\_hour,;

Trains\_routes.departure\_minutes;

FROM ;

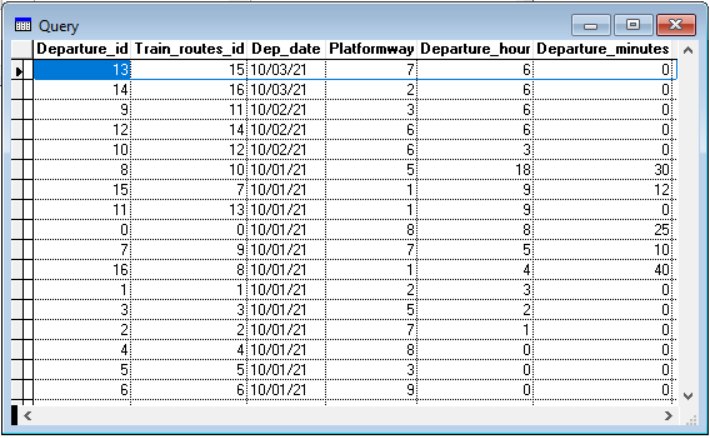
data1!departures,;

data1!trains\_routes;

WHERE Departures.train\_routes\_id = Trains\_routes.train\_routes\_id;

ORDER BY Departures.dep\_date DESC,;

Trains\_routes.departure\_hour DESC



1. Выбрать все номера станций, с которых отправятся поезда 01.10.2021:

SELECT DISTINCT Trains\_routes.station\_from\_id, Departures.dep\_date;

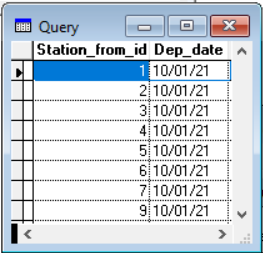
FROM ;

departures,;

trains\_routes;

WHERE Departures.dep\_date = DATE(2021,10,1);

AND Departures.train\_routes\_id = Trains\_routes.train\_routes\_id



1. Показать таблицу маршрутов, у которых даты прибытия совпадают с датами отправления:

SELECT Trains\_routes.\*, Departures.dep\_date, Arrivals.arr\_date;

FROM ;

data1!trains\_routes,;

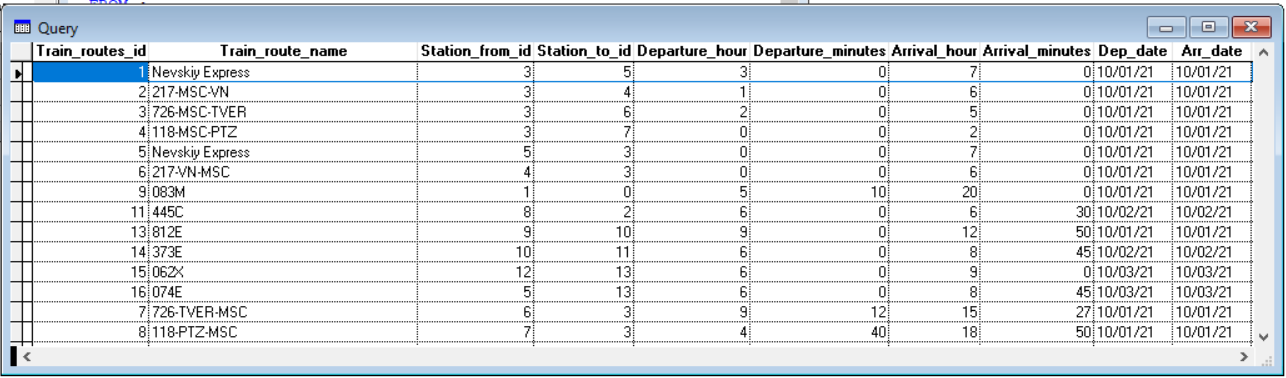
departures,;

arrivals;

WHERE ( Trains\_routes.train\_routes\_id = Departures.train\_routes\_id;

AND Trains\_routes.train\_routes\_id = Arrivals.train\_routes\_id );

AND ( dep\_date ) = ( arr\_date )



