Одеський національний політехнічний університет

Кафедра «Комп'ютеризовані системи управління»

**КУРСОВА РОБОТА**

з дисципліни «Основи технології .NET»

на тему: «Створення баз даних»

Варіант 21

Студента 1 курсу, групи НАТ–193

спеціальності «Автоматизація та комп'ютерно-інтегровані технології»

Прізвище: Мамін Є.Д.

Керівник: доц. Сперанський В.О.

Національна шкала:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Кількість балів:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Оцінка: ECTS\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Члени комісії \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(підпис) (прізвище та ініціали)

\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(підпис) (прізвище та ініціали)

\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(підпис) (прізвище та ініціали)

м. Одеса – 2020 рік

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 3](#_2bn6wsx)

[Постановка задачи варианта №21 3](#_qsh70q)

[РАЗДЕЛ 1 ЭТАПЫ ПРОЕКТИРОВАНИЯ И ИХ РЕАЛИЗАЦИЯ 5](#_3as4poj)

[1.1](#_1pxezwc) Календарное планирование проекта 5

[1.1](#_49x2ik5) Описание проектирования программы 6

[1.1.1](#_2p2csry) Создание базы данных. 6

[1.1.2](#_147n2zr) Функции программы 6

[1.1.3](#_3o7alnk) Создание интерфейса пользователя 9

[1.1.4](#_23ckvvd) Написание главной функции 10

[1.1.5](#_ihv636) Библиотеки 12

[РАЗДЕЛ 2 СПРАВОЧНАЯ ИНФОРМАЦИЯ ДЛЯ ПОЛЬЗОВАТЕЛЯ 14](#_32hioqz)

[1.1](#_lnxbz9) Краткое описание продукта 14

[1.2](#_1hmsyys) Ограниченный вход 14

[1.3](#_41mghml) Панель управления 14

[1.4](#_2grqrue) Импортирование базы данных 15

[1.5](#_vx1227) Создание базы данных 16

[1.6](#_3fwokq0) Управление информацией базы данных 16

[1.6.1](#_1v1yuxt) Добавление абитуриента 16

[1.6.2](#_4f1mdlm) Добавление данных нового абитуриента после указанного пользователем. 17

[1.6.3](#_2u6wntf) Удаление абитуриента со средним баллом аттестата ниже заданного 17

[1.6.4](#_19c6y18) Вывод данных абитуриента по ключу 17

[ВЫВОДЫ 19](#_3tbugp1)

[СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ 20](#_28h4qwu)

ВВЕДЕНИЕ

**База данных** – это организованная структура, предназначенная для хранения информации. В те годы, когда формировалось понятие баз данных, в них хранились только данные, однако сегодня большинство систем управления базами данных (СУБД) позволяет размещать в своих структурах не только данные, но и методы (то есть программный код), с помощью которых происходит взаимодействие с потребителем или с другими программно-аппаратными комплексами. В современных базах данных хранятся не только данные, но и информация.

Под базой данных понимают хранилище структурированных данных. При этом данные должны быть непротиворечивы, минимально избыточны и целостны. База данных создаётся для хранения и доступа к данным, содержащим сведения о некоторой предметной области. Степень детализации информации определяется рядом факторов. Прежде всего, целью использования информации из баз данных и сложностью информационных процессов, существующих в пределах предметной области в конкретных условиях.

Постановка задачи варианта № 21

1. Спроектировать структуру информационной базы для своего варианта задания (Приложение А).
2. Создать основную таблицу информационной базы в виде динамического массива структур (использование списка из элементов типа структура оценивается выше).
3. Предусмотреть функции контроля корректности всех вводимых данных (например, неотрицательный возраст, дата продажи не ранее, чем 100 лет назад и т.п.).
4. Заполнить основную таблицу не менее чем 20 записями.
5. Обеспечить сохранение и загрузку данных во внешний файл/из файла.
6. Реализовать меню пользователя, соответствующие функции которого обеспечивают:
7. создание информационной базы той предметной области, которая определена постановкой задачи;
8. добавление новых записей в базу данных (с контролем корректности вводимых значений);
9. поиск записей по *ключевому* полю;
10. редактирование записей базы данных;
11. удаление ненужных записей;
12. выдачу отчётов по запросу.
13. Выполнить оригинальную обработку, указанную в постановке конкретной задачи для вашего варианта (Приложение А).
14. Организовать ввод пароля при выполнении операций изменения данных.
15. Добавить справочную систему пользователя по созданному приложению.
16. Подготовить презентацию по разработанному приложению.

Сформировать базу данных – расписание движения автобусов из города Одесса. Каждая запись содержит следующие сведения следующие сведения:

* город назначения;
* *номер автобуса;*
* время отправления;
* время прибытия в пункт назначения;
* стоимость проезда.

*Предусмотреть:*

а) выдачу справки о всех автобусах указанного направления (Киев, Варшава и т.д.).

б) выдачу справки о времени прибытия каждого из автобусов указанного направления в пункт назначения;

в) выдачу информации о времени отправления автобуса с указанным номером;

г) удаление из базы записей об автобусах указанного направления.

