МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УО «ПОЛОЦКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет информационных технологий

Кафедра технологий программирования

КУРСОВОЙ ПРОЕКТ

По дисциплине «Основы алгоритмизации и программирования»

Тема: «База данных Топ самых быстрых поездов мира»

|  |  |
| --- | --- |
| Выполнил: | студент гр.16-ИТ-3, Бадретдинов А.Р. |
| Проверил: | преподаватель-стажер, В.В. Магеров |

Новополоцк,

2017 г.

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 4](file:///D:\Учёба\ПГУ\ОАиП\Курсовая\СОДЕРЖАНИЕ.docx#_Toc421140012)

[1 АНАЛИЗ ЗАДАНИЯ И ПОСТАНОВКА ЗАДАЧИ 5](file:///D:\Учёба\ПГУ\ОАиП\Курсовая\СОДЕРЖАНИЕ.docx#_Toc421140013)

[2 ТЕОРЕТИЧЕСКАЯ ЧАСТЬ 7](file:///D:\Учёба\ПГУ\ОАиП\Курсовая\СОДЕРЖАНИЕ.docx#_Toc421140014)

[3 ПРОЕКТИРОВАНИЕ ПРОГРАММЫ 9](file:///D:\Учёба\ПГУ\ОАиП\Курсовая\СОДЕРЖАНИЕ.docx#_Toc421140015)

[4 РЕАЛИЗАЦИЯ ПРОГРАММЫ 11](file:///D:\Учёба\ПГУ\ОАиП\Курсовая\СОДЕРЖАНИЕ.docx#_Toc421140016)

[5 ТЕСТИРОВАНИЕ ПРОГРАММЫ 13](file:///D:\Учёба\ПГУ\ОАиП\Курсовая\СОДЕРЖАНИЕ.docx#_Toc421140017)

[6 ЗАКЛЮЧЕНИЕ 22](file:///D:\Учёба\ПГУ\ОАиП\Курсовая\СОДЕРЖАНИЕ.docx#_Toc421140018)

[СПИСОК ЛИТЕРАТУРЫ 23](file:///D:\Учёба\ПГУ\ОАиП\Курсовая\СОДЕРЖАНИЕ.docx#_Toc421140019)

[ПРИЛОЖЕНИЕ А 24](file:///D:\Учёба\ПГУ\ОАиП\Курсовая\СОДЕРЖАНИЕ.docx#_Toc421140020)

[ПРИЛОЖЕНИЕ Б 26](file:///D:\Учёба\ПГУ\ОАиП\Курсовая\СОДЕРЖАНИЕ.docx#_Toc421140021)

[ПРИЛОЖЕНИЕ В 32](file:///D:\Учёба\ПГУ\ОАиП\Курсовая\СОДЕРЖАНИЕ.docx#_Toc421140021)

Изм.

Лист

№ докум.

Подпись

Дата

Лист

3

Б А Р. 1610503. ПЗ

Разраб.

Бадретдинов А.Р.

Провер.

Магеров В.В.

Реценз.

Н. Контр.

Утверд.

.

Однофайловая база данных

«Топ самых быстрых поездов мира»

Лит.

Листов

32

УО «ПГУ» 16-ИТ-3

**ВВЕДЕНИЕ**

После года изучения дисциплины «Основы алгоритмизации и програмирования» учебными планами предусмотрено написание курсовой работы, чтобы студент мог применить полученные знания и умения для написания более объемной работы, которая может быть использована для решения серьезной задачи.

Спустя год по данной дисциплине мы изучили множество тем, выполнили немало лабораторных работ, и полученных знаний и опыта достаточно, чтобы написать курсовой проект, который будет удовлетворять всем предъявлеямым требованиям.

Чтобы выполнить данную задачу, в нашем распоряжении есть весь богатый синтаксис языка программирования С++. Этого достаточно, чтобы успешно реализовать поставленную нам задачу.

Написание курсовой работы это отличная возможность для студента решить более серьезную задачу и реализовать действительно объемный проект. При этом студент может раскрыть весь свой потенциал, продемонстрировать свои умения, приобритенные не только в университете, но и в результате самостоятельного изучения материала.

На написание такой работы необходимо достаточно времени, и это также позволяет раскрыть такие качества, как усидчивость, терпение, умение ставить себе конкретную задачу и решать её в срок, грамотно распределять свои время и силы.

Полученный в результате опыт будет бесценным, и приобритенные умения и качества обязательно пригодятся при выполнении более сложных проектов в будущем.

Изм.

Лист

№ докум.

Подпись

Дата

Лист

4

Курсовой проект

**1 АНАЛИЗ ЗАДАНИЯ И ПОСТАНОВКА ЗАДАЧИ**

Изм.

Лист

№ докум.

Подпись

Дата

Лист

5

Курсовой проект

В данном курсовом проекте нам предстоит разработать приложение, которое содержит базу данных о высокоскоростных поездах.

Программа будет содержать и оперировать следующими данными:

1. Название поезда – поле, которое будет содержать название железнодорожного транспортного средства. Поле может содержать несколько слов;
2. Скорость – поле, содержащее рабочую скорость поезда. Единицы измерения – километр в час (км/ч);
3. Максимальная скорость – поле, содержащее максимально развитую скорость, установленную поездом на испытаниях. Единицы измерения – километр в час (км/ч);
4. Страна – поле, содержащее название страны, где поезд курсирует;
5. Компания – поле, содержащее полное наименование компании, которая разработала высокоскоростной поезд;
6. Страна – поле, содержащее название страны, в которой был выпущен высокоскоростной поезд;
7. Количество пассажиров – поле, содержащее число пассажирских мест;
8. Дата – поле, содержащее полную дату первого испытания высокоскоростного поезда. Формат даты – ДД.ММ.ГГГГ;

Определившись с данными, содержащихся в базе данных, далее следует четко установить задачи, выполняемые нашим приложением с базой данных, а именно, в базе данных необходимо предусмотреть и полностью реализовать следующие операции:

1. Добавление записи о конкретном высокоскоростном поезде;
2. Просмотр всех добавленных записей в виде таблицы с заголовками каждого поля;
3. Подробный просмотр отдельно выбранной записи со всеми полями отдельно от таблицы с записями;
4. Удаление выбранной записи;
5. Редактирование записи по отдельному полю;

Изм.

Лист

№ докум.

Подпись

Дата

Лист

6

Курсовой проект

1. Поиск записей в базе данных по любому полю, предусматривая частичное совпадение;
2. Сортировка записей в базе данных по любому полю в любом направлении;
3. Возможность сохранение результатов в файл;
4. Загрузка сохраненных ранее результатов из файла в базу данных;
5. Вывод результатов на печать в виде текстового файла;

Также, кроме вышеописанных функций, в программе должен быть предусмотрен удобный и понятный интерфейс, помогающий пользователю без труда взаимодействовать с программой. И данному аспекту будет уделено особое внимание.

Внимание также будет уделено обработке исключений, которые будут предусматривать ошибки, которые могут возникнуть при работе с программой, и незамедлительное оповещение об этом пользователя.

Когда мы четко установили цели и задачи данного курсового проекта, приступим к обсуждению алгоритмов, которые должны быть реализованы в данном приложении в обязательном порядке.

**2 ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ**

Изм.

Лист

№ докум.

Подпись

Дата

Лист

7

Курсовой проект

Задача нашей программы заключается в работе с данными, их обработка и хранение. Данную задачу будет выполнять такой элемент языка С++, как структура. Определение которой представлено ниже.

Итак, что же понимается под структурой?

Структура – это сложный тип данных представляющий собой упорядоченное в памяти множество элементов различного типа. Каждый элемент в структуре имеет свое имя и называется полем. Размер структуры определяется суммой размеров всех элементов.

В качестве структуры данных, которая позволит нам работать с данными, был выбран Дек.

Дек – это структура данных, представляющая из себя двунаправленную очередь, в которой добавление новых элементов производится с конца списка, а удаление элементов – из начала.

Как уже было сказано выше, дек совмещает в себе двунаправленную очередь, а именно, он позволяет производить добавление и удаление элементов как из начала списка, так и с конца.

Эта структура поддерживает как FIFO (Первым пришлом, первым и вышел), так и LIFO (Последним пришел, первым ушел) и поэтому, в деке можно реализовать как стек, так и очередь.

Любой дек поддерживает как минимум четыре операции:

1. добавление элемента в начало списка;
2. добавление элемента в конец списка;
3. проверка на наличие элементов в списке;
4. удаление элемента из начала списка;
5. удаление элементов из конца списка.

Также стоит отметить, что в наши задачи входит создание с нуля структуры данных, а именно, дека, на основе связаного хранения, используя указатели. О чем мы собственно и поговорим ниже.

Указатель **–** этопеременная, значением которой является адрес, по которому располагаются данные. В нашем случае, указатель будет хранить адрес одной записи, по которому будут располагаться подробные сведения о записи.

Изм.

Лист

№ докум.

Подпись

Дата

Лист

8

Курсовой проект

Под адресом понимается номер ячейки памяти, в которой или с которой располагаются данные.

Для того, чтобы каждый раз добавлять новый элемент в нашу структуру данных, нам предстоит работа с динамической памятью.

Под динамической памятью понимают следующее:

Динамическая память – это область (блок) памяти выделенный для нужд программы в процессе работы программы (а не заранее).

Основными действиями над динамической памятью являются: выделение и освобождение.

Также при работе с данными нам потребуется сохранять результат работы в файл, чтобы иметь возможность получить доступ и обрабатывать заранее введенные данные.

Под файлом понимается следующее:

Файл – именованная область данных на каком-либо носителе информации (жесткий диск, дискета, компакт-диск и т.д.).

Язык С++, как и другие языки программирования высокого уровня, позволяет осуществлять операции файлового ввода и вывода. Основной алгоритм обработки файлов выполняется в три действия в следующем порядке: открытие файла, чтение и/или запись данных в файл, а также закрытие файла.

Наконец, определившись со структорой данных и некоторыми алгоритмами, которые понадобятся нам в выполнии данного курсового проекта, мы можем приступить к подробному описанию выбранных нами алгоритмов, которые будут обеспечивать полное функционирование нашей программы.

**3 ПРОЕКТИРОВАНИЕ ПРОГРАММЫ**

Изм.

Лист

№ докум.

Подпись

Дата

Лист

9

Курсовой проект

При запуске программы, пользователь увидит главное меню программы, содержащее определенные пункты:

* Приступить к работе
* Выход из программы

Навигацией в главном и в последующих меню служат стрелочные клавиши: вверх, вних, влево и вправо, а за выбор пункта отвечает клавиша «Enter».

При выборе пункта «Приступить к работе» – пользовать перейдет в основной раздел программы, отвечающий за работой с данными.

После перехода в основной раздел пользователь увидит меню для работы со структурой данных:

* Добавить запись;
* Загрузить из файла;
* Сохранить в файл;
* Очистить таблицу;
* Поиск;
* Сортировка;
* Меню программы.

При выборе пункта «Добавить запись» – пользователь увидит форму добавления записи, с кнопками «Завершить ввод» и «Отмена».

При выборе пункта «Загрузить из файла» – пользователь увидит специальное поле для ввода имени файла, данные которого будут загружены в таблицу.

При выборе пункта «Сохранить в файл» – пользователь увидит форму для ввода имени файла, в который будут сохранятся результаты работы с программой.

При выборе пункта «Очистить таблицу» – программа удалит все данные, тем самым очистив таблицу.

При выборе пункта «Поиск» – пользователю будет продложена форма для поиска записей в таблице по полному или частичному совпадению.

Изм.

Лист

№ докум.

Подпись

Дата

Лист

10

Курсовой проект

При выборе пункта «Сортировка» – пользователь увидет вверху программы меню для сортировки записей по определенным полям. Управляя стрелочными клавишами пользователь сможет передвигаться по заголовкам полем для выбора конкретного сортируемого поля.

При выборе пункта «Меню программы» – программа сделает возврат в главное меню.

Также, будет реализовано дополнительное меню для отдельной записи, которое будет включать в себя следующие пункты:

* Просмотреть запись
* Редактировать запись
* Удалить запись

При выборе пункта «Просмотреть запись» – программа отобразит на экране сведения об отдельно выбранной записи.

При выборе пункта «Редактировать запись» – курсор переместится в первое поле выбранной записи тем самым предоставляя пользователю инструмент редактирования.

При выборе пункта «Удалить запись» - будет выполненно удаление записи с экрана таблицы.

**4 РЕАЛИЗАЦИЯ ПРОГРАММЫ**

Изм.

Лист

№ докум.

Подпись

Дата

Лист

11

Курсовой проект

Для реализации программы была создана следующая структура данных, выполняющая роль дека:

Листинг 1 – структура Deque

1: typedef struct

2: {

3: char name[33], country[32], by\_country[32],

4: company[32];

5: char speed[6], max\_speed[6], pass\_count[4], date[11];

6: } DataType;

7:

8: struct Element

9: {

10: DataType data;

11: Element \*prev, \*next;

12: };

13:

14: struct Deque

15: {

16: Element \*Start, \*End;

17: unsigned int size;

18: };

Листинг 2 – модуль main

1: #include "Menu.h"

2:

3: int main()

4: {

5: Menu();

6: return 0;

7: }

Модуль **main.cpp** :

*main* – запускает функцию *Menu* из модуля **Menu.cpp**

Модуль **Table.cpp** :

void GetDeque(); - Отображение таблицы и меню

Модуль **Record.cpp** :

void PutDequeBegin(); - добавление элемета вначало списка

void Del\_Edit\_DequeElement(Element\*, int&, int); - удаление элемента из любой позиции списка

void Print\_Element(Element\*, int, int); - вывод элемента в таблице

void Scrolling(int, Element\*, Element\*, boolean); - прокрутка элементов в таблицы

void Move\_Cursor(Element\*, Element\*, boolean, short, int, int); - выделение текущей записи в таблице

void Item\_View(Element\*); - вывод отдельной записи

void Clear\_Screen(int, Element\*, Element\*, int, boolean, Element\*); - очистка экрана без потери данных в таблице

Изм.

Лист

№ докум.

Подпись

Дата

Лист

12

Курсовой проект

Модуль **Deque.cpp** :

void InitDeque(); - инициализация дека

int CheckEmptyDeque(); - проверка дека на наличие элементов

void Deque\_Clear(); - очистка дека

void quick\_sort(DataType\*, int, int, int, int, int); - сортировка записей медотом быстрой сортировки(сортировка Хоара)

void search(); - поиск записей в таблице

Модуль **File.cpp** :

void Load\_Deque(); - сохранение данных в файл

void Save\_Deque(); - загрузка данных из файла

Модуль **Menu.cpp** :

void Menu(); - открывает меню программы

Модуль **Check.cpp** :

void SetColor(int, int); - устанавливает цвет текста и контура.

void SetColor(int, ConsoleColor); - устанавливает цвет текста и его контура

void input\_int(char\*, char\*, int, short, short); - ввод целого числа с одновременной проверкой на ввод

void input\_float(char\*, char\*, int, short, short); - ввод вещественного числа с одновременной проверкой на ввод

void input\_str(char\*, char\*, int, int, short, int, short, int); - ввод строки с одновременной проверкой на ввод

void input\_date(char\*, char\*, int, short, short); - ввод даты с одновременной проверкой на ввод

**5 ТЕСТИРОВАНИЕ ПРОГРАММЫ**

После запуска программы пользователь увидит главное меню программы.

Изм.

Лист

№ докум.

Подпись

Дата

Лист

13

Курсовой проект

Данное меню представлено на рисунке 5.1.

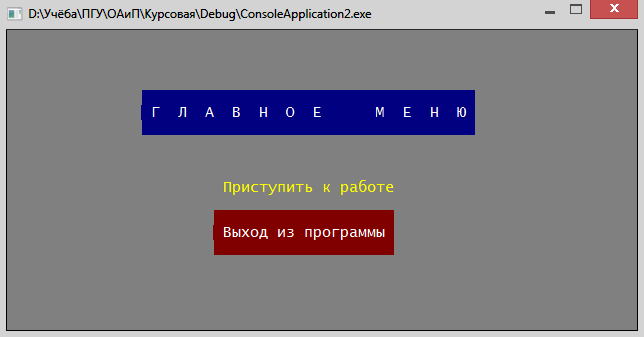


Рисунок 5.1 – Главное меню программы

Нажимая «Приступить к работе» вы переходите в рабочую область программы. Изначально в таблице уже содержит данные, загруженные из текстового файла. Таблица представлена на рисунке 5.2.

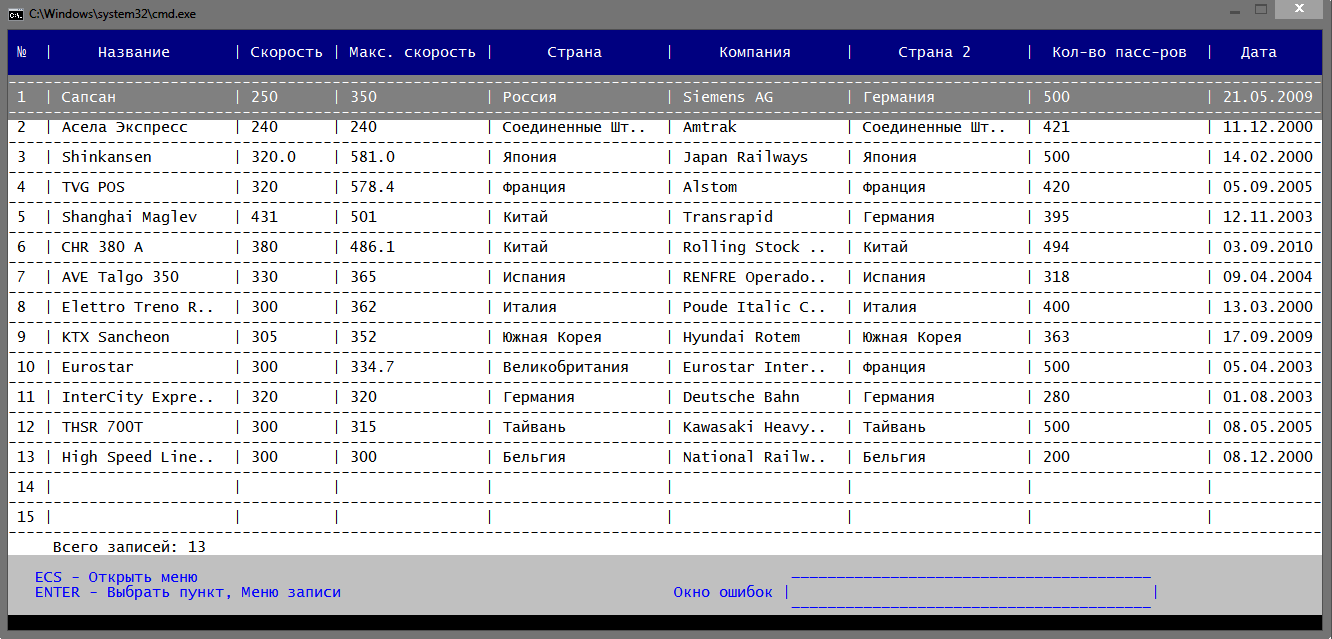


Рисунок 5.2 – Таблица с данными

Нажав клавишу «ECS» пользователь увидит главное меню таблицы.

Изм.

Лист

№ докум.

Подпись

Дата

Лист

14

Курсовой проект

Данное действите представлено на рисунке 5.4.

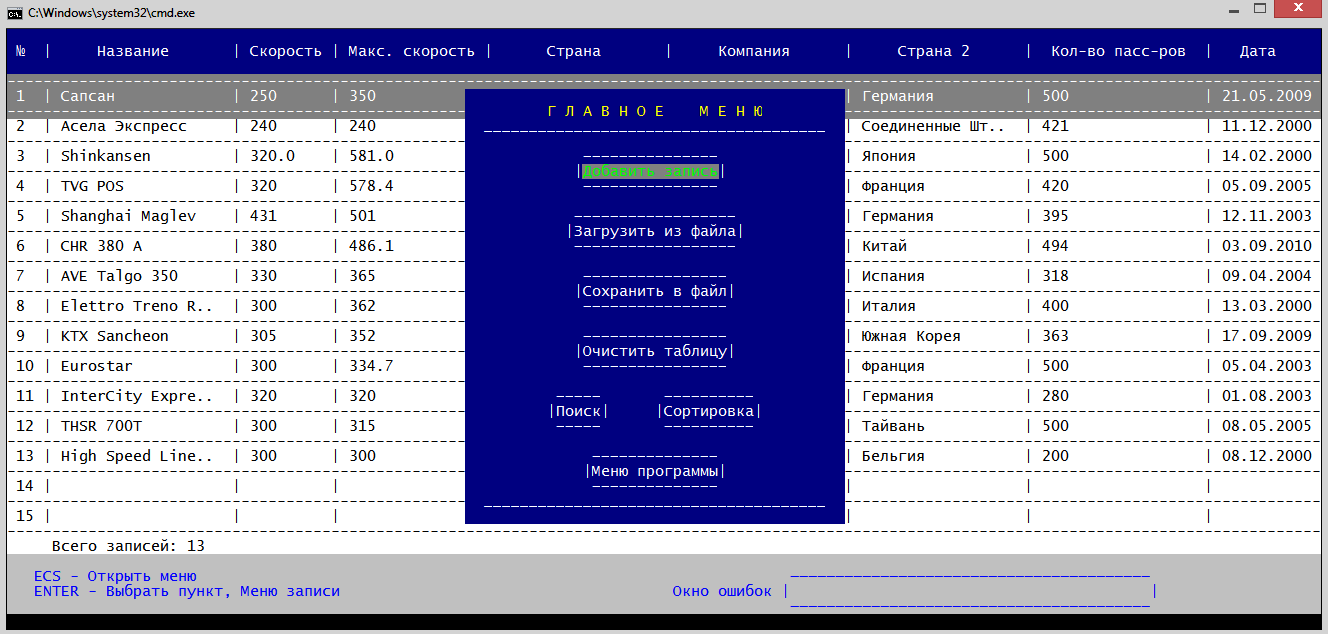


Рисунок 5.4 – Главное меню таблицы

Выбрав пункт «Добавить запись», пользователь увидит форму с добавление записи.

Данное действие проиллюстрировано на рисунке 5.5.

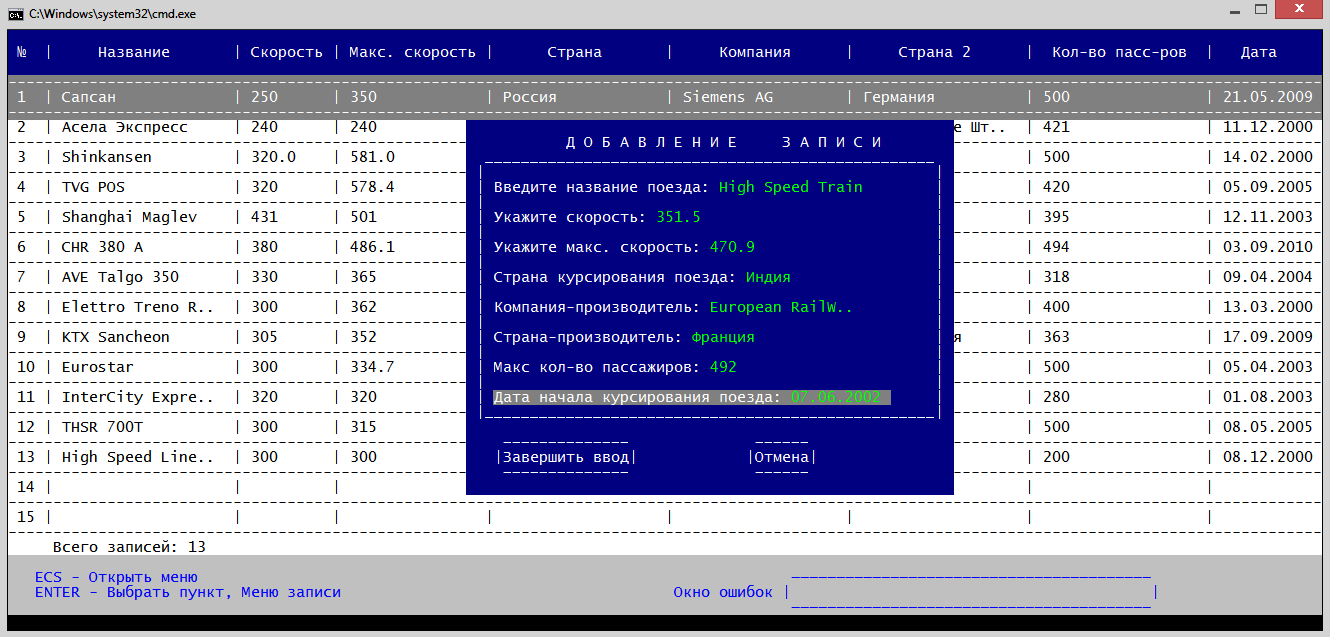


Рисунок 5.5 – Добавление записи

Выбрав пункт «Загрузить из файла», пользователю будет предложена форму, в которую нужно вписать имя файла.

Данное действие проиллюстрировано на рисунке 5.6.

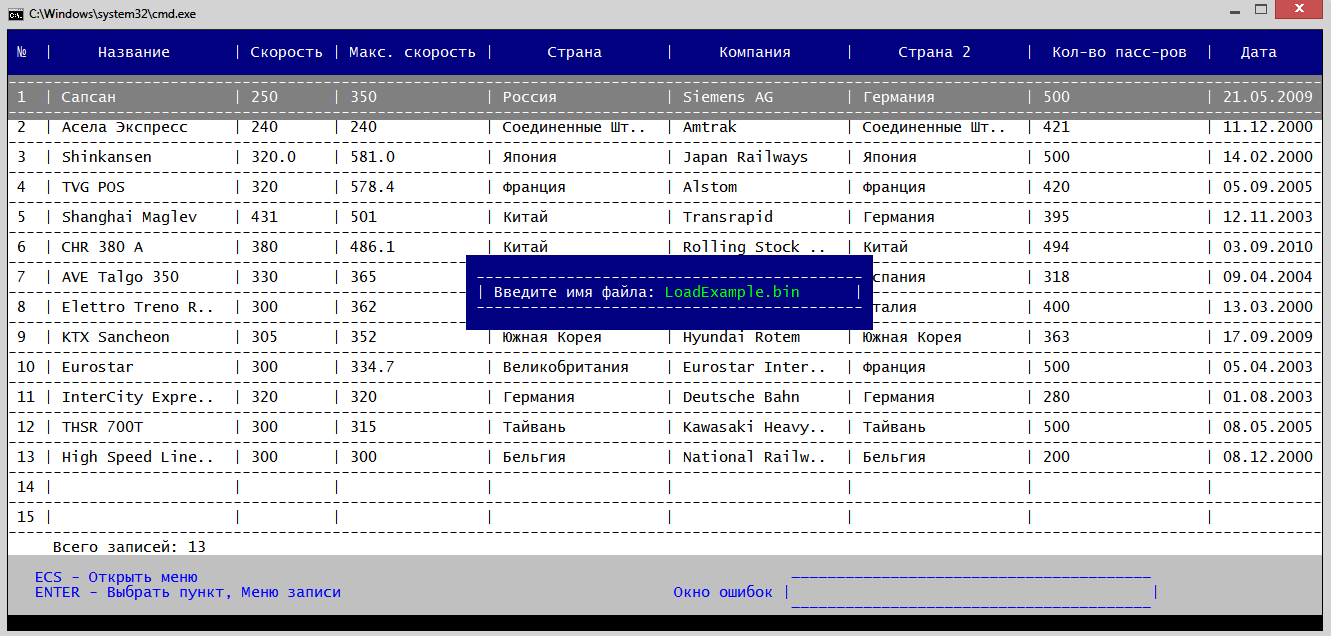


Рисунок 5.6 – Загрузка данных из файла

Выбрав пункт «Cохранить в файл», пользователю будет продложена форма для ввода имени файла.

Данное действите проиллюстрировано на рисунке 5.7.

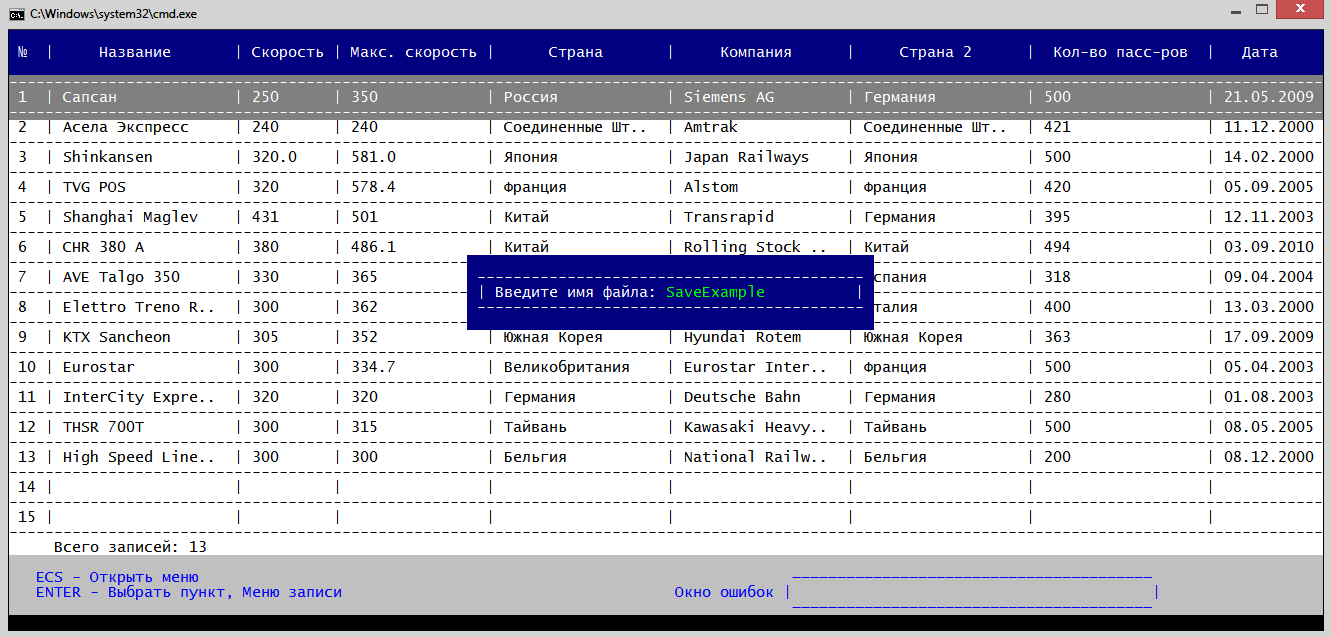


Рисунок 5.7 – Загрузка данных из файла

Изм.

Лист

№ докум.

Подпись

Дата

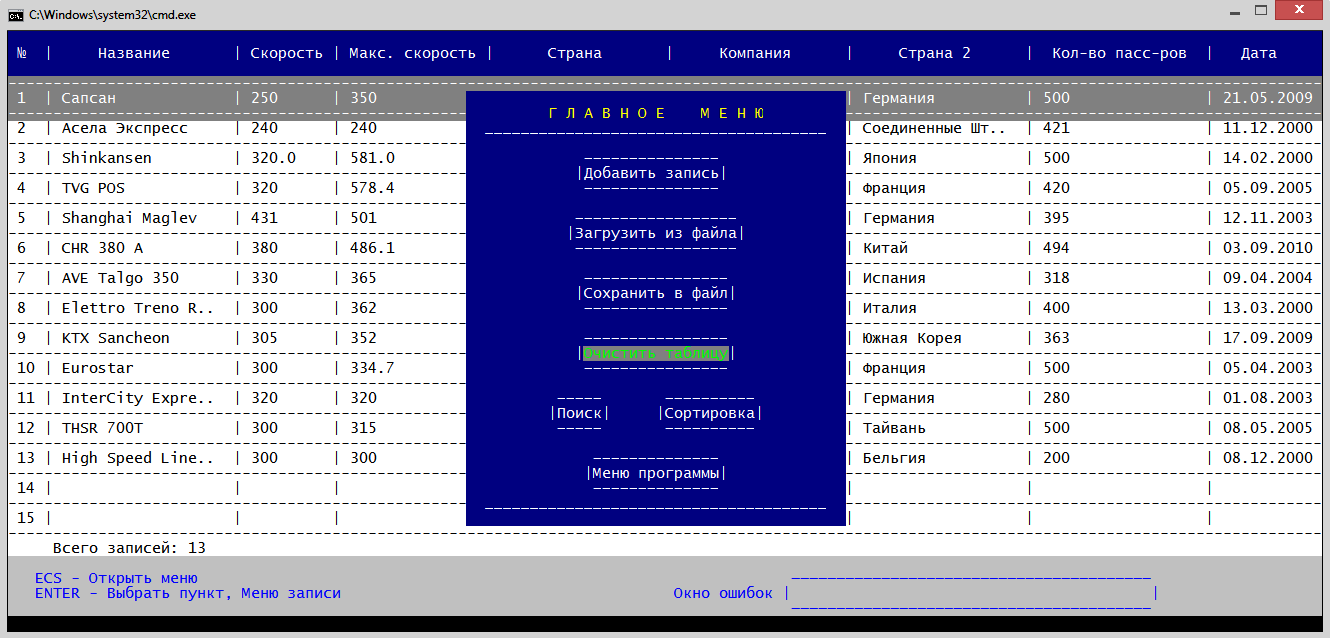
Лист

15

Курсовой проект

Выбирав пункт «Очистить таблицу», пользователь удалит все записи с данной таблицы.

Данное действите проиллюстрировано на рисунке 5.8.



Изм.

Лист

№ докум.

Подпись

Дата

Лист

16

Курсовой проект

Рисунок 5.8 – Пункт меню «Очистить таблицу»

На рисунке 5.9. проиллюстрирован результат данного действия.

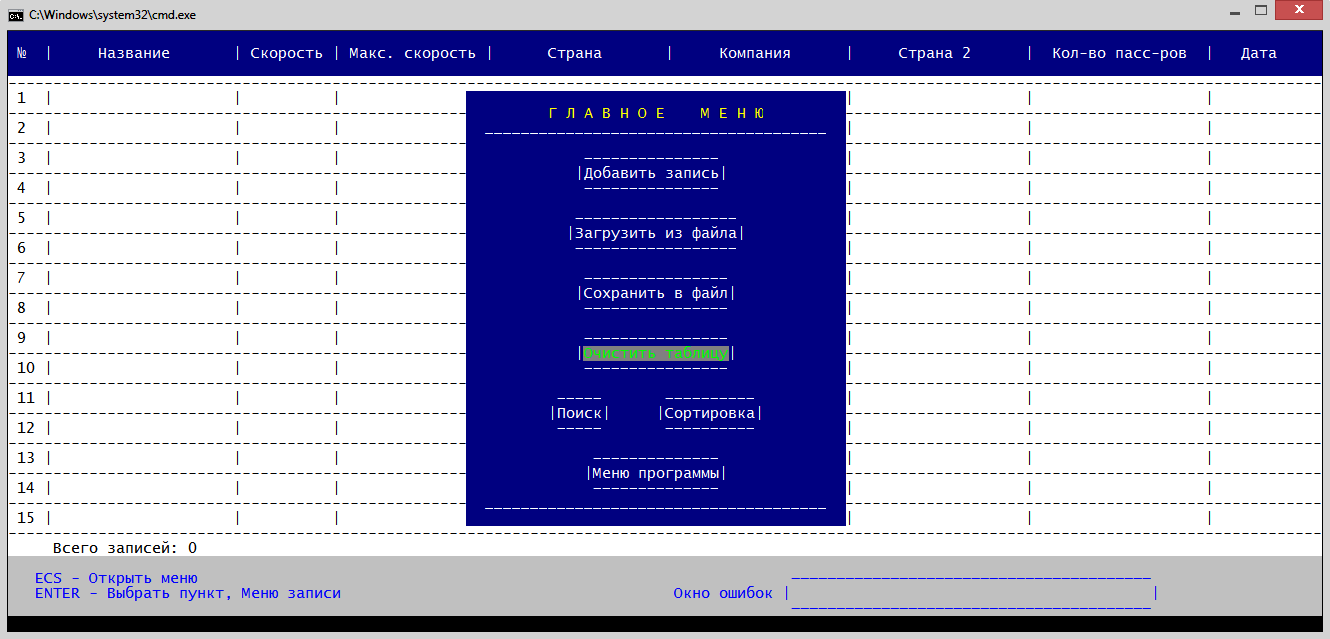


Рисунок 5.9 – Результат очистки таблицы

Выбрав пункт «Поиск», пользователю будет предложена форма для поиска записей в таблице по полному или частичному совпадению, после ввода искомого символого или слова, необходимо нажать на клавишу подтверждения ввода «Enter», в случае совпадения, программа отобразит найденные записи в таблице.

Данное действите проиллюстрировано на рисунке 5.10.

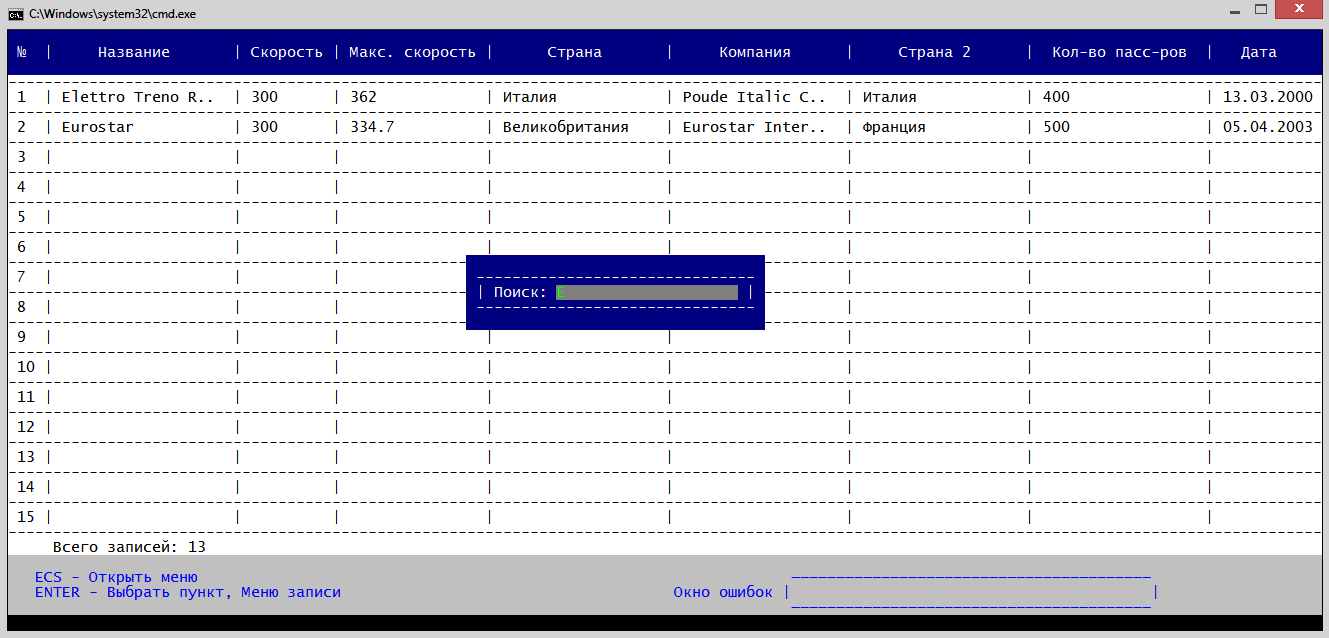


Рисунок 5.10 – Поиск элемента в таблице

Далее, нажав клавишу «ESC», программа установит курсор на первую найденную запись.

Данное действите проиллюстрировано на рисунке 5.11.

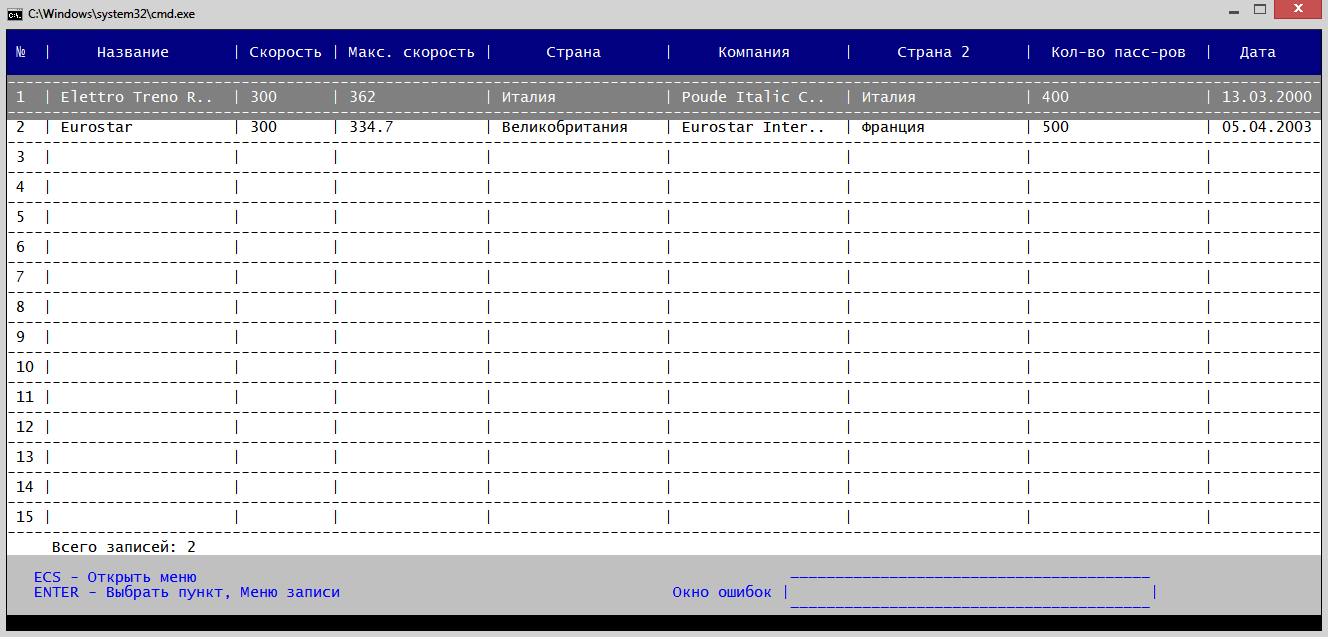


Рисунок 5.11 – Результаты поиска записей на букву «E»

Изм.

Лист

№ докум.

Подпись

Дата

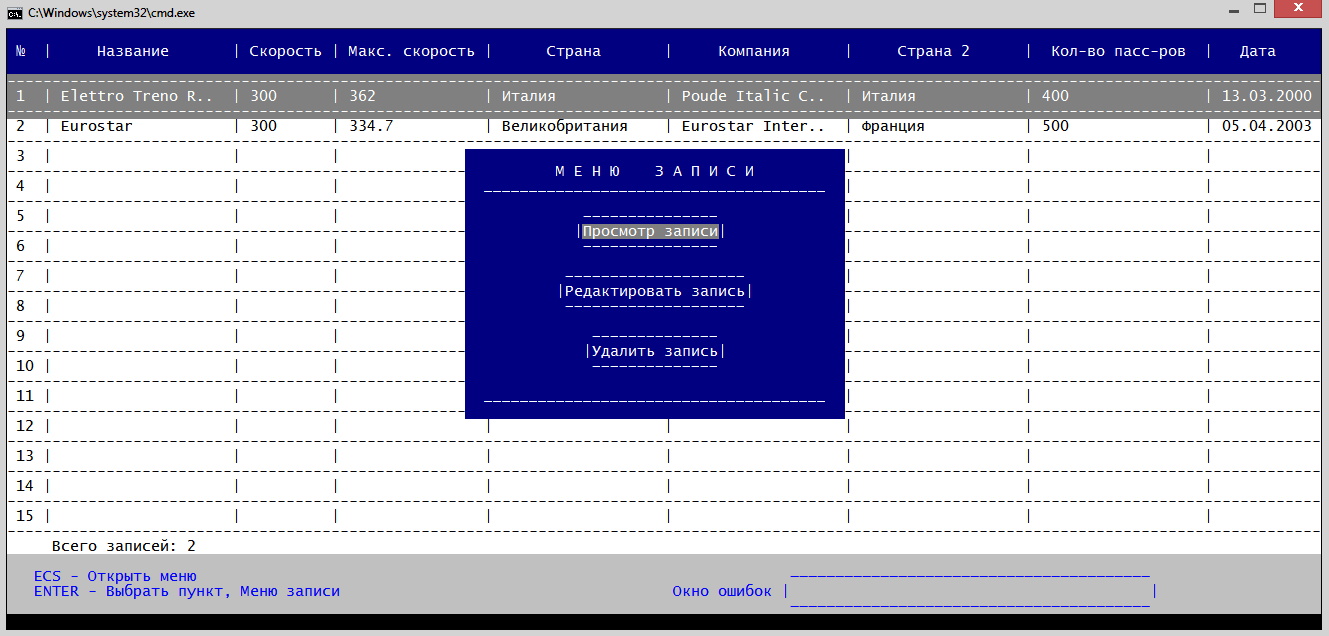
Лист

17

Курсовая проект

Далее, нажав на клавишу «Enter», программа предоставит нам меню для работы с выбранной записью.

Данное действие проиллюстрировано на рисунке 5.12.



Изм.

Лист

№ докум.

Подпись

Дата

Лист

18

Курсовой проект

Рисунок 5.12 – Меню для выбранной записи

Выбрав пункт «Просмотр записи» из текущего меню, программа покажет пользователю подробную информацию о выбранной записи.

Данное действие проиллюстрировано на рисунке 5.13.

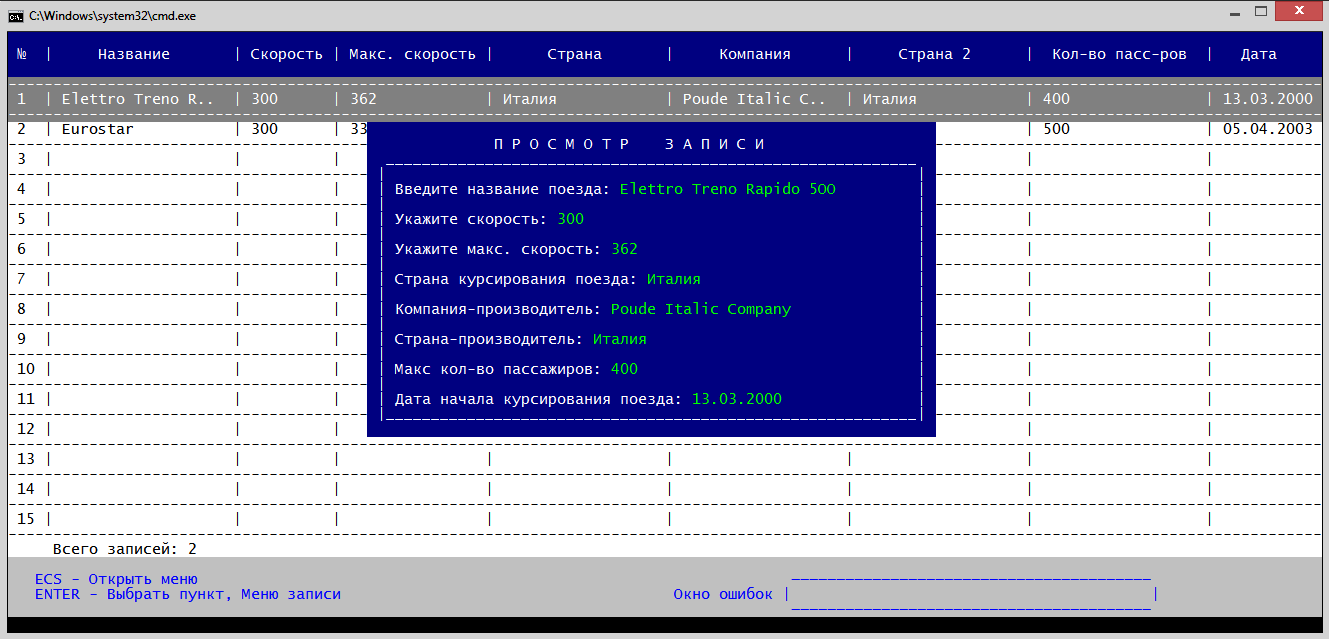


Рисунок 5.13 – Пункт меню «Просмотр записи»

Нажав клавишу «ECS2 – вы закроете данное окно и вернетесь назад к найденным записям.

Выбрав пункт «Редактировать запись» из «Меню записи», вам будет предоставлена возможность редактирования записи прямо в таблице.

Данное действие проиллюстрировано на рисунке 5.14.

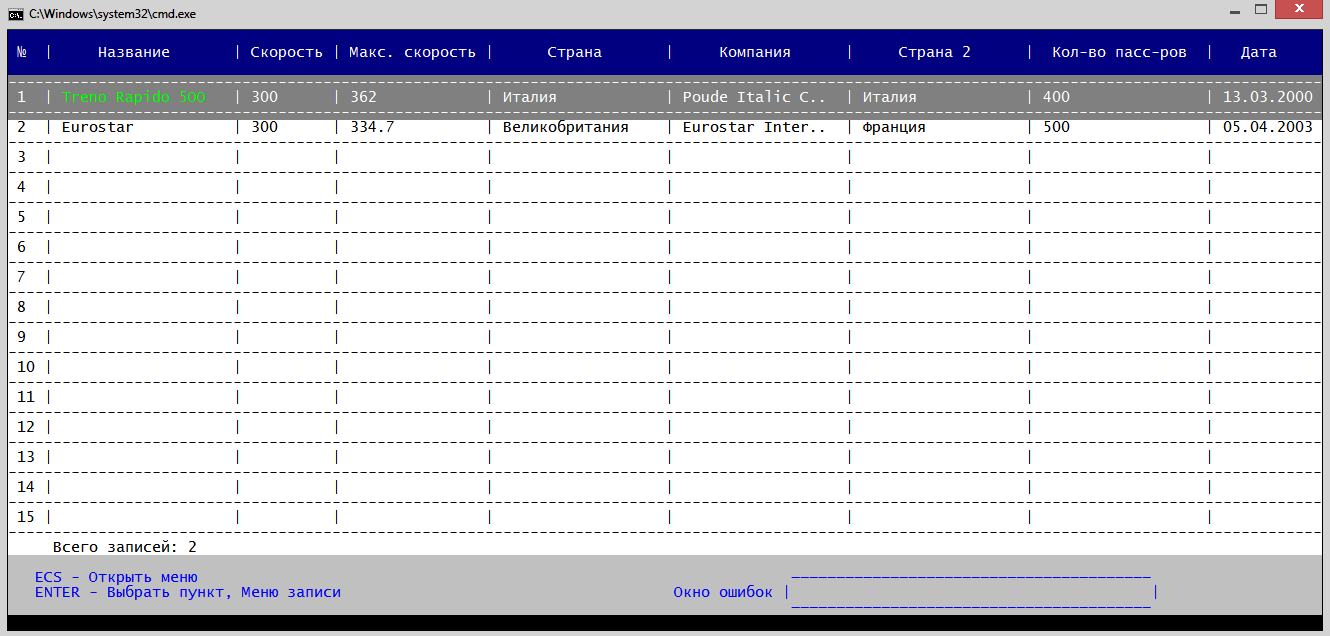


Рисунок 5.14 – Редактирование записи

Изм.

Лист

№ докум.

Подпись

Дата

Лист

19

Курсовой проект

Нажав клавишу «Enter», курсор перейдет к следующему полю, которое можно будет изменить.

Возвращаемся в основное меню для работы с таблицей.

Основное меню проиллюстрировано на рисунке 5.15.

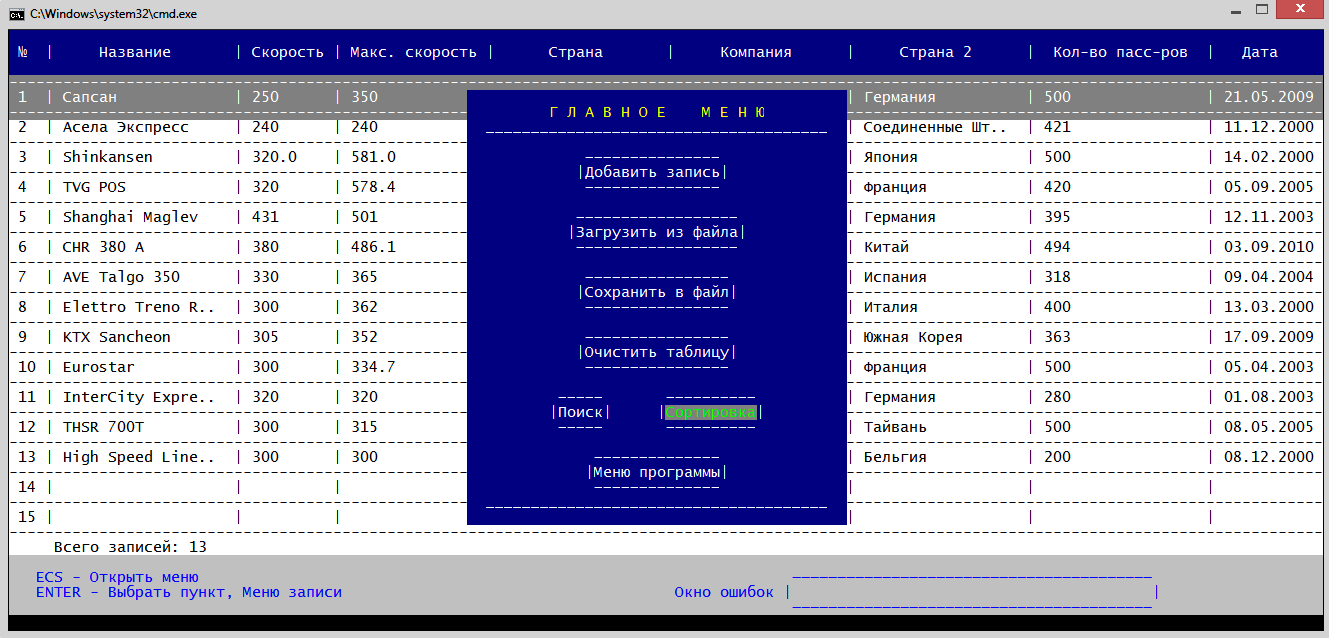
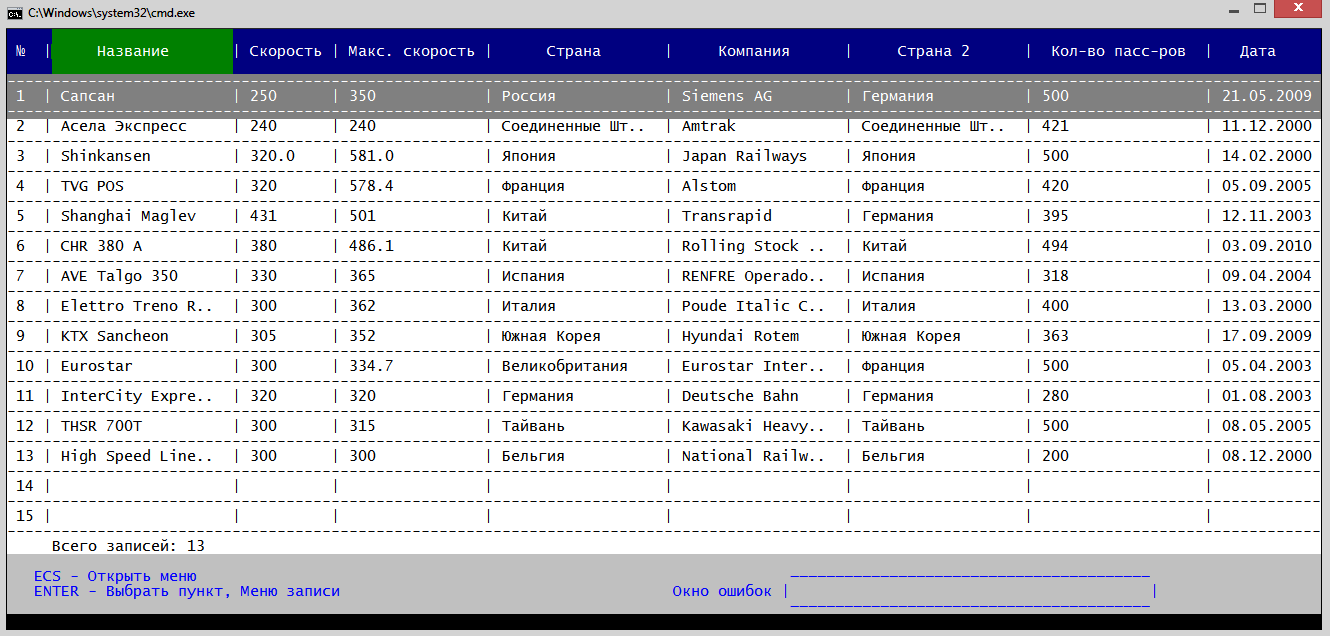


Рисунок 5.15 – Основное меню, пункт «Сортировка»

Выбирав пункт «Сортировка», курсор перейдет на заголовок поля в первом столбце таблицы, предварительно выделив его как поля, по которому может быть произведена сортировка записей.

Данное действие проиллюстрировано на рисунке 5.16.



Изм.

Лист

№ докум.

Подпись

Дата

Лист

20

Курсовой проект

Рисунок 5.16 – дополнительное меню раздела «Сортировка»

Нажав клавишу «Enter», программа выполнит сортировку записей по данному полю в порядке возрастания.

Данное действите проиллюстрированно на рисунке 5.17.

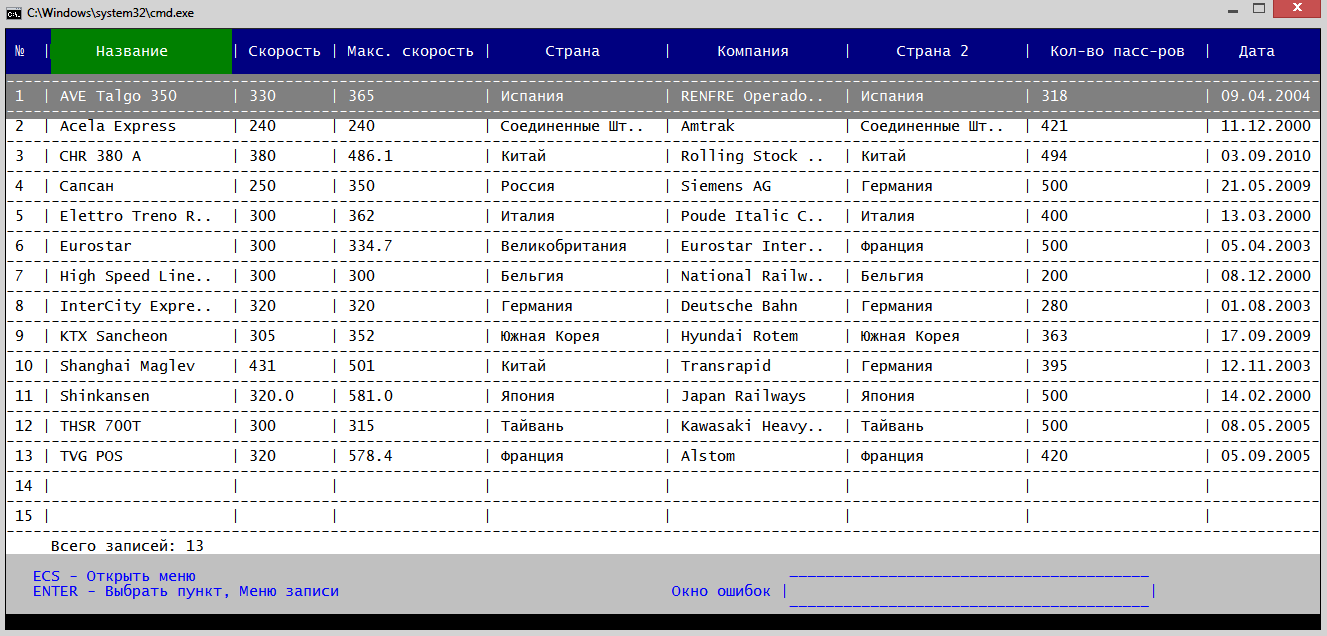
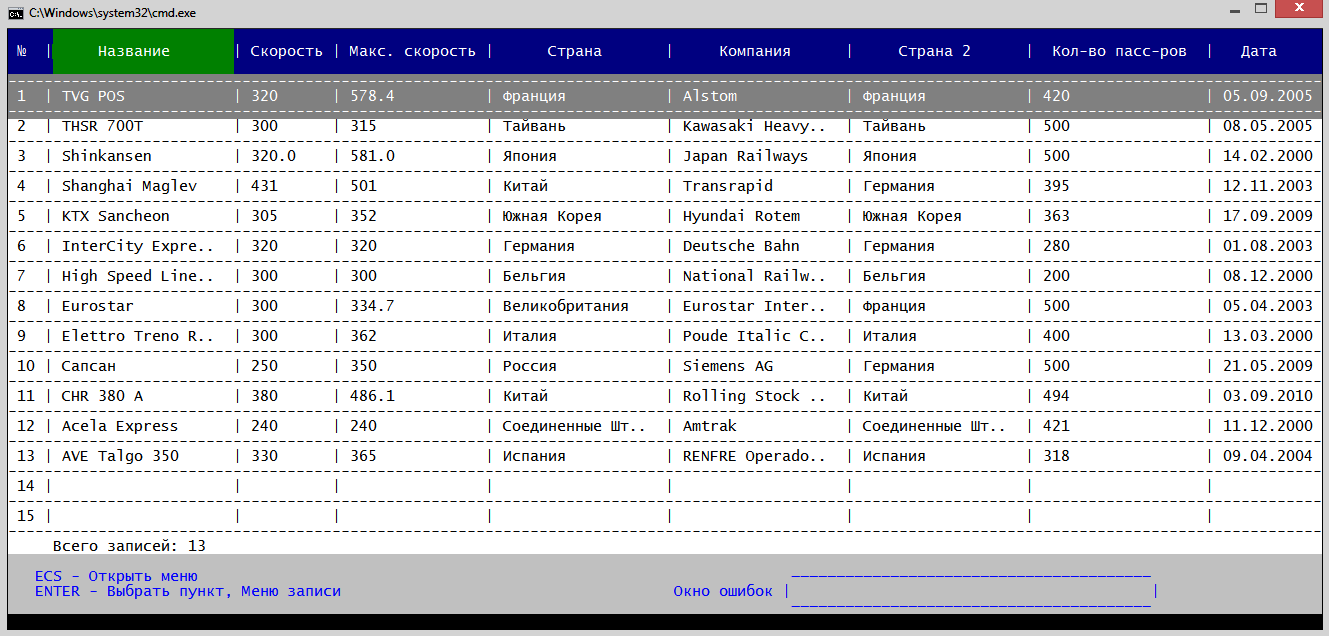


Рисунок 5.17 – Сортировка записей по названию в порядке возрастания

Если после выполнения сортировки по возрастанию пользователь нажмет еще раз на клавишу «Enter», то будет произведена сортировка по убыванию.

Данное действите проиллюстрировано на рисунке 5.18.



Изм.

Лист

№ докум.

Подпись

Дата

Лист

21

Курсовой проект

Рисунок 5.18 – Сортировка записей по названию в порядке убывания

Навигация по заголовкам полей таблицы осуществляется с помощью клавиш стрелок, выбор сортировка по тому или иному полю осуществляется через клавишу «Enter». Признанок завершения работы с сортировкой таблицы служит нажатая клавиша «ESC».

# ЗАКЛЮЧЕНИЕ

Изм.

Лист

№ докум.

Подпись

Дата

Лист

22

Курсовой проект

В данном курсовом проекте была поставлена задача, которая заключается в написании программы по работе с данными.

Под работой понимается ряд функций, которые программа в обязательном порядке должна выполнять.

Еще раз напомним предъявленные требования к данному программе:

* Добавление новых записей
* Удаление записей
* Просмотр отдельно взятой записи.
* Просмотр всех записей в таблице
* Редактирование выбранной записи
* Поиск записей в таблице
* Сортировка записей по любому полю и в любом направлении
* Сохранение результатов работы в файл
* Загрузка сохраненных ранее результатов из файла
* Вывод всех записей на печать в виде текстового файла

В результате реализации и тестирования разработанной нами программы, все выше описанные функции полностью работоспособны.

Также большое внимание было уделено и интерфейсу программы, который обеспечивает удобное и понятное взаимодействие с приложением.

Без внимания не осталась обработка исключений, ошибок, которые могут возникнуть при работе с данной программой.

В результате всей проделанной работы был приобретен бесценный опыт по разработке объемного и более сложного проекта.

Полученный нами опыт несомненно найдет свое применение в будущем при разработке более сложных программ.

**СПИСОК ЛИТЕРАТУРЫ**

Изм.

Лист

№ докум.

Подпись

Дата

Лист

23

Курсовой проект

1. Лафоре Р. Объектно-ориентированное программирование в С++ (4-е издание, 2004).
2. Конспект лекций по дисциплине «Основы алгоритмизации и программирование».
3. Конспект лекций по дисциплине «Языки программирования».

**ПРИЛОЖЕНИЕ А**

Изм.

Лист

№ докум.

Подпись

Дата

Лист

24

Курсовой проект

начало

file\_name

Введите имя файла

Ввод file\_name

File

File.open(file\_name)

нет

да

File == TRUE

K

File.read

massive

Файл не найден

i = k

--i >=0

Element

element->data = massive[i]

element->next = deque->Start->next

element->prev = deque->Start

deque->Start->next->prev = element

deque->Start->next = element

deque->size++

delete[] massive

конец

Рисунок А.1 – Блок-схема функции Load\_Deque

начало

file\_name

changed\_name

Введите имя файла

Ввод file\_name

ptr = deque->start->next

massive

k\_for\_mas = 0

Ptr != Deque->End

train = ptr->data

ptr = ptr->next

massive[k\_for\_mas++] = train

strcat(changed\_name, “.bin”

bin\_file

bin\_file.write

bin\_file.close

ptr = deque->Start->next

strcat(file\_name, “.txt”)

file(file\_name)

ptr != Deque->End

train

train = ptr->data

ptr = ptr->next

Вывод в файл train.name,

train.speed

train.max\_speed

train.county

train.company

train.by\_country

train.pass\_count

train.date

file.close

delete[] massive

конец

Изм.

Лист

№ докум.

Подпись

Дата

Лист

25

Курсовой проект

Рисунок A.2 – Блок-схема функции Save\_Deque

**ПРИЛОЖЕНИЕ Б**

Изм.

Лист

№ докум.

Подпись

Дата

Лист

26

Курсовой проект

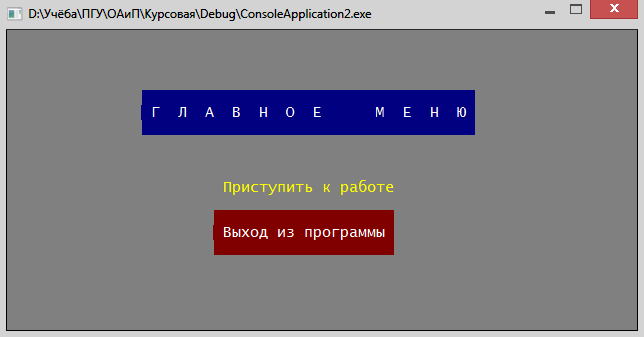


Рисунок Б.1 – Главное меню программы

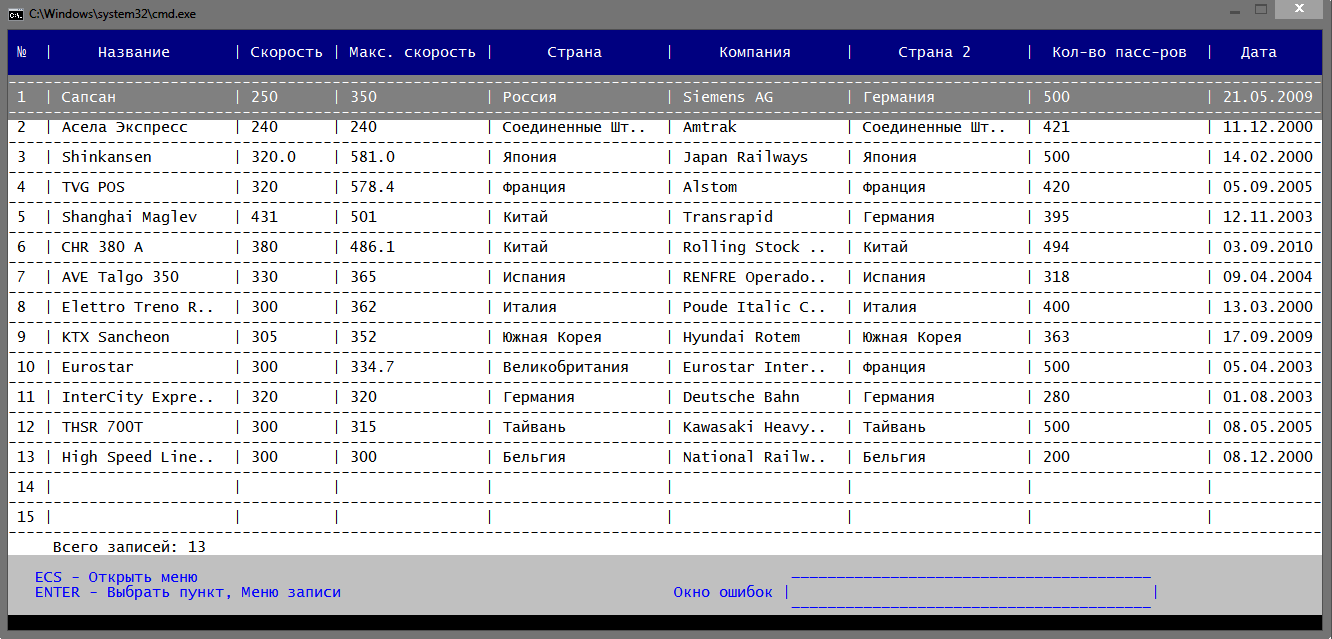


Рисунок Б.2 – Таблица с данными

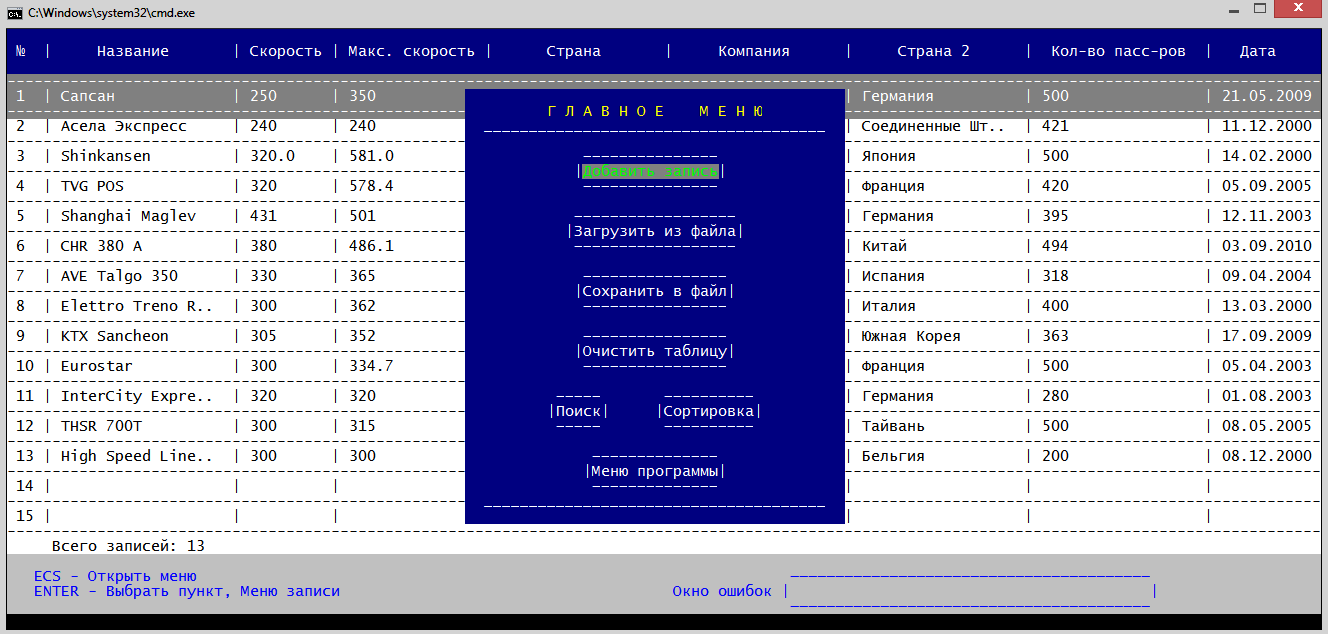


Рисунок Б.3 – Главное меню таблицы

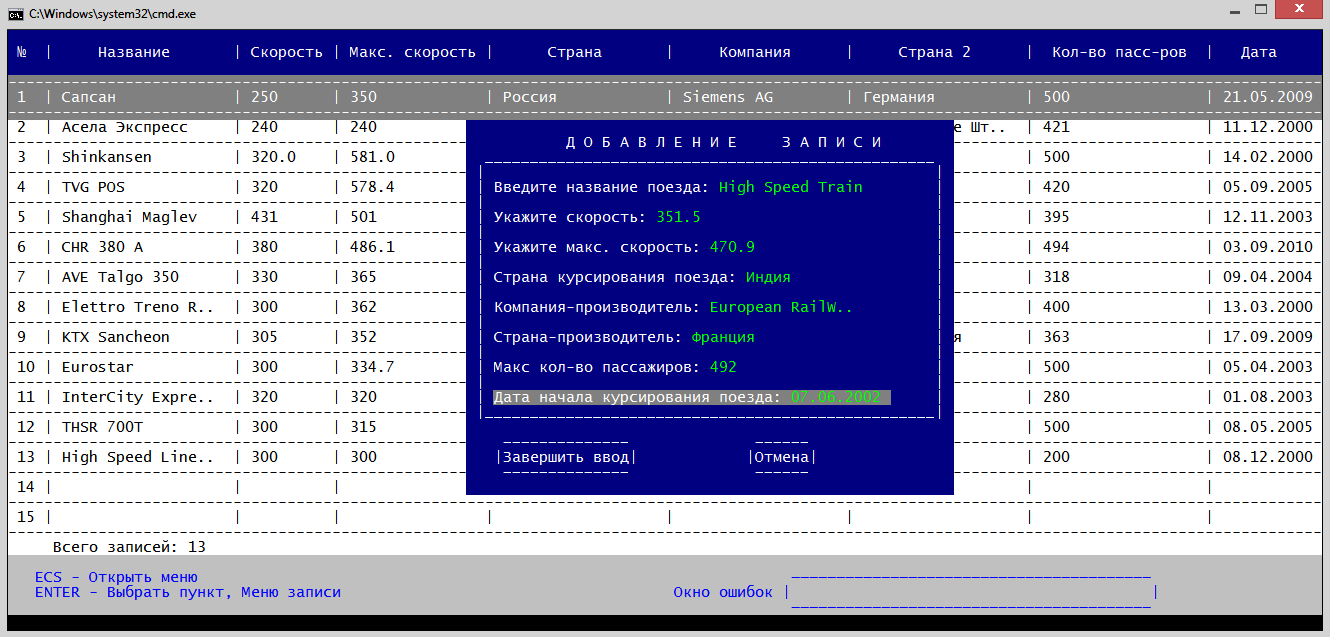


Рисунок Б.4 – Добавление записи

Изм.

Лист

№ докум.

Подпись

Дата

Лист

27

Курсовой проект

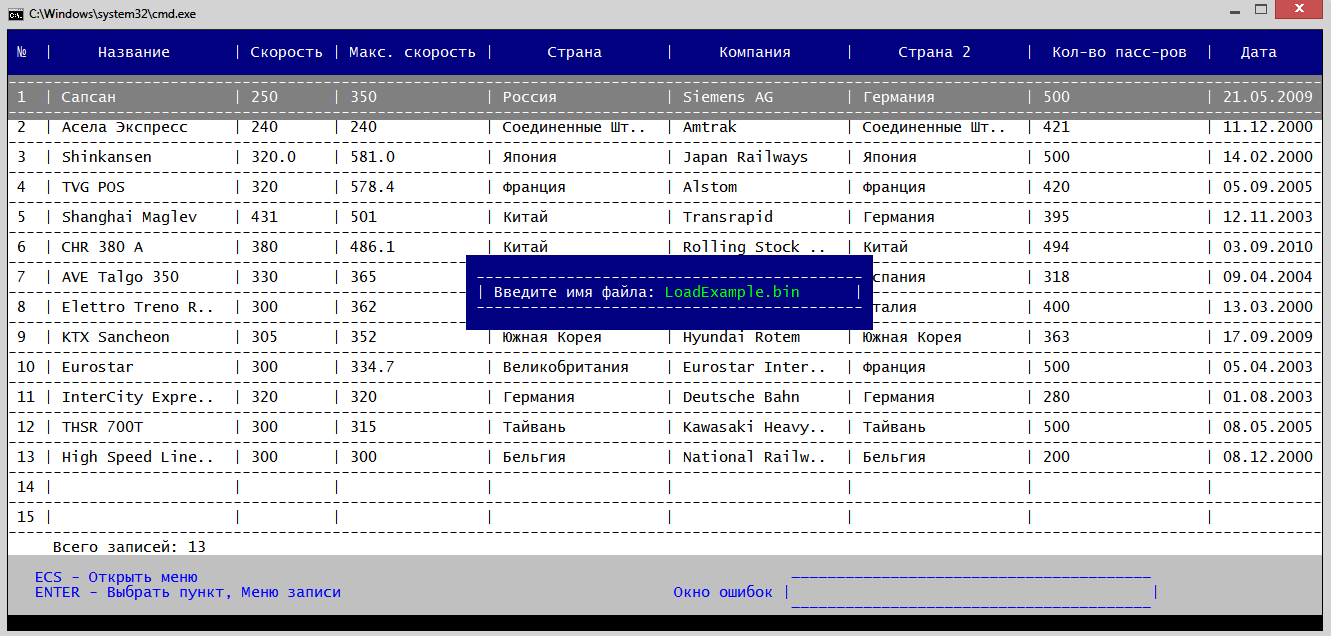


Рисунок Б.5 – Загрузка данных из файла

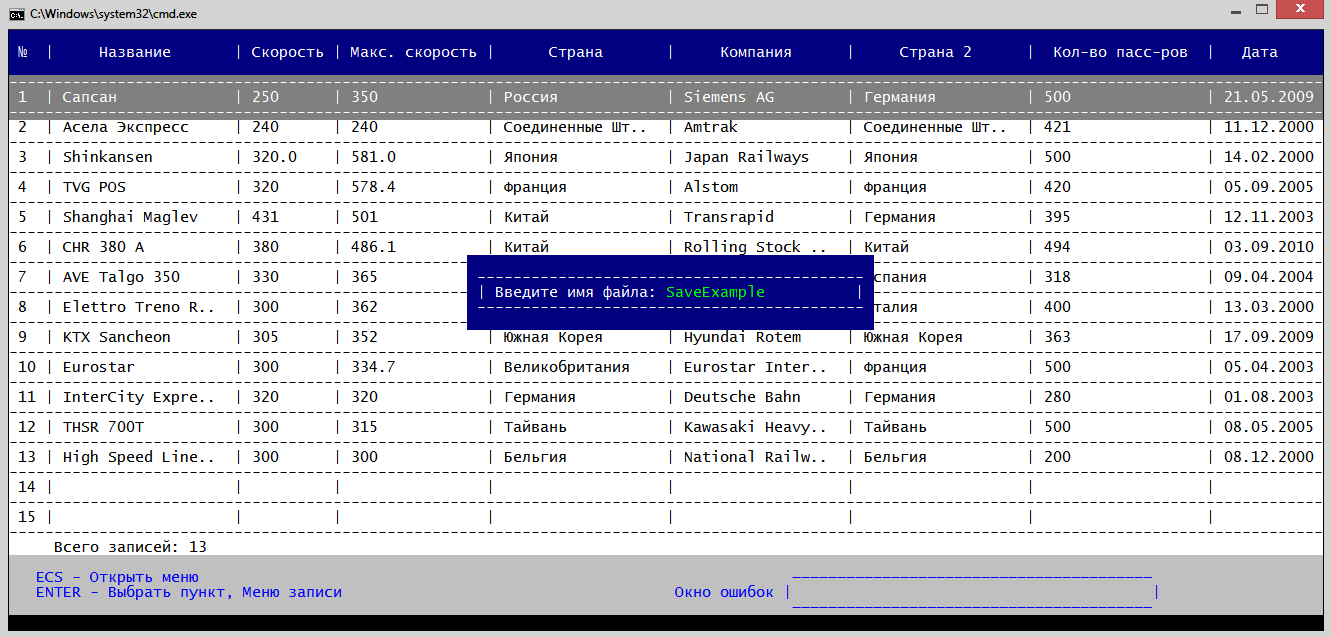


Рисунок В.6 – Загрузка данных из файла

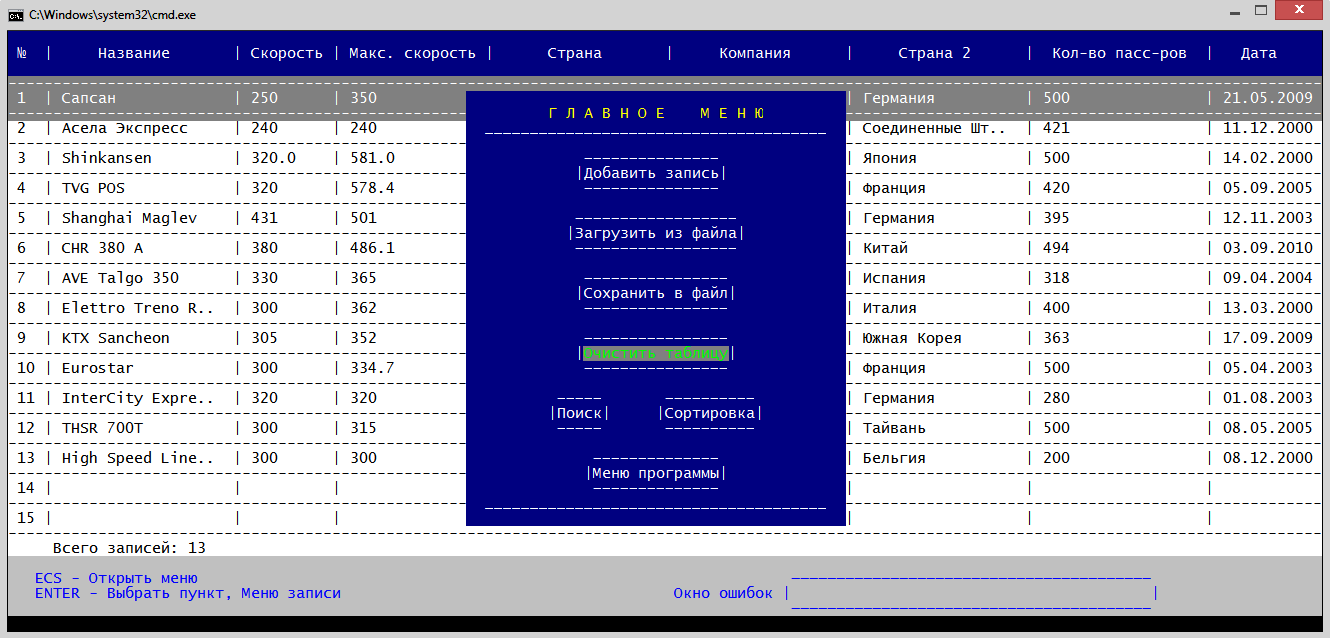
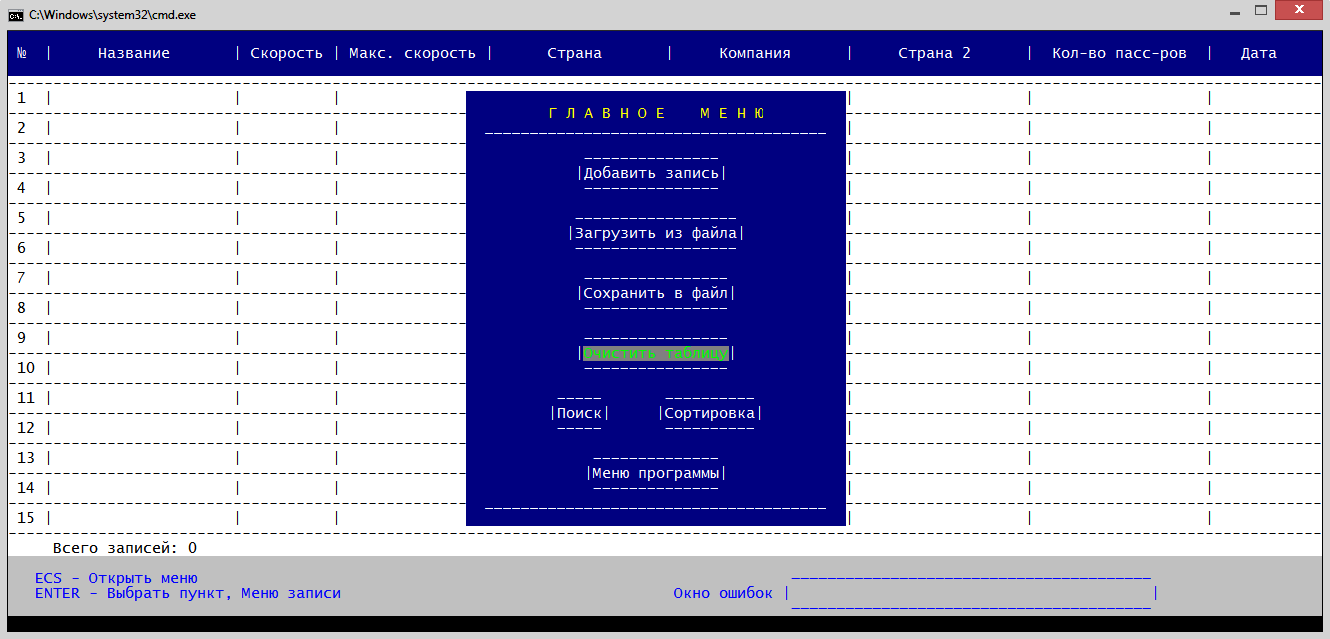


Рисунок B.7 – Пункт меню «Очистить таблицу»



Изм.

Лист

№ докум.

Подпись

Дата

Лист

28

Курсовой проект

Рисунок Б.8 – Результат очистки таблицы

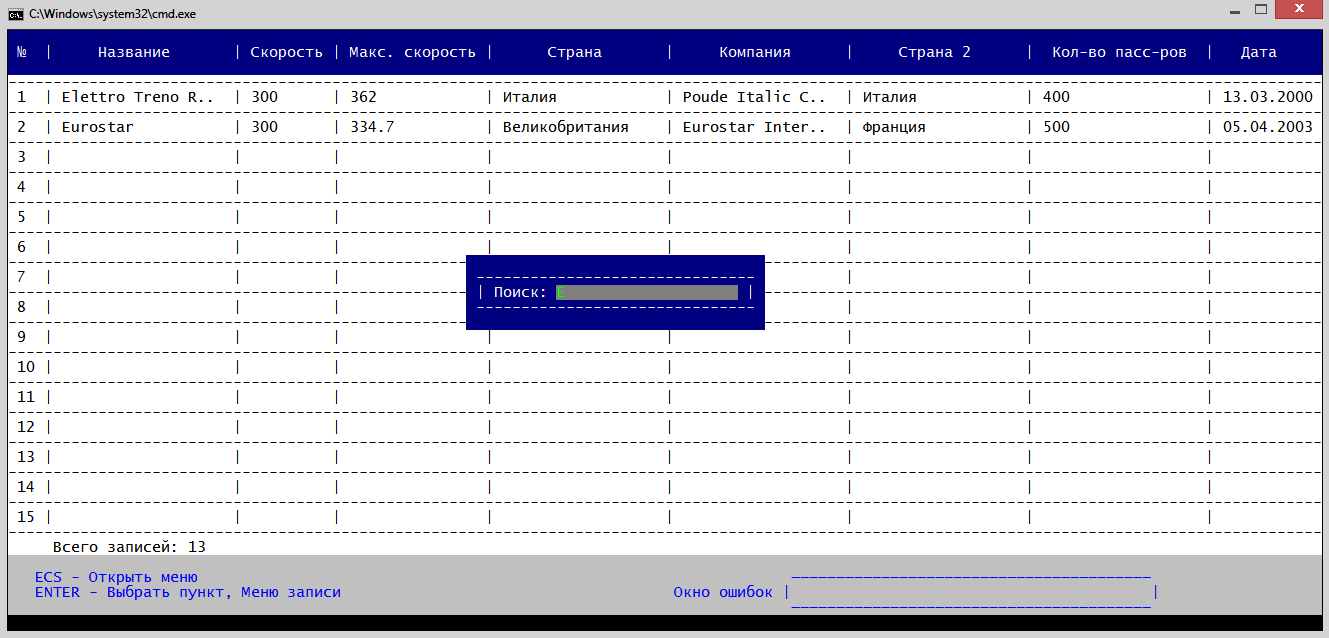


Рисунок Б.9 – Поиск элемента в таблице

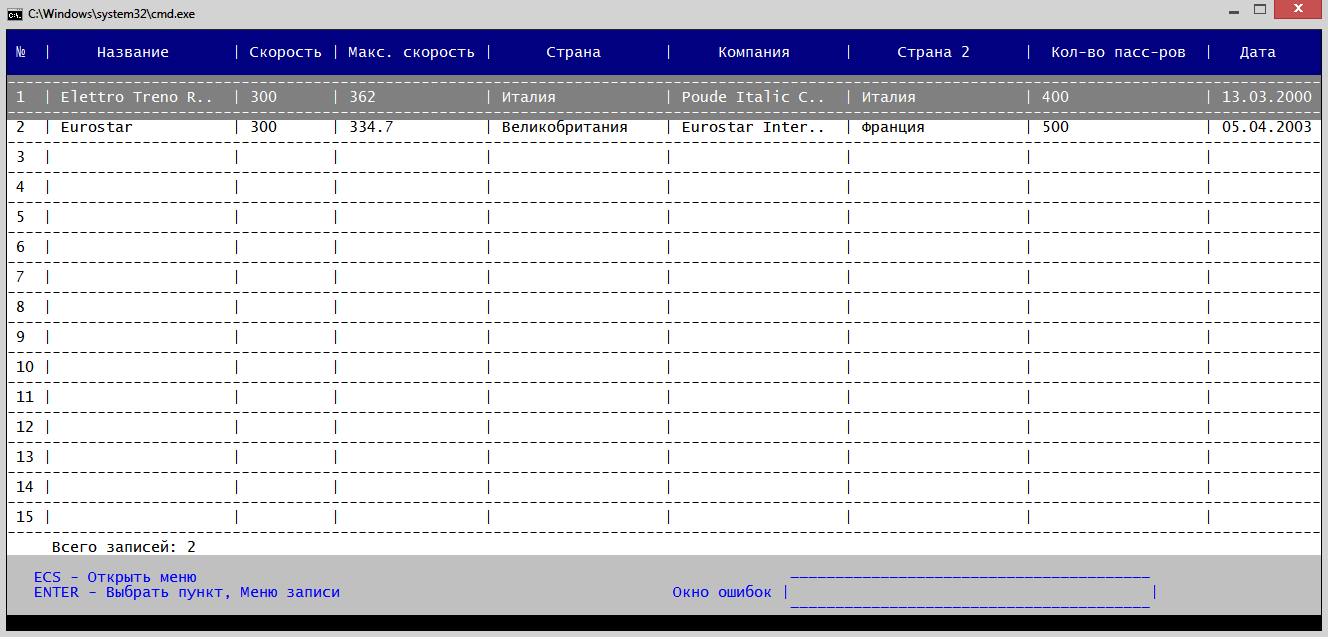


Рисунок Б.10 – Результаты поиска записей на букву «E»

Изм.

Лист

№ докум.

Подпись

Дата

Лист

29

Курсовая проект

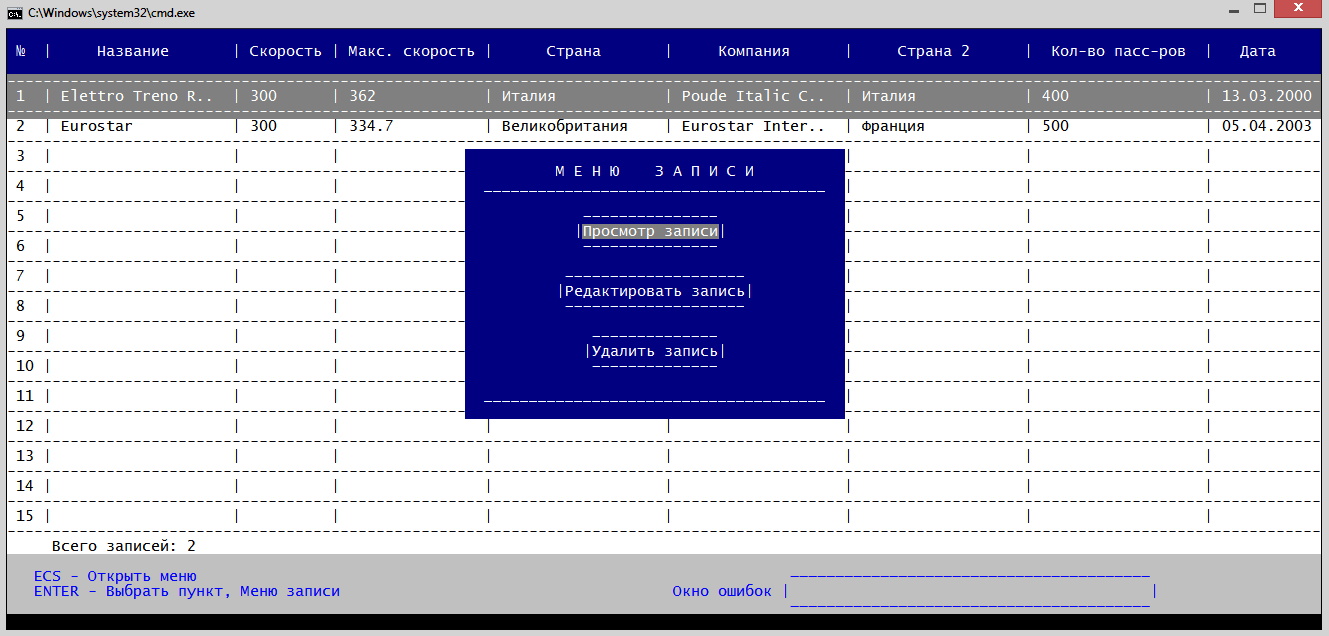


Рисунок Б.11 – Меню для выбранной записи

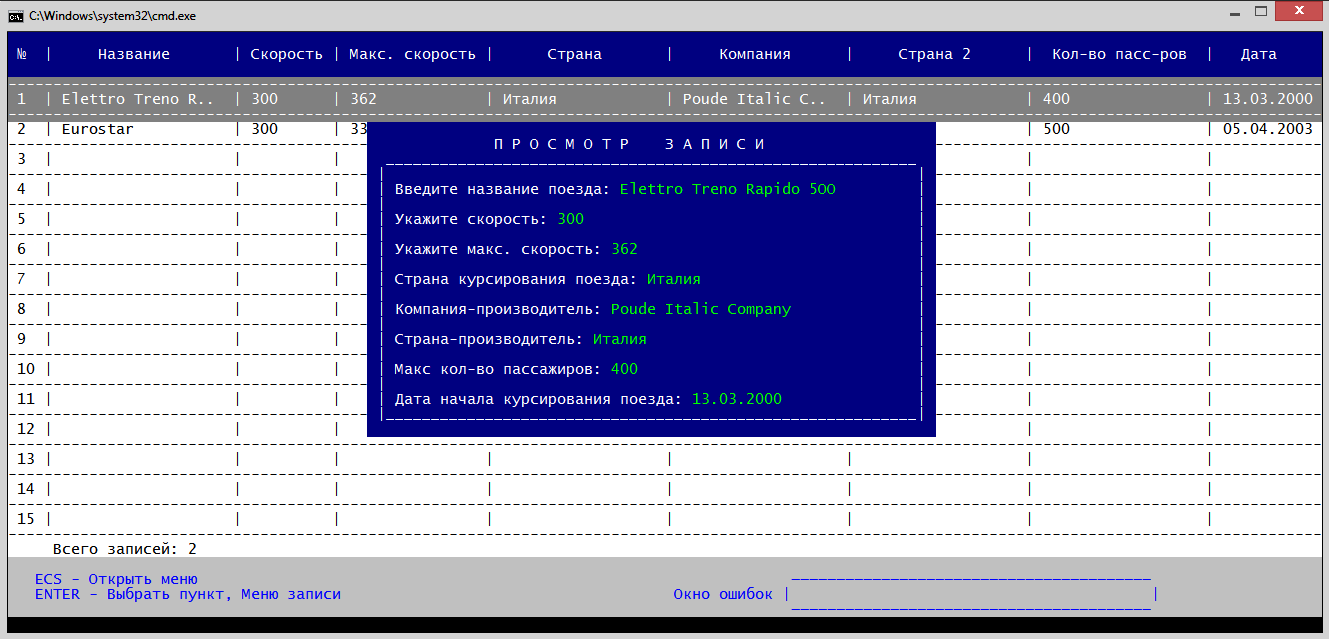


Рисунок Б.12 – Пункт меню «Просмотр записи»

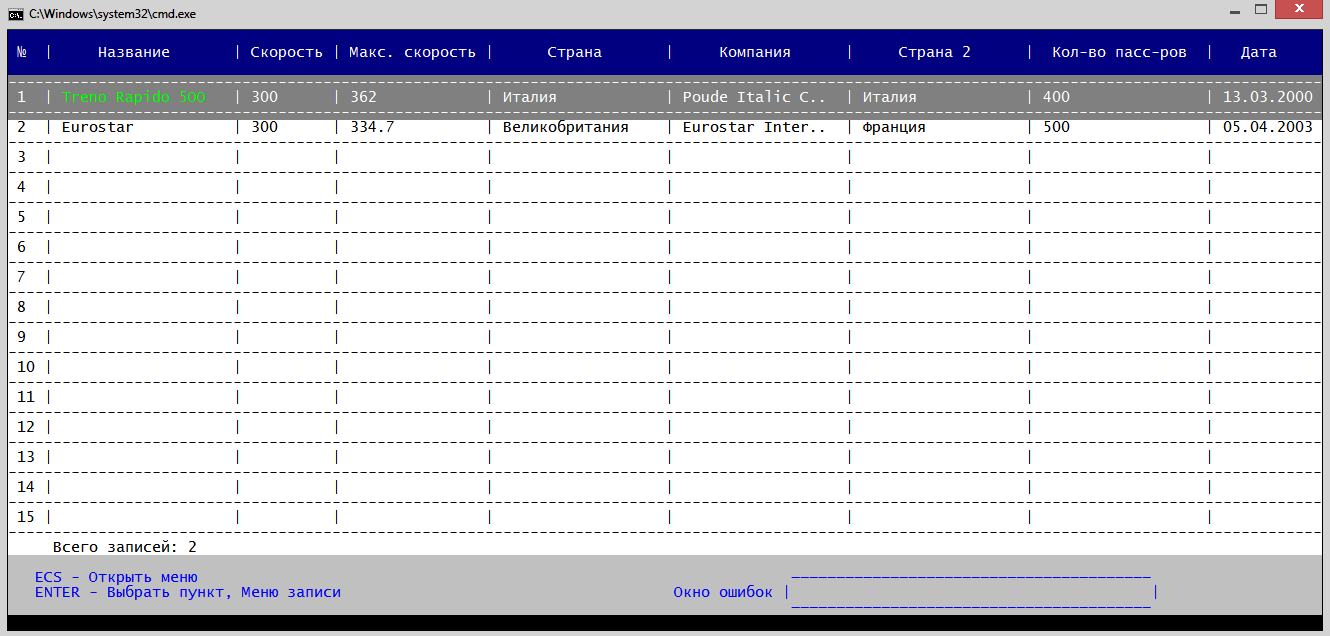


Рисунок Б.12 – Редактирование записи

Изм.

Лист

№ докум.

Подпись

Дата

Лист

30

Курсовой проект

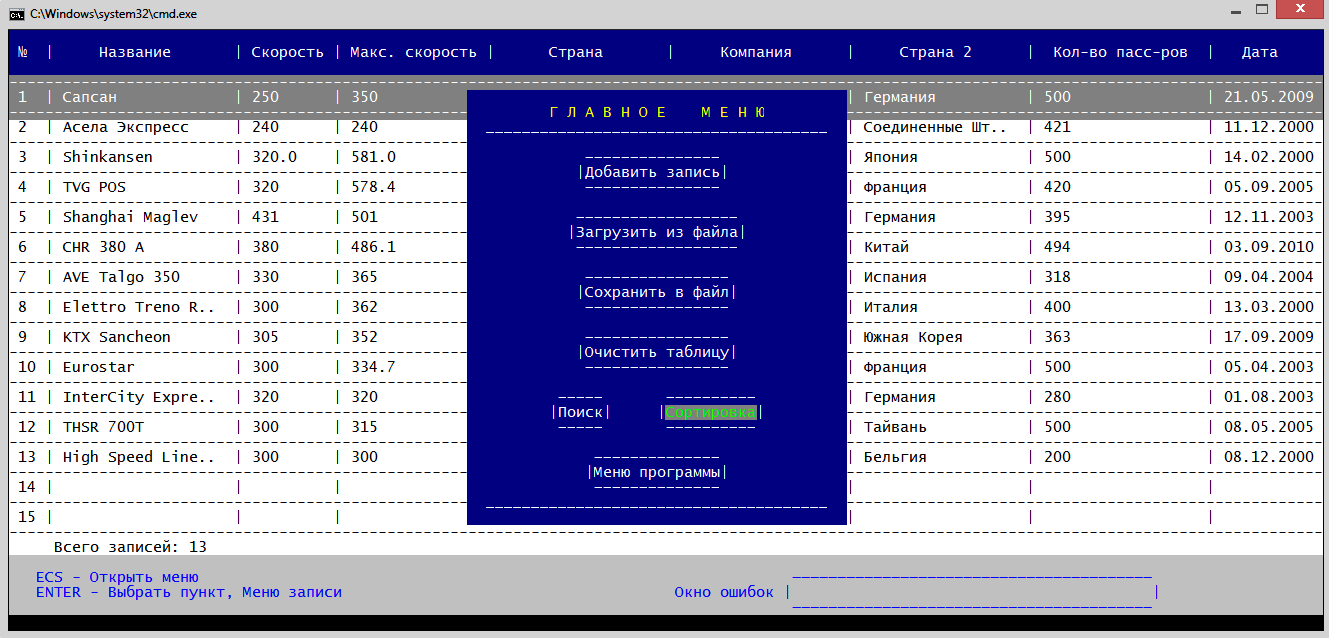


Рисунок Б.13 – Основное меню, пункт «Сортировка»

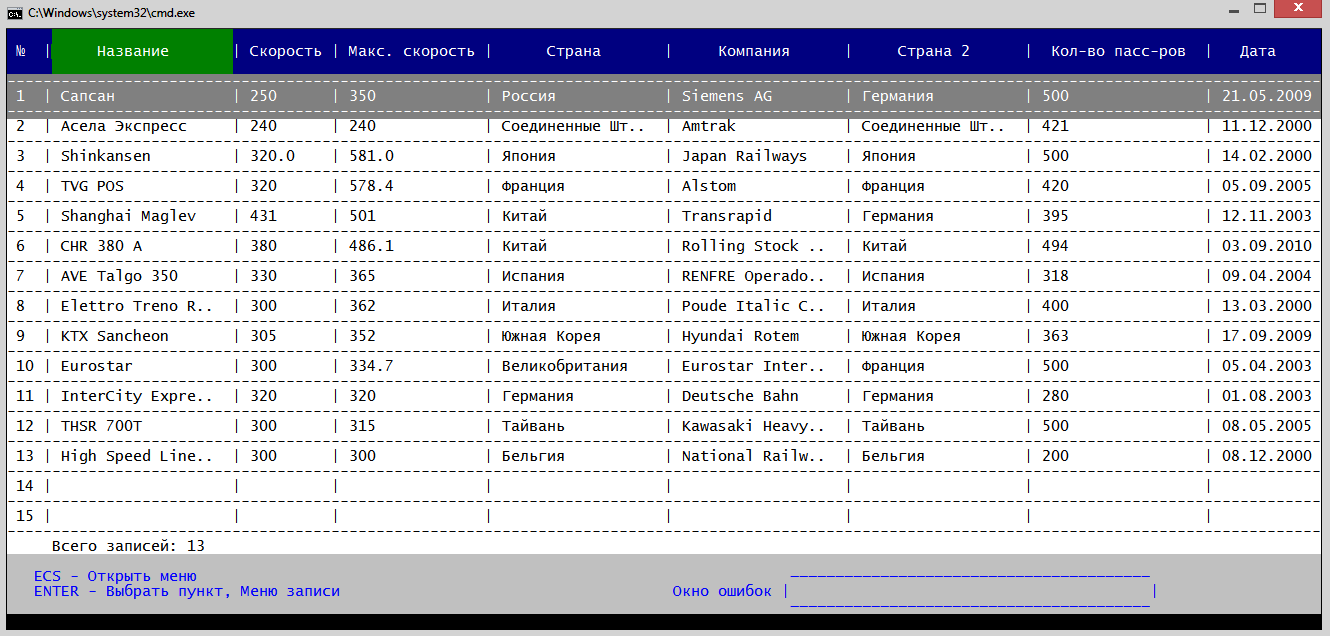


Рисунок Б.14 – Дополнительное меню раздела «Сортировка»

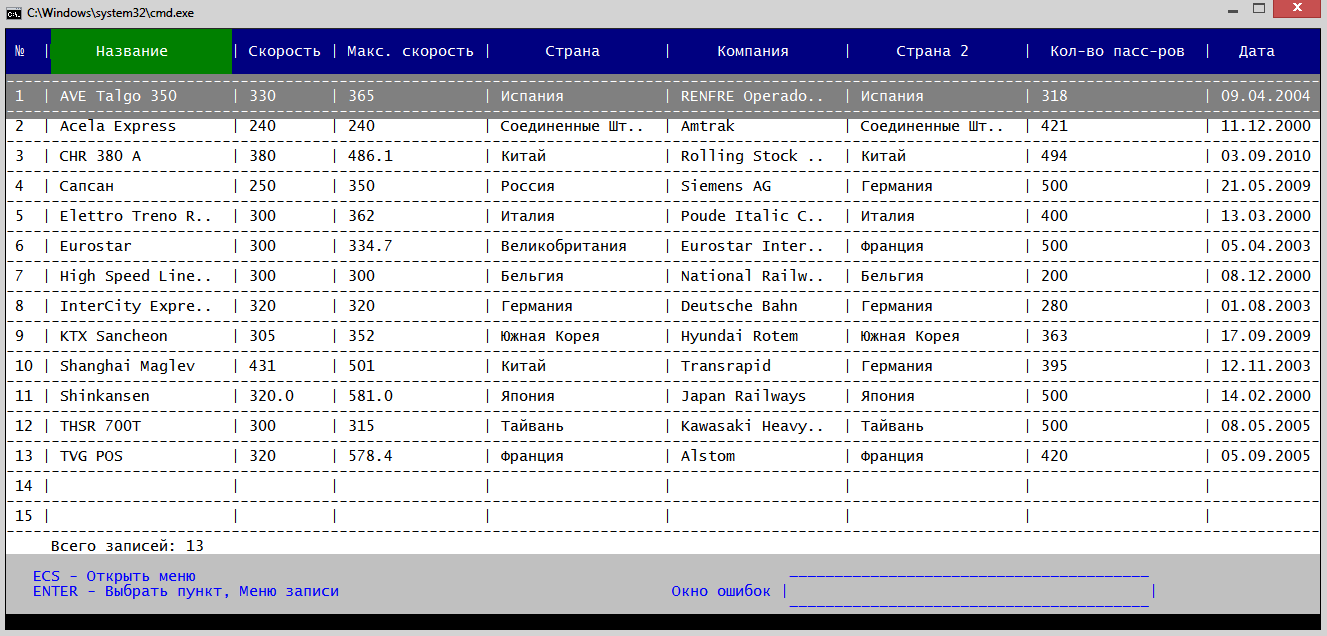
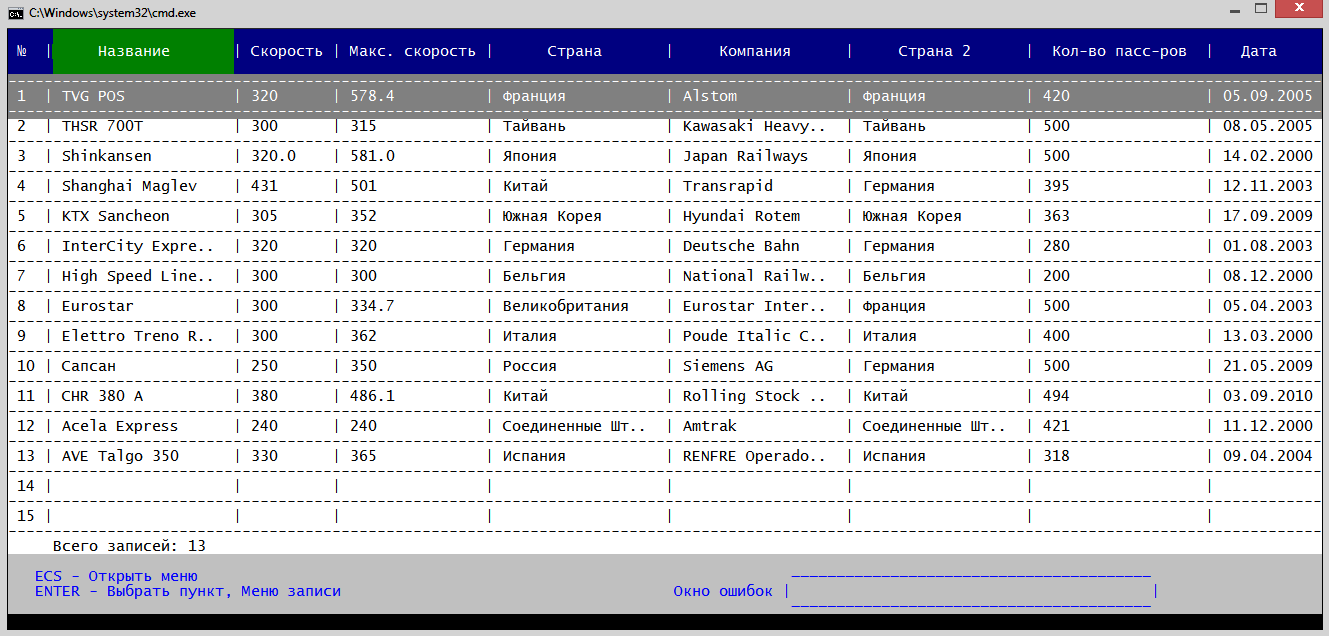


Рисунок Б.15 – Сортировка записей по названию в порядке возрастания



Изм.

Лист

№ докум.

Подпись

Дата

Лист

31

Курсовой проект

Рисунок Б.16 – Сортировка записей по названию в порядке убывания

**ПРИЛОЖЕНИЕ В**

Готовая программа, исходные коды и записка к данному курсовому проекту находятся на CD диске.

Изм.

Лист

№ докум.

Подпись

Дата

Лист

32

Курсовая проект