МАЗМҰНЫ

КІРІСПЕ....................................................................................................................2

1. С++ ТУРАЛЫ ЖАЛПЫ ТҮСІНІК....................................................................3

1.1.С++ ТІЛІНІҢ ШЫҒУ ТАРИХЫ.......................................................................4

2.НЕГІЗГІ БӨЛІМ....................................................................................................5

2.1.ФУНКЦИЯ.........................................................................................................5

2.2.ФАЙЛ.................................................................................................................6

2.3.ШАРТТЫ ОПЕРАТОРЛАР..............................................................................6

2.4.ЦИКЛДАР..........................................................................................................7

2.5.ФУНЦИЯЛАРДЫ ТҮСІНДІРУ..................................................................9-10

ҚОРЫТЫНДЫ.......................................................................................................11

ҚОЛДАНЫЛҒАН ӘДЕБИЕТТЕР………………………………………............12

БАҒДАРЛАМА БӨЛІМІ.................................................................................13-22

КІРІСПЕ

Курстық жұмыс бойынша “Автосалон” атты деректер базасын құру керек. Еңгізілген деректер деректер базасында сақталуы керек. Қолданушыға деректер базасымен ыңғайлы жұмыс жасауын қамтамасыз ету үшін бағдарламаға меню қосамыз. Меню арқылы қолданушыныңжұмысын жеңілдетуге болады. Бұл бағдарламада деректермен осындай операциялар қарастырылады:

0.Бағдарламадан шығy.

1.Ақпаратты еңгізy.

2.Ақпаратты шығарy.

3.Ақпараттыжаңартy.

4.Ақпаратты өшірy.

5.Ақпаратты қосy.

“Aвтocaлoн” деректер базасында осындай айнымалылар қолданылады:

Көлік брeнды – жoлдық тbп string

Көлік моделі – жолдық тип string

Көліктің шыққан жылы – бүтін тип int

Көліктің бағасы – бүтін тип int

Көліктің түсі – жолдық тип string

Көлік қозғалтқышының көлемі – нақты тип float

Data input – командасы көмегімен біз деректер қорына ақпараттарды еңгізе аламыз.

Data output – деректер қорына еңгізілген ақпаратты экранға шығару.

Data change – деректер қорына еңгізілген ақпаратты өзгерту.

Data deletion - деректер қорына еңгізілген ақпаратты өшіру.

Adding data - деректер қорына қосымша ақпаратты қосу.

Exit– командасы арқылы бағдарламадан шығамыз.

С++ ТУРАЛЫ ЖАЛПЫ ТҮСІНІК

C ++ - бұл құрастырылған, статикалық терілген жалпы мақсаттағы бағдарламалау тілі. Ол процедуралық бағдарламалау, объектіге бағытталған бағдарламалау, жалпыланған бағдарламалау сияқты бағдарламалау парадигмаларын қолдайды. Тілде кең таралған контейнерлер мен алгоритмдер, енгізу-шығару, көп ағынды қолдау және басқа да мүмкіндіктер бар бай стандартты кітапхана бар. C ++ жоғары және төмен деңгейлі тілдердің қасиеттерін біріктіреді. Алдыңғы С тілімен салыстырғанда, объектіге бағытталған және жалпыланған бағдарламалауды қолдауға көп көңіл бөлінген.

C ++ бағдарламалық жасақтаманы әзірлеу үшін кеңінен қолданылады, ол ең танымал бағдарламалау тілдерінің бірі болып табылады.Оны қолдану аясына операциялық жүйелер, әртүрлі қолданбалы бағдарламалар, құрылғы драйверлері, ендірілген жүйелерге арналған қосымшалар, өнімділігі жоғары серверлер, сонымен қатар ойындар кіреді. Ақысыз және коммерциялық, сонымен қатар әртүрлі платформаларға арналған C ++ тілінің көптеген нұсқалары бар. Мысалы, x86 платформасында бұлар GCC, Visual C ++, Intel C ++ Compiler, Embarcadero (Borland) C ++ Builder және басқалар. C ++ басқа бағдарламалау тілдеріне, ең алдымен Java және C #, қатты әсер етті

C ++ синтаксисі С тілінен мұра болып табылады.Дизайн принциптерінің бірі С-мен үйлесімділікті сақтау болды.