1. Выбрать станции (имя и город), на которые прибудет поезд 03.10.2021, показать имя маршрута:

SELECT Stations.location, Stations.station\_name,;

Trains\_routes.train\_route\_name, Arrivals.arr\_date;

FROM ;

stations,;

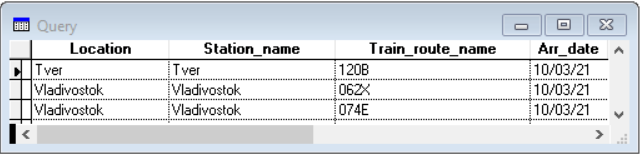
trains\_routes,;

arrivals;

WHERE ( Stations.station\_id = Trains\_routes.station\_to\_id;

AND Trains\_routes.train\_routes\_id = Arrivals.train\_routes\_id );

AND ( arr\_date ) = DATE(2021,10,3)



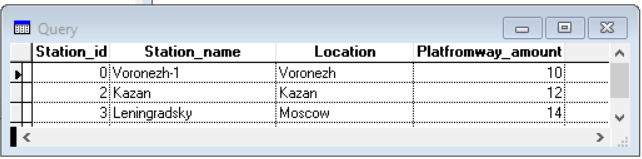
1. Показать таблицу станций, число путей которых не меньше 10:

SELECT Stations.\*;

FROM ;

data1!stations;

WHERE Stations.platfromway\_amount >= ( 10 )



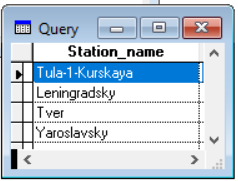
1. Выбрать имена станций, расположенных в Москве, Туле и Твери:

SELECT Stations.station\_name;

FROM ;

data1!stations;

WHERE Stations.location IN (“Tver”,”Moscow”,”Tula”)



1. Показать таблицу станций, где первая буква города - «М» или вторая – «е»:

SELECT Stations.\*;

FROM ;

data1!stations;

WHERE Stations.location LIKE ( “M%” );

OR Stations.location LIKE ( “\_e%” )



Сложные запросы:

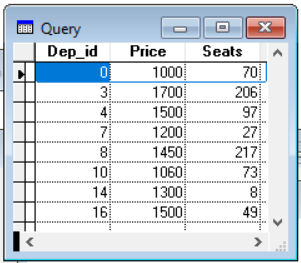
1. Выбрать билеты, цена которых не меньше средней:

SELECT Tickets.\*;

FROM ;

tickets;

WHERE Tickets.price >= (SELECT AVG(Tickets.price) FROM Tickets)



1. Выбрать станцию, с которой связано больше всего маршрутов, указать их количество:

SELECT COUNT(Trains\_routes.train\_routes\_id) AS count\_of\_routes,;

Trains\_routes.station\_from\_id AS station;

FROM ;

trains\_routes;

GROUP BY Trains\_routes.station\_from\_id;

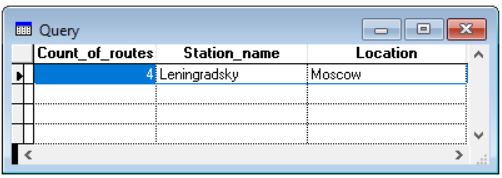
INTO CURSOR tmp

SELECT count\_of\_routes, station\_name, location;

FROM tmp, Stations;

WHERE count\_of\_routes >= ALL(SELECT count\_of\_routes FROM tmp);

AND station\_id = station;



1. Выбрать все маршруты, где прибытие должно произойти 02.10.2021:

SELECT Arrivals.arr\_date AS \_date;

FROM ;

Arrivals;

INTO CURSOR tmp

SELECT Trains\_routes.train\_routes\_id, Trains\_routes.train\_route\_name,;

tmp.\_date as \_date,;

Trains\_routes.arrival\_hour AS \_hour,;

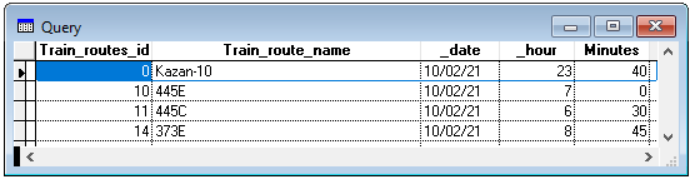
Trains\_routes.arrival\_minutes AS minutes;

FROM ;

trains\_routes;

WHERE Exists (SELECT train\_routes\_id FROM Arrivals;

WHERE arr\_date = Date(2021, 10, 02) AND train\_routes\_id = trains\_routes.train\_routes\_id)



1. Показать отправления, если номер пути, с которого отправляется поезд, присутствует в номерах путей таблицы прибытия:

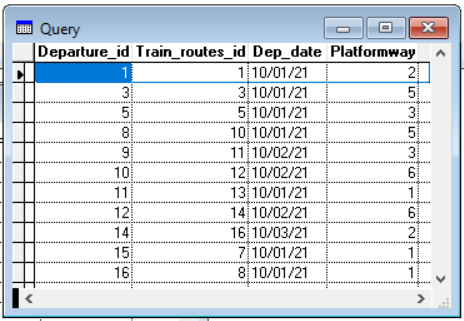
SELECT Departures.\*;

FROM ;

departures;

WHERE EXISTS (SELECT arrivals.platformway FROM arrivals;

WHERE departures.platformway = arrivals.platformway)



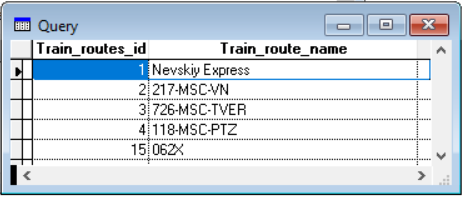
1. Выбрать те маршруты, у которых станция отправления находятся в Москве:

SELECT Trains\_routes.train\_routes\_id, Trains\_routes.train\_route\_name;

FROM ;

data1!trains\_routes;

WHERE ( station\_from\_id ) IN (SELECT station\_id FROM Stations WHERE Location = "Moscow")



1. Выбрать всевозможные пары имён маршрутов, кроме совпадающих, у которых совпадают станции отправления:

SELECT Fir.train\_route\_name AS id\_f1, Se.train\_route\_name AS id\_f2,;

Fir.station\_from\_id;

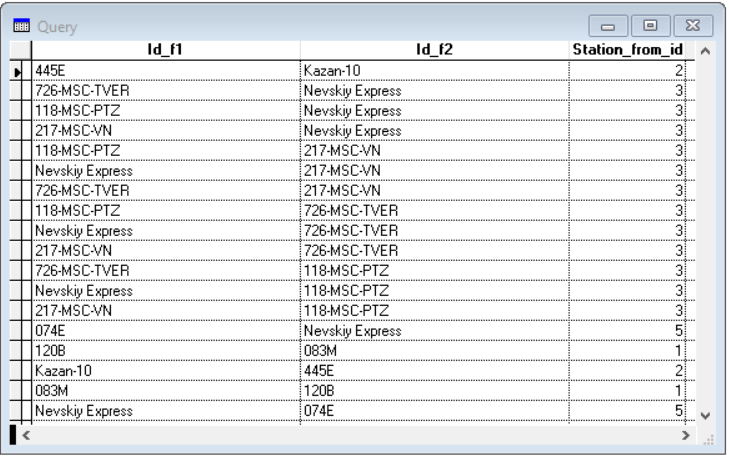
FROM ;

data1!trains\_routes Fir,;

data1!trains\_routes Se;

WHERE Fir.station\_from\_id = Se.station\_from\_id;

AND Fir.train\_route\_name <> Se.train\_route\_name



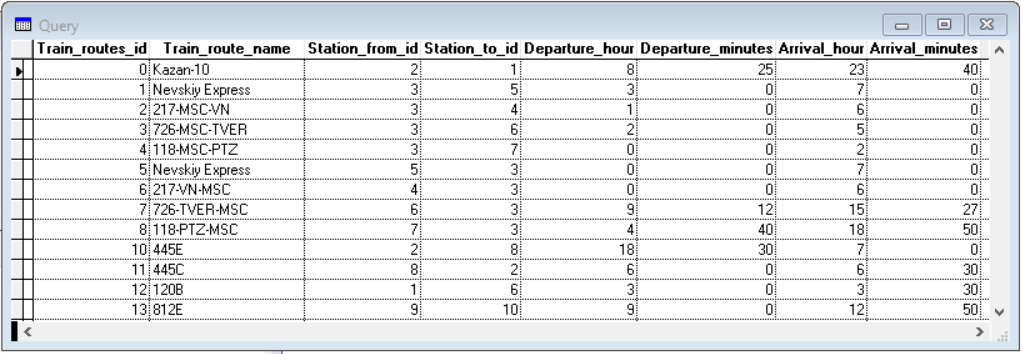
1. Выбрать все маршруты, у которых станция прибытия встречается среди всех станций отправления:

SELECT \*;

FROM ;

trains\_routes;

WHERE ( station\_to\_id ) = SOME (SELECT station\_from\_id FROM trains\_routes)



1. Выбрать те маршруты, время в пути которых больше среднего времени пути маршрутов из Москвы:

SELECT ((Arrivals.arr\_date-Departures.dep\_date)\*24\*60+(arrival\_hour\*60+arrival\_minutes)-(departure\_hour\*60+departure\_minutes)) AS traveltime;

FROM ;

trains\_routes,;

arrivals,;

departures;

WHERE ( Trains\_routes.train\_routes\_id = Arrivals.train\_routes\_id;

AND Trains\_routes.train\_routes\_id = Departures.train\_routes\_id );

AND ( station\_from\_id ) IN (3,12);

INTO CURSOR tmp

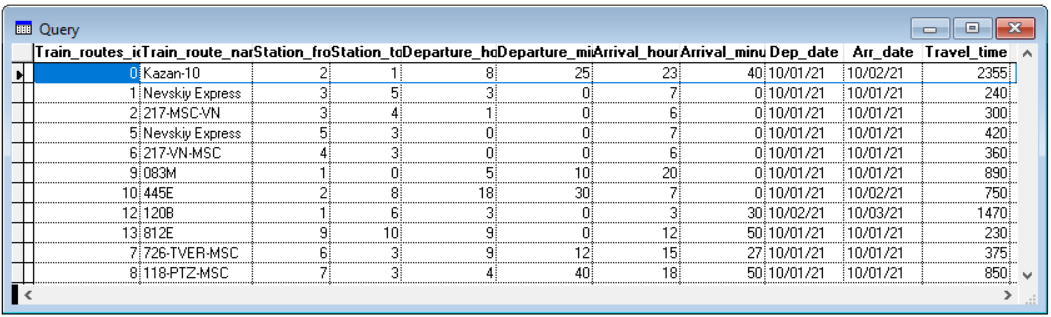
SELECT trains\_routes.\*, departures.dep\_date, arrivals.arr\_date,((arrivals.arr\_date - departures.dep\_date)\*24\*60 + (arrival\_hour\*60+arrival\_minutes) - (departure\_hour\*60+departure\_minutes)) as travel\_time;

FROM trains\_routes, arrivals, departures;

WHERE trains\_routes.train\_routes\_id = arrivals.train\_routes\_id AND;

trains\_routes.train\_routes\_id = departures.train\_routes\_id AND;

((arrivals.arr\_date - departures.dep\_date)\*24\*60 + (arrival\_hour\*60+arrival\_minutes) - (departure\_hour\*60+departure\_minutes)) > (SELECT AVG(traveltime) FROM tmp);



1. Показать всех машинистов, которые должны отправиться в путь 02.10.2021 или отправляются из Саратова, допускаются повторения:

FROM driver, departures, arrivals,;

trains\_routes, stations, driver\_route;

WHERE driver.driver\_id = driver\_route.driver\_id;

AND driver\_route.route\_id = trains\_routes.train\_routes\_id;

AND trains\_routes.train\_routes\_id = departures.train\_routes\_id;

AND trains\_routes.train\_routes\_id = arrivals.train\_routes\_id;

AND trains\_routes.station\_from\_id = stations.station\_id;

AND trains\_routes.station\_from\_id = 8;

UNION ALL;

SELECT driver.\*, dep\_date, arr\_date, station\_name as \_from;

FROM driver, departures, arrivals,;

trains\_routes, stations, driver\_route;

WHERE driver.driver\_id = driver\_route.driver\_id;

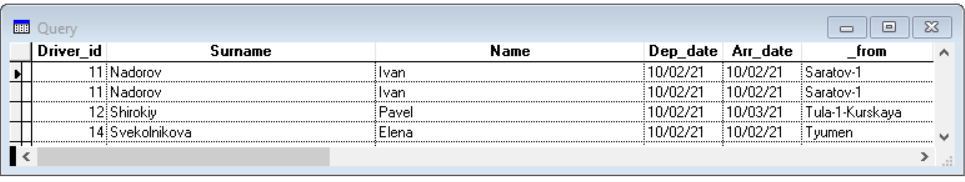
AND driver\_route.route\_id = trains\_routes.train\_routes\_id;

AND trains\_routes.train\_routes\_id = departures.train\_routes\_id;

AND trains\_routes.train\_routes\_id = arrivals.train\_routes\_id;

AND trains\_routes.station\_from\_id = stations.station\_id;

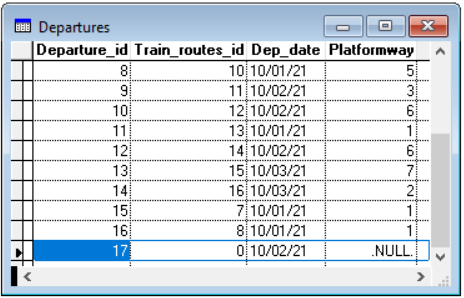
AND dep\_date = DATE(2021, 10, 02);



1. Добавить в таблицу отправлений: номер отправления – 17, номер маршрута – 0, дата отправления – 02.10.2021, путь – NULL:

INSERT INTO departures (departure\_id, train\_routes\_id, dep\_date, platformway);

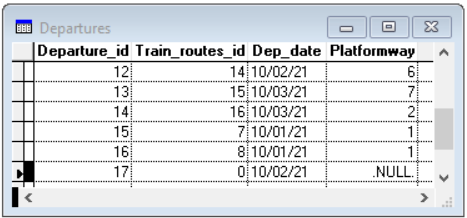
VALUES (17, 0, {^2021/10/02}, NULL);



1. Удалить отправление под номером 17:

DELETE FROM departures;

WHERE departure\_id = 17;

\*

\*Microsoft Visual FoxPro пометил эту строку под удаление

1. Изменить отправление под номером 4: поменять номер пути на с 8 на 9, если он не будет занят другим поездом в этот день:

SELECT Trains\_routes.station\_from\_id;

FROM ;

trains\_routes,;

departures;

WHERE Trains\_routes.train\_routes\_id = Departures.train\_routes\_id;

INTO CURSOR tmp

UPDATE departures SET platformway = 9;

WHERE departure\_id = 4 AND NOT EXISTS;

(SELECT platformway;

FROM departures, trains\_routes;

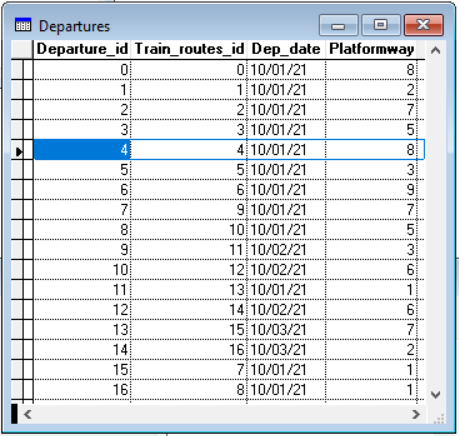
WHERE trains\_routes.station\_from\_id = tmp.station\_from\_id;

AND trains\_routes.train\_routes\_id = departures.train\_routes\_id;

AND platformway = 9;

AND dep\_date = DATE(2021, 10, 01));

До применения запроса:



После применения запроса:

