Министерство образования и науки РФ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Глазовский государственный педагогический институт имени В.Г. Короленко»

**ОТЧЕТ**

**по учебной практике**

**ПМ 01 Разработка программных модулей программного обеспечения для компьютерных систем**

**Специальность 09.02.03 Программирование в компьютерных системах**

|  |
| --- |
| Выполнил: |
| студент ОП СПО группа 33 |
| Романов Владислав Викторович |
|  |
| Руководитель практики: ст. преподаватель кафедры математики и информатики Касаткин К.А. |

г. Глазов, 2017

Спецификация

Название приложения: Справочник работника ГИБДД.

Создание базы данных стоящих на учете автомобилей с полями: марка, цвет, заводской и бортовой номера, дата выпуска, особенности конструкции и окраски, дата последнего техосмотра транспортного средства.

Просмотр информации о транспортном средстве, введенных в базу данных с помощью выбора из предоставленного списка.

Для работы с базой данных нужно, чтобы на компьютере был установлен сервер MySQL 5.5.

Текст программы

procedure TForm1.FormCreate(Sender: TObject);

begin

try

MySQL55Connection1.Connected:=true;

except

ShowMessage('Подключение к БД: ОШИБКА!');

exit;

end;

try

SQLTransaction1.Active:=true;

except

ShowMessage('Транзакция: ОШИБКА!');

exit;

end;

end;

procedure TForm1.MySQL55Connection1AfterConnect(Sender: TObject);

begin

end;

procedure TForm1.Button2Click(Sender: TObject);

var a,b,c:string;

begin

try

SQLQuery1.Active:=false;

SQLQuery1.ExecSQL;

SQLQuery1.SQL.Clear;

a:='SELECT \* FROM tbl\_pass\_driver';

SQLQuery1.SQL.Add(a);

SQLQuery1.Open;

except

SQLQuery1.Close;

end;

ComboBox1.Text:= ''; ComboBox2.Text:= ''; ComboBox3.Text:= ''; ComboBox4.Text:= ''; ComboBox5.Text:= '';

try

SQLQuery2.Active:=false;

SQLQuery2.ExecSQL;

SQLQuery2.SQL.Clear;

b:='SELECT \* FROM tbl\_pass\_auto';

SQLQuery2.SQL.Add(b);

SQLQuery2.Open;

except

SQLQuery2.Close;

end;

try

SQLQuery3.Active:=false;

SQLQuery3.ExecSQL;

SQLQuery3.SQL.Clear;

c:='SELECT \* FROM tbl\_mark';

SQLQuery3.SQL.Add(c);

SQLQuery3.Open;

except

SQLQuery3.Close;

end;

end;

procedure TForm1.Button3Click(Sender: TObject);

var

str\_sql : string;

begin

try

SQLQuery1.Active := false;

SQLQuery1.SQL.Clear;

str\_sql := 'SET character\_set\_client='+#39+'utf8'+#39+',character\_set\_connection='+#39+'cp1251'+#39+',character\_set\_results='+#39+'utf8'+#39+';';

SQLQuery1.sql.add(str\_sql);

SQLQuery1.ExecSQL;

SQLQuery1.SQL.Clear;

SQLQuery1.sql.add('SELECT \* FROM tbl\_pass\_driver;');

SQLQuery1.Open;

except

ShowMessage('SQL-запрос: ОШИБКА!');

exit;

end;

end;

procedure TForm1.Button4Click(Sender: TObject);

var

str\_sql : string;

begin

try

SQLQuery2.Active := false;

SQLQuery2.SQL.Clear;

str\_sql := 'SET character\_set\_client='+#39+'utf8'+#39+',character\_set\_connection='+#39+'cp1251'+#39+',character\_set\_results='+#39+'utf8'+#39+';';

SQLQuery2.sql.add(str\_sql);

SQLQuery2.ExecSQL;

SQLQuery2.SQL.Clear;

SQLQuery2.sql.add('SELECT \* FROM tbl\_pass\_auto;');

SQLQuery2.Open;

except

ShowMessage('SQL-запрос: ОШИБКА!');

exit;

end;

end;

procedure TForm1.Button5Click(Sender: TObject);

var

str\_sql : string;

begin

try

SQLQuery3.Active := false;

SQLQuery3.SQL.Clear;

str\_sql := 'SET character\_set\_client='+#39+'utf8'+#39+',character\_set\_connection='+#39+'cp1251'+#39+',character\_set\_results='+#39+'utf8'+#39+';';

SQLQuery3.sql.add(str\_sql);

SQLQuery3.ExecSQL;

SQLQuery3.SQL.Clear;

SQLQuery3.sql.add('SELECT \* FROM tbl\_mark;');

SQLQuery3.Open;

except

ShowMessage('SQL-запрос: ОШИБКА!');

exit;

end;

end;

procedure TForm1.ComboBox1Change(Sender: TObject);

begin

end;

procedure TForm1.Button1Click(Sender: TObject);

var a,b,c:string;

begin

try

SQLQuery1.Active:=false;

SQLQuery1.ExecSQL;

SQLQuery1.SQL.Clear;

a:='SELECT \* FROM tbl\_pass\_driver where family\_name = "'+ComboBox1.Text+'" and first\_name= "'+ComboBox2.Text+'" and father\_name= "'+ComboBox3.Text+'"';

b:='SELECT \* FROM tbl\_pass\_auto where state\_number = "'+ComboBox4.Text+'"';

SQLQuery1.SQL.Add(a);

SQLQuery1.Open;

except

SQLQuery1.Close;

end;

try

SQLQuery2.Active:=false;

SQLQuery2.ExecSQL;

SQLQuery2.SQL.Clear;

b:='SELECT \* FROM tbl\_pass\_auto where state\_number = "'+ComboBox4.Text+'"';

SQLQuery2.SQL.Add(b);

SQLQuery2.Open;

except

SQLQuery2.Close;

end;

try

SQLQuery3.Active:=false;

SQLQuery3.ExecSQL;

SQLQuery3.SQL.Clear;

c:='SELECT \* FROM tbl\_mark where factory\_number = "'+ComboBox5.Text+'"';

SQLQuery3.SQL.Add(c);

SQLQuery3.Open;

except

SQLQuery3.Close;

end;

end;

end.

Описание программы

Программа Справочник работника ГИБДД может выводить информацию о транспортном средстве, информацию о водителе, осуществлять просмотр автомобиля владельца и возвращать первоначальную таблицу.

Формуляр

Операционная система: Windows® Vista/7/ 8/10 64-bit с последним пакетом обновлений.

Процессор: Premium Vision AMD

• Видеокарта: Radeon X1950Pro 256MB

• Память: видеопамять — 50 МБ, минимальная ОЗУ системы — 100 MB

• Жесткий диск: 2100 об./мин., с 200 MB свободного места

Описание применения

Программа Справочник работника ГИБДД может применяться в ГИБДД, и с помощью выборки в базе данных найти данные о владельце и транспортном средсвте.

Руководство системного программиста

Использовать операционная систему: Windows® Vista/7/ 8/10 64-bit с последним пакетом обновлений. Установить MySQL Server 5.5. Подключить базу данных captain к MySQL Server с помощью HeidiSQL. Запустить программу Справочник командира с помощью среды программирования Lazarus.

Руководство программиста

В случае замены базы данных (БД) нужно скачать исходный код с репозитория(?). В запросах к базе данных нужно поменять поля, при этом сохраняя количество полей в запросе такое же, как количество полей в базе данных. Можно в базу данных добавлять новые таблицы и поля. Уже в созданные таблицы можно добавлять новые поля. Модифицировать базу данных можно с помощью программы HeidiSQL.

Руководство оператора

При входе в программный продукт открывается таблица с военнослужащими. Выбрать определенного военнослужащего из заданного списка для вывода информации о нем. Кнопка «Мать» выводить информацию о матери военнослужащего. Действия кнопки «Отец» аналогично. Далее кнопка «Сброс» возвращает первоначальную таблицу warpeople.

Описание языка.

Для создания программного продукта Справочник работника ГИБДД использовался язык программирования Pascal и среда программирования Lazarus 1.6. Библиотека компонентов для MySQL версии 5.5.

Руководство по техническому обслуживанию.

Единственное, что нужно выяснить это минимальные требования к компьютеру они были указаны в формуляре. Для тестирования компьютера можно использовать программу (AIDA), если ваш компьютер удовлетворяет минимальным требованиям, то можно использовать программу Справочник работника ГИБДД.

Программа и методика испытания

Тест план.

Проверить все функции программы Справочник работника ГИБДД на корректность работы:

1. Вывод таблицы «tbl\_mark».

2. Выбор фамилии, имени и отчество водителя.

3. Вывод информации о ТС.

4. Вывод информации о водителе.

5. Вывод всей информации.

6. Возращение к исходной таблице.

7. Проверка корректности вывода таблицы «tbl\_mark».

8. Проверка корректности вывода таблицы «tbl\_pass\_driver».

9. Проверка корректности вывода таблицы «tbl\_pass\_auto».

10. Проверка корректности вывода информации.

Чек лист: Поиск спортсмена

1. Проверка корректности поиска по полю full\_name .

Тест case: Поиск информации о транспортном средстве

**Условия воспроизведения:**

1.Установить программу Справочник работника ГИБДД.

**Как воспроизвести:**

1. Выбор водителя.

2. Проверить на корректность вывода данных в таблице.

**Фактический результат:**

Выводятся не корректные данные в таблице.

**Ожидаемый результат:**

Корректные результаты при поиске данных.

Пояснительная записка

Введение

Название программы: Справочник работника ГИБДД.

Назначение и область применения

Программа Справочник работника ГИБДД применяется в сфере поиска информации транспортного средства водителя в ГИБДД. Программа Справочник работника ГИБДД будет хорошим дополнение для любого работника ГИБДД, так как она может автоматизировать трудовой процесс ГИБДД.

Классы решения задач: Программа Справочник работника ГИБДД решает проблему поиска данных водителя и транспортного средства.

Технические характеристики

1. Задачи: создать функционал для работы программы Справочник работника ГИБДД, создать базу данных, написать пользовательское приложение.

2. Алгоритмы: алгоритмы в данной программе не применялись, потому что все алгоритмы были реализованы разработчиками MySQL.

3. Описание и обоснованный выбор технических и программных средств: Lazarus был выбран, потому что там есть компоненты для работы с базой данных, MySQL был выбран, потому что он свободно распространяемый и имеет широкий спектр запросов, HeidiSQL был выбран, потому что он свободно распространяемый и мало занимает места на диске и имеет понятный интерфейс.

Технико-экономические показатели.

Программа Справочник работника ГИБДД имеет преимущество перед ручным поиском данных водителя и данных транспортного средства тем, что действия автоматизированы, поиск данных станет легче, возможность формирования электронных документов.

Источники

1. Иванова Г.С. Технология программирования: учебник / Г.С. Иванова. — М. : КНОРУС, 2011. — 336 с. ISBN 978-5-406-00519-4

2. Компоненты для работы с MySQL в Lazarus [Электронный ресурс]. – Режим доступа: http://www.freepascal.ru/article/lazarus/20090416150500 (дата обращения 11.05.2017)

3. Работа с компонентами подклюения к базе данных MySQL в Lazarus [Электронный ресурс]. – Режим доступа: http://wiki.freepascal.org/Lazarus\_Database\_Tutorial/ru (дата обращения 12.05.2017)

4. Создание базы данных в MySQL [Электронный ресурс]. – Режим доступа: http://generaltea.ru/kak-pomenyat-kodirovku-vsex-tablic-v-mysql (дата обращения 18.05.2017)

5. Запросы MySQL в Lazarus [Электронный ресурс]. – Режим доступа: http://www.programmersforum.ru/showthread.php?t=227174 (дата обращения 18.05.2017)

6. Автомобильные номера [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/Автомобильные \_номера (дата обращения 24.05.2017)

7. Заводской номер (VIN) [Электронный ресурс]. – Режим доступа: http://www.tks.ru/auto/9000000004 (дата обращения 24.05.2017)