РАЗДЕЛ 1 ПРОЕКТИРОВАНИЕ И ИХ РЕАЛИЗАЦИЯ

* 1. Календарное планирование проекта

Поэтапное календарное планирование для выполняемого курсового проекта представлено в табл. 1.1.

Таблица 1.1 – Календарный план курсового проектирования

|  |  |  |  |
| --- | --- | --- | --- |
| № | Наименование | Содержание | Дата окончания |
| 1 | Начало | Изучение постановочного задания, предметной области | 10.05.2020 |
| 2 | Теоретическая подготовка | Изучение литературы по дисциплины | 10.05.2020 |
| 3 | Проектирование модулей проекта | Разработка компонентов программы, их функций, свойств, а также связь между ними. | 13.05.2020 |
| 4 | Разработка программной реализации | Написание непосредственно самой программы с использованием изученных теоретических знаний | 14.05.2020 |
| 5 | Тестирование | Проверка созданного приложения на предмет некорректного поведения при вводе ошибочных значений, определение уровня эргономики, устойчивости разработки | 25.05.2020 |
| 5 | Написание документации | Написание документации с описанием выполненной работы и трудностями с которыми пришлось встретиться во время её выполнения | 27.05.2020 |
| 6 | Создание презентации | Выбор наиболее интересных этапов написания программы и подготовка презентации | 30.05.2020 |

* 1. Описание проектирования программы
     1. Создание базы данных. Проектирование архитектуры приложения начинается с формирования базы данных согласно пунктам задания. Экземпляром базы данных в коде являются две структуры: структура «Password check» из 4 полей со строковым типом данных; структура «Bus schedule» из 5 полей, 4 из который со строковым типом данных и 1 численным с плавающей точкой типом данных.
     2. Пример кода данной структуры:

public struct PasswordCheck

{

public string name;

public string surname;

public string login;

public string password;

}

public struct BusSchedule

{

public string destination;

public string busnum;

public string departuretime;

public string arrivetime;

public double price;

}

* + 1. Функции программы

При описании методов работы с массивами были выполнены такие задачи как: создание массива, удаление массива, создание базы, открытие базы, сохранение базы и загрузка базы. Пример кода предоставлен ниже.

//работа со массивами

public static int N = 0, M = N;

public static BusSchedule[] busschedules = new BusSchedule[N];

public static int K = 0, D = K;

public static PasswordCheck[] passwordcheck = new PasswordCheck[K];

public void ArrayResBusSchedule()

{

Array.Resize(ref busschedules, N);

}

public void ArrayResPasswordCheck()

{

Array.Resize(ref passwordcheck, K);

}

public void Delete(int actual)

{

int NN = N - 1;

BusSchedule[] TempArray = new BusSchedule[NN];

Array.Copy(busschedules, 0, TempArray, 0, actual);

Array.Copy(busschedules, actual + 1, TempArray, actual, N - actual - 1);

Clear();

N = NN;

M = N;

ArrayResBusSchedule();

Array.Copy(TempArray, 0, busschedules, 0, N);

Array.Clear(TempArray, 0, N);

}

public void Clear()

{

Array.Clear(busschedules, 0, N);

N = 0;

M = 0;

}

//Сохранение и загрузка файлов.

public void Write()

{

SaveFileDialog savefiledialog = new SaveFileDialog();

if (savefiledialog.ShowDialog() == DialogResult.OK)

{

string filename = savefiledialog.FileName;

FileStream fs = new FileStream(filename, FileMode.OpenOrCreate, FileAccess.Write);

BinaryWriter bw = new BinaryWriter(fs, Encoding.UTF8);

bw.Write(ClassforClass.N);

bw.Write(ClassforClass.M);

for (int i = 0; i < ClassforClass.N; i++)

{

bw.Write(ClassforClass.busschedules[i].arrivetime);

bw.Write(ClassforClass.busschedules[i].busnum);

bw.Write(ClassforClass.busschedules[i].destination);

bw.Write(ClassforClass.busschedules[i].departuretime);

bw.Write(ClassforClass.busschedules[i].price);

}

MessageBox.Show("Данные сохранены");

bw.Close();

fs.Close();

}

}

public void Read()

{

ClassforClass cc = new ClassforClass();

OpenFileDialog openfiledialog = new OpenFileDialog();

if (openfiledialog.ShowDialog() == DialogResult.OK)

{

string filename = openfiledialog.FileName;

FileStream fs = new FileStream(filename, FileMode.Open, FileAccess.Read);

BinaryReader br = new BinaryReader(fs, Encoding.UTF8);

ClassforClass.N = br.ReadInt32();

ClassforClass.M = br.ReadInt32();

cc.ArrayResBusSchedule();

for (int i = 0; i < ClassforClass.N; i++)

{

ClassforClass.busschedules[i].arrivetime = br.ReadString();

ClassforClass.busschedules[i].busnum = br.ReadString();

ClassforClass.busschedules[i].destination = br.ReadString();

ClassforClass.busschedules[i].departuretime = br.ReadString();

ClassforClass.busschedules[i].price = br.ReadInt32();

}

MessageBox.Show("Данные загружены");

br.Close();

fs.Close();

}

}

Также в программе были созданы функции, предусмотренные постановкой задачи:

* Функция вывода данных о маршруте. Не возвращает никаких значений.
* Функция, которая находит маршрут по номеру автобуса и выводит информацию в псевдографическом окне.
* Функция, которая редактирует данные о маршруте. Внутри функции происходит ввод известных данных искомого маршрута. Если маршрут найден, то в псевдографическом окне происходит ввод данных о найденом маршруте. Если маршрут не найден, то выводится соответствующее псевдографическое окно.
* Функция, которая удаляет из базы записей об автобусах указанного направления. Поиск автобуса происходит внутри функции. Если данные о автобусе найдены, то они удаляются из базы, если автобусов не было найдено, то выводится соответствующее псевдографическое окно.
* Функция, которая выводит информацию о всех маршрутах, которые находятся в списке, который привязан к базе в псевдографическом окне. Если маршрутов нет, то выводится соответствующее сообщение в псевдографическом окне.
* Функция, которая выводит информацию об абитуриенте по заданному ключу. Она вызывает интерактивный набор ключа. По выбранному ключу в псевдографическом окне выводится информация об абитуриентах, подходящих под заданный ключ. Если абитуриентов не найдено, выводится соответствующее сообщение в псевдографическом окне.
* Функция, которая удаляет маршрут из базы.

Пример кода данных функций:

//функции, предусмотренные постановкой задачи

void Delete(int actual)

void Clear()

void ArrayResBusSchedule()

void ArrayResPasswordCheck()

void Write()

void Read()

Запись массива в файл осуществлялось с помощью циклов, указанных во фрагменте кода:

for (int i = 0; i < ClassforClass.N; i++)

{

bw.Write(ClassforClass.busschedules[i].arrivetime);

bw.Write(ClassforClass.busschedules[i].busnum);

bw.Write(ClassforClass.busschedules[i].destination);

bw.Write(ClassforClass.busschedules[i].departuretime);

bw.Write(ClassforClass.busschedules[i].price);

}

* + 1. Создание интерфейса пользователя

Для реализации данного проекта были использованы Windows Forms и соответствующие элементы.

* + 1. Написание главной функции

Написание главной функции, начинается с создания проверки на знание пользователем пароля для открытия уже созданной базы данных, ее редактирования и удаления. При создании нового пароля, старый пароль затирается и происходит выделение памяти для нового ключа вода,сохраняя его в файл. Реализация пароля в коде:

for (int i = 0; i < ClassforClass.K; i++)

{

if (login == ClassforClass.passwordcheck[i].login && password == ClassforClass.passwordcheck[i].password)

{

ClassforClass.D = i;

count++;

this.Hide();

MainForm mainform = new MainForm();

mainform.Show();

break;

}

}

for (int i = 0; i < ClassforClass.K; i++)

{

if (login == ClassforClass.passwordcheck[i].login && password == ClassforClass.passwordcheck[i].password)

{

ClassforClass.D = i;

count++;

this.Hide();

MainForm mainform = new MainForm();

mainform.Show();

break;

}

}

Дальнейшее выполнение программы происходит через следующие формы.

* + 1. Библиотеки

Для написания данной программы необходимо подключить некоторые стандартные библиотеки языков С и С++. Их перечень представлен в коде

using System;

using System.IO;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

РАЗДЕЛ 2 СПРАВОЧНАЯ ИНФОРМАЦИЯ ДЛЯ ПОЛЬЗОВАТЕЛЯ

* 1. Краткое описание продукта

Приложение представляет собой ПО, в котором разработана база данных «Автовокзал». Функциональность программы включает:

* управление данными информационной базы: добавление, редактирвоание, удаление маршрутов;
* выдачу справок о маршрутке с указанными данными;
* удаление записей об автобусах указанного направления;
* выдачу сведений о всех маршрутах;
  1. Ограниченный вход

При открытии базы данных пользователь должен ввести уникальный код, который позволит ему управлять информацией, записанной в базе (рис. 2.1), в противном случае будет выдано соответствующее сообщение в псевдографическом окне и доступ к базе будет закрыт.

Рисунок 2.1 Авторизация пользователя

* 1. Панель управления

Панель управления (рис. 2.2) представляет собой список из 4 пунктов и 4 управляющих кнопки меню, каждый из которых выполняет свою функцию.

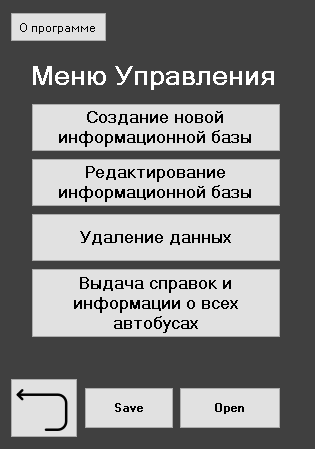


Рисунок 2.2 Панель