Жалпы С тілінде жазылған барлық бағдарламалар С++ бағдарламалау тілінің компиляторлары арқылы өңдеуге болады. Компилятор – үш процессорлардан тұратын және олардың әрқайсысы жеке болатын тәуелсіз бағдарламалар.Оларға процессор,объекті кодтың генераторы, С++ бағдарламалау тілінің компиляторының алдыңғы жобасы жатады.

С++ бағдарламалау тілінде жазылған кез-келген бағдарламалар үш жағдайда болуы мүмкін:

1.Бастапқы файлдар(\*.c; \*.c++; \*.cpp)

2.Компиляциядан өткен бағдарламаның объекті файлдар(\*.o; \*.obj;)

3.Компоновщик қосылғаннан кейін орындалатын файлдар(\*.exe;)

С++ ТІЛІНІҢ ШЫҒУ ТАРИХЫ

Тіл 1980-жылдардың басында Bell Labs компаниясының қызметкері Бьорн Штрауструп С тілін өзінің қажеттіліктері үшін бірқатар жетілдірулер жасаған кезде пайда болды.1970 жылдардың соңында Штрауструп Bell Labs лабораториясында кезек теориясының проблемалары бойынша жұмыс істей бастағанда, ол кездегі модельдеу тілдерін қолдануға тырысулар нәтижесіз болғанын, ал олардың тиімділігі шектеулі болғандықтан жоғары тиімді машиналық тілдерді қолдану өте қиын болатынын анықтады.

Диссертацияның тәжірибесін еске түсіре отырып, Штрауструп С тілін симула тіліндегі мүмкіндіктермен толықтыруға шешім қабылдады. C тілі, Bell компьютерлері жұмыс істейтін UNIX жүйесінің негізгі тілі бола отырып, жылдам, ерекшелігі мол және портативті. Штрауструп оған класстармен және объектілермен жұмыс істеу мүмкіндігін берді.

1983 жылға қарай тілге виртуалды функциялар, жүктеме функциялары мен операторлар, сілтемелер, тұрақтылар, пайдаланушылардың бос жадыны басқаруды бақылау, жақсартылған типті тексеру және коменттарийлердің жаңа стилі сияқты жаңа функциялар қосылды (//). Шығарылған тіл классикалық С тілінің кеңейтілген нұсқасы болуды тоқтатты және жаңа атқа ие болып, «C ++» болып өзгертілді. Оның алғашқы коммерциялық шығарылымы 1985 жылдың қазан айында болды.

НЕГІЗГІ БӨЛІМ

Дeрeктeр бaзaсы(Data Base) – бұл өзaрa бaйлaныcты aқпaрaтты, көбінесе үлкен көлемде сақтауға, өзгертуге және өңдеуге арналған ұйымдастырылған құрылым. Деректер базасы динамикалық сайттар үшін белсенді түрде қолданылады – көбінесе бұл интернет-дүкендер, порталдар және т.б.

ФУНКЦИЯ

Функция – бұл белгілі бір аяқталған әрекетте орындайтын сипаттамалар мен операторлардың атау берілген тізбегі. Функция параметрлерді қабылдап, мән қайтара алады.

Функциялар C++ тілінің маңызды компонентерінің бірі болып табылады.

Кез келген функцияның және кез келген айнымалының типі болады.

Егер функция ешқандай мән бермесе, онда ол void типін қолдану керек.

Функцияға келтірілген аргументтер тізімінен кейін, ашылатын жақша қойылады, содан кейін функцияның денесі орналасады.

Функцияны анықтауды құрама бөліктерін қарастырайық.

Міндетті түрде қажет етілмейтін класс модификаторының көмегімен, extern және static түйінді сөздерін қолдана

отырып, функцияның көріну аймағын айқын түрде көрсетуге болады.

Extern – функцияның программаның барлық модульдерінде ауқымды, яғни глобалды (келісім бойынша) түрде көрінуі;

Static – функцияның тек өзі анықталған модуль шеңберінде көрінуі.

Функцияның қайтаратын мәнінің типі жиым мен функциядан (бірақ жиымға немесе функцияға нұсқауыш бола алады) басқа кез келген тип түрінде болуы мүмкін. Егер функция оны шақырған программаға ешқандай мән қайтармайтын болса, онда void типі көрсетіледі.

ФАЙЛ

Файл - бұл мәліметтер массивін сақтауға бөлінген сыртқы жадтың аталатын аймағы. Файлдардағы мәліметтер әртүрлі сипатта болады: алгоритмдік немесе машиналық тілдегі бағдарламалар; бағдарламалардың жұмысына немесе бағдарламаны орындау нәтижелеріне арналған бастапқы мәліметтер; жеке мәтіндер; графикалық кескіндер және т.б.

Мәліметтеріне қол жеткізу тәсіліне қарай файлдарды *тізбекті* және *кездейсоқ қол жеткізу* файлдары деп екіге бөледі.Тізбекті файлға мәлімет жазу мен оқу оның басынан бастап байтқа байт тіркеле орналастырылып жүргізіледі, ал екінші түрдегі файлдар мәлдіметтерді өзіміз көрсеткен позицияға жазу мен оқу мүмкіндігін береді.

Стандартты кітапханада файлдармен жұмыс істеуге арналған үш класс бар, олар:

ifstream-кіріс файлдық ағымдар класы;

ofstream-шығыс файлдық ағымдар класы;

fstream-екібағытты файлдық ағымдар класы;

ШАРТТЫ ОПЕРАТОРЛАР

Шартты опереторды қандай да бір шарттың орындалуынан тәуелді құрастырғышты тармақталу деп атаймыз. Тармақталу операторлары: if, switch. if шартты операторы есептеу үрдісінің екі бағытта тармақталуы үшін қолданылады

Шартты оператор If   
Мұнда шарт логикалық өрнек, айнымалы немесе тұрақты.

Шартты оператор келесідей жұмыс істейді. Біріншіден, шарт ретінде жазылған өрнектің мәндері есептеледі. Егер бұл рас болса, мәлімдеме\_1 орындалады. Әйтпесе (жалған (жалған)) оператор\_2.

Switch операторы  
Бұл айнымалы мәндеріне байланысты белгілі бір операторлар орындалуы керек болған жағдайларда қажет.

ЦИКЛДАР

Цикл дегеніміз - бағдарламада сол код бөлімін қайталау. Қайталанатын әрекеттер тізбегі цикл денесі деп аталады. Бір цикл - бұл қадам немесе итерация. Цикл ішінде өзгеретін және оның соңына әсер ететін айнымалылар цикл параметрлері деп аталады.

Белгілі бір әрекеттерді бірнеше рет орындау үшін шартқа байланысты C ++ тілінде циклдар қолданылады. С++ тілінде келесі цикл түрлері бар:

for

while

For – eң жиі қолданылатын цикл. For циклі белгілі бір шарт ақиқат болғанша дейін орындала береді.

For циклының синтаксисі:

for( инициализация; условие; обновление ) {

  Исполняемый код

}

While цикл операторы шартты алдын ала белгісіз рет қайталап отыратын цикл операторы болып табылады.Егер жауап дұрыс болмаса немесе логикалық өрнек орындалмаса, программа жабылады.Себебі логикалық өрнектін дұрыстығы программаның басында тексеріледі де цикл мүлдем орындалмайды.Ал егер программа дұрыс болса, цикл қайталанып отыра береді.

While циклының синтаксисі:

while(условие)

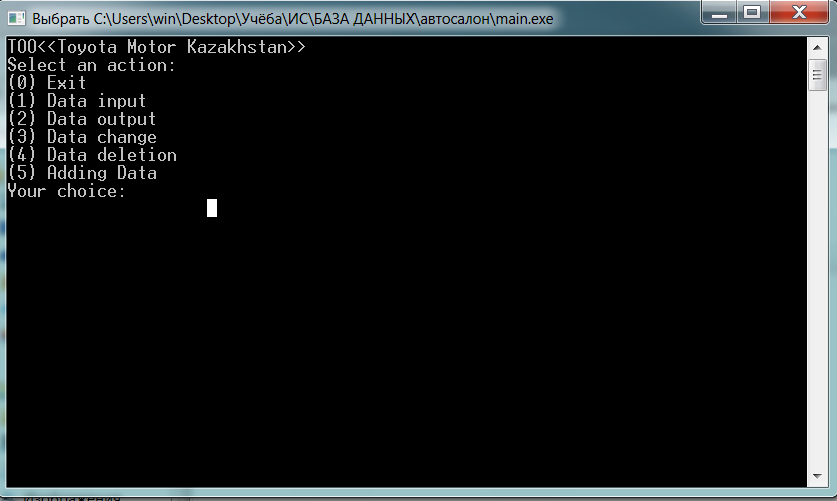
{

    // выполняемые действия

}

ФУНКЦИЯЛАРДЫ ТҮСІНДІРУ

Меню



Data input – командасы көмегімен біз деректер қорына ақпараттарды еңгізе аламыз.

Data output – деректер қорына еңгізілген ақпаратты экранға шығару.

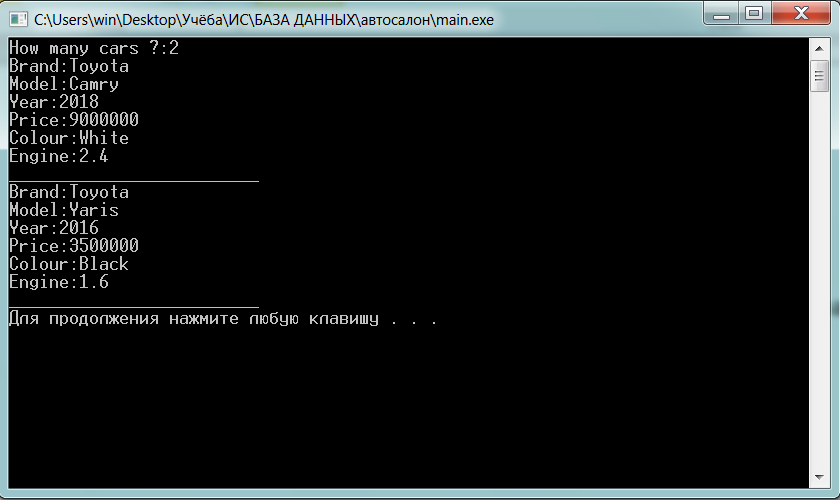
Data change – деректер қорына еңгізілген ақпаратты өзгерту.

Data deletion - деректер қорына еңгізілген ақпаратты өшіру.

Adding data - деректер қорына қосымша ақпаратты қосу.

Exit – командасы арқылы бағдарламадан шығамыз.

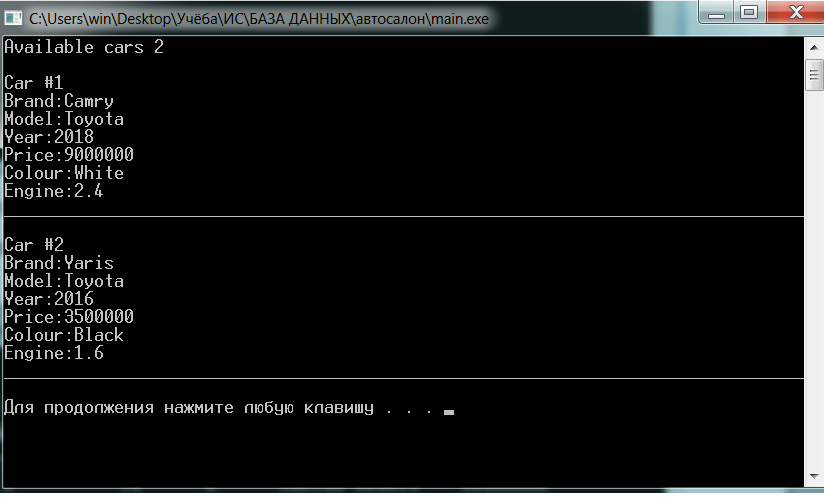
Data Input



(1-сурет)

1-суретте көрсетілгендей біз машинанын Маркасын, моделін, шыққан жылын, бағасын, түсін және қозғалтқыш көлемін еңгіземіз.

Data Output



(2-сурет)

2-суретте Data output (Ақпаратты шығару) функциясы деректер базасына еңгізілген (1-сурет) 2 автокөлікті энкранға шығарды.

ҚОРЫТЫНДЫ

Біз бұл курстық жұмысты С++ бағдарламалау тілінде жасадым.Ізденістеріммен талпынысымнын нәтижесінде көп материалды игеріп, С++Builder бағдарламасымен жұмыс жасау тәжірбиесіне ие болдым.Өзімнің берілген тапсырмам бойынша “Автосалон” машина сату жолын қарастырып, оны барлық жағынан көрсету болатын.Берілген кодты орындай отыра мәліметтерді файлға дұрыс сақтауды, әр тапсырманы орындау үшін фунцияны қолдануды меңгеріп, білімімді толықтырдым.

Деректерді қағаз жүсінде сақтап, реттеуге қарағанда, жаңа технологиярдың қол жетерлік қарапайым мүмкіншілігін пайдаланып, жұмыс барысын тиімді бағытта жеңілдетуге болады. Бағдарлама арқылы жеңілдетілген жұмыс уақытты үнемдейді. Жеңілдетілген жұмыс және үнемделген уақыт жақсы нәтижелерге жеткізеді.

Жалпы курстық жұмыс мен үшін өте пайдалы болды.Себебі курстық жұмысты орындау барысында мен файл туралы, структура туралы,шартты және циклдық операторлар туралы толығырақ танысып, үйрендім.Сонымен қатар С++ тілінің көмегімен деректер базасын құрып үйрендім, және кең көлемді бағдарламалық проектілер жасауға болатыны білдім.Менің ойымша бағдарламалауды әрбіреуіміз жақсы білуіміз кажет себебі бұл болашақтың көзі деп білемін.

ҚОЛДАНЫЛҒАН ӘДЕБИЕТТЕР

1.        Архангельский А.Я. Программирование в C++ Buіlder 6. М.: Бином, 2003.

2.        Бейзер Б. Тестирование чѐрного ящика. Технологии функционального тестирования программного обеспечения и систем. - СПб.: Питер, 2004. — 320 с.

3.        Буч Г., Рамбо Д., Якобсон И. Язык UML. Руководство пользователя. 2-е изд.: Пер. с англ. Мухин Н. – М.: ДМК Пресс, 2006. – 496 с.

4.        Вендров А. М. Практикум по проектированию программного обеспечения экономических информационных систем: Учеб.пособие. — 2-е изд., перераб. и доп. — М.: Финансы и статистика, 2006. - 192 с.

5.        Вендров А. М. Проектирование программного обеспечения экономических информационных систем: Учебник. — М.: Финансы и статистика, 2002. - 352 с.

6.        Гленфорд Майерс, Том Баджетт, Кори Сандлер. Искусство тестирования программ, 3-е издание = TheArtofSoftwareTesting, 3rd Edition. — М.: «Диалектика», 2012. — 272 с.

7.        Иванова Г. С. Технология программирования: Учебник для вузов.- М.: Изд-во МГТУ им. Н. Э. Баумана, 2003- 320 с. /Серия: Информатика в техническом университете/

8.        КрэгЛарман. Применение UML 2.0 и шаблонов проектирования. Введение в объектно-ориентированный анализ, проектирование и итеративную разработку. М.:Вильямс, 2007 – 736с.

9.        ЛайзаКриспин, Джанет Грегори. Гибкое тестирование: практическое руководство для тестировщиковПО и гибких команд = AgileTesting: A PracticalGuidefor TestersandAgileTeams. - М.: «Вильямс», 2010. - 464 с.

10.    Молчанов А. Ю. Системное программное обепечение. Лабораторный практикум. - СПб.: Питер, 2005.- 284 с.

Бағдарламалау Бөлімі

#include <iostream>

#include <fstream>

#include <string>

#include <windows.h>

using namespace std;

int \_stateMenu;

void Menu () {

cout<< "Select an action:" <<endl

<<"(0) Exit" <<endl

<<"(1) Data input" <<endl

<<"(2) Data output" <<endl

<<"(3) Data change" <<endl

<<"(4) Data deletion"<<endl

<<"(5) Adding Data" <<endl

<<"Your choice:";

cin>> \_stateMenu;

}

voidDataInitialization(){

ofstream \_buf("buffer.txt");

if(!\_buf)

cout<< "Error:Can not create a file!!!"<<endl;

\_buf.close();

}

voidDataEntry(){

string \_Brand,\_Model,\_Colour,\_Engine;

int \_Year,\_Price;

int n;

cout<<"How many cars ?:";

cin>>n;

ofstream record("Buffer.txt",ios::app);

if (record){

record<<n<<endl;

for (inti=0;i<n;i++){

cout<< "Brand:";

cin>>\_Brand;

cout<<"Model:";

cin>>\_Model;

cout<<"Year:";

cin>>\_Year;

cout<<"Price:";

cin>>\_Price;

cout<<"Colour:";

cin>>\_Colour;

cout<<"Engine:";

cin>>\_Engine;

record<<\_Model<<endl;

record<<\_Brand<<endl;

record<<\_Year<<endl;

record<<\_Price<<endl;

record<<\_Colour<<endl;

if (i<n-1)

record<<\_Engine<<endl;

else

record<<\_Engine;

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

}

}

else

cout<<"Error:Can not open the file!!!"<<endl;

record.close();

}

void Copy(){

ifstream reading ("Buffer.txt");

ofstream record ("Buffer\_.txt",ios::out);

if (reading) {

if (record){

string \_Brand,\_Model,\_Colour,\_Engine;

int \_Year,\_Price;

int n;

reading>> n;

record<< n <<endl;

for (inti = 0; i<n;i++){

reading>>\_Brand;

record<<\_Brand <<endl;

reading>>\_Model;

record<<\_Model <<endl;

reading>>\_Year;

record<<\_Year <<endl;

reading>>\_Price;

record<<\_Price <<endl;

reading>>\_Colour;

record<<\_Colour<<endl;

reading>>\_Engine;

if (i<n-1)

record<<\_Engine<<endl;

else

record<<\_Engine;

}

}

else

cout<<"Error:Can not open file!!!"<<endl;

}

else

cout<<"Error:Can not open buffer file!!!"<<endl;

record.close();

reading.close();

}

void Print(){

ifstream reading ("Buffer.txt");

if (reading){

string \_Brand,\_Model,\_Colour,\_Engine;

int \_Year,\_Price;

int n;

reading>> n;

cout<<"Available cars " << n <<endl<<endl;

for (inti = 0; i<n;i++)

{

cout<<"Car #"<<i+1<<endl;

reading>> \_Brand;

cout<< "Brand:"<<\_Brand<<endl;

reading>> \_Model;

cout<<"Model:"<<\_Model<<endl;

reading>> \_Year;

cout<<"Year:"<<\_Year<<endl;

reading>> \_Price;

cout<<"Price:"<<\_Price<<endl;

reading>> \_Colour;

cout<<"Colour:"<<\_Colour<<endl;

reading>> \_Engine;

cout<<"Engine:"<<\_Engine<<endl;

cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

}

}

else

cout<<"Error:Can not open File!" <<endl;

reading.close();

}

voidDataChange(){

Copy();

ifstream reading ("Buffer\_.txt");

ofstreamrecord ("Buffer.txt",ios::out);

if (reading ){

if (record){

string \_Brand,\_Model,\_Colour,\_Engine;

int \_Year,\_Price;

intn,\_n;

reading>> n;

cout<<"Choose one of the element"<< n <<"):";

cin>>\_n;

\_n--;

system ("cls");

record<<n<<endl;

if (\_n>=0&& \_n<n) {

for(inti=0;i<n;i++){

if (i !=\_n){

reading>>\_Brand;

record<<\_Brand<<endl;

reading>>\_Model;

record<<\_Model<<endl;

reading>>\_Year;

record<<\_Year<<endl;

reading>>\_Price;

record<<\_Price<<endl;

reading>>\_Colour;

record<<\_Colour<<endl;

reading>>\_Engine;

if (i< n - 1)

record<<\_Engine<<endl;

else

record<<\_Engine;

}

else {

cout<< "Brand:";

cin>>\_Brand;

cout<<"Model:";

cin>>\_Model;

cout<<"Year:";

cin>>\_Year;

cout<<"Price:";

cin>>\_Price;

cout<<"Colour:";

cin>>\_Colour;

cout<<"Engine:";

cin>>\_Engine;

record<<\_Brand<<endl;

record<<\_Model<<endl;

record<<\_Year<<endl;

record<<\_Price<<endl;

record<<\_Colour<<endl;

if (i<n-1)

record<<\_Engine<<endl;

else

record<<\_Engine;

reading>>\_Brand;

reading>>\_Model;

reading>>\_Year;

reading>>\_Price;

reading>>\_Colour;

reading>>\_Engine;

}

}

cout<<"Data has been changed!!!"<<endl;

}

else

cout<<"Wrong entered number!!!"<<endl;

} else

cout<<"Error:Can not open the file!!!"<<endl;

} else

cout<<"Error:Can not open the buffer file!!!"<<endl;

record.close();

reading.close();

remove("Buffer\_.txt");

}

voidDeleteData(){

Copy();

ifstream reading ("Buffer\_.txt");

ofstream record ("Buffer.txt",ios::out);

if (reading){

if(record){

string \_Brand,\_Model,\_Colour,\_Engine;

int \_Year,\_Price;

intn , \_n;

reading>>n;

int b = n - 1;

cout<<"Choose one car to change"<<n<<"): ";

cin>>\_n;

\_n--;

system("cls");

record<<b<<endl;

if (\_n>=0&&\_n<n){

for(inti=0;i<n;i++){

if(i !=\_n){

reading>>\_Brand;

record<<\_Brand<<endl;

reading>>\_Model;

record<<\_Model<<endl;

reading>>\_Year;

record<<\_Year<<endl;

reading>>\_Price;

record<<\_Price<<endl;

reading>>\_Colour;

record<<\_Colour<<endl;

reading>>\_Engine;

if (i< n - 1)

record<<\_Engine<<endl;

else

record<<\_Engine;

}

else {

reading>>\_Brand;

reading>>\_Model;

reading>>\_Year;

reading>>\_Price;

reading>>\_Colour;

reading>>\_Engine;

}

}

cout<<"Data has been deleted!!!"<<endl;

}

else

cout<<"Wrong entered number!!!"<<endl;

}

else

cout<<"Error:Can not open the file!!!"<<endl;

}

cout<<""<<endl;

record.close();

reading.close();

remove("Buffer\_.txt");

}

intAmountOfData(){

ifstream \_buf ("Buffer.txt");

int n;

if (\_buf){

\_buf>>n;

}

else cout<<"Error:Can not open buffer file!!!"<<endl;

\_buf.close();

return n;

}

voidAddData(){

string \_Brand,\_Model,\_Colour,\_Engine;

int \_Year,\_Price;

int n = AmountOfData() + 1;

ofstream record("Buffer.txt",ios::app);

ofstream record\_("Buffer.txt",ios::out | ios::in);

if (record\_){

record\_ << n <<endl;

}

else

cout<< "Error:Can not open the buffer file!!!" <<endl;

if (record) {

record<<endl;

cout<< "Brand:";

cin>>\_Brand;

cout<<"Model:";

cin>>\_Model;

cout<<"Year:";

cin>>\_Year;

cout<<"Price:";

cin>>\_Price;

cout<<"Colour:";

cin>>\_Colour;

cout<<"Engine:";

cin>>\_Engine;

record<<\_Brand<<endl;

record<<\_Model<<endl;

record<<\_Year<<endl;

record<<\_Price<<endl;

record<<\_Colour<<endl;

record<<\_Engine;

}

else

cout<< "Error:can not open the file!!!"<<endl;

record.close();

record\_.close();

}

int main () {

SetConsoleCP(1251);

DataInitialization();

Menu();

int \_actions;

stringfileName;

while (\_stateMenu != 0)

{

switch( \_stateMenu)

{

case 1:

system("cls");

DataEntry ();

system("pause");

system("cls");

Menu();

break;

case 2:

system ("cls");

Print();

system("pause");

system("cls");

Menu();

break;

case 3:

system("cls");

DataChange();

system("pause");

system("cls");

Menu();

break;

case 4:

system("cls");

DeleteData();

system("pause");

system("cls");

Menu();

break;

case 5:

system("cls");

AddData();

system("pause");

system("cls");

Menu();

break;

}

}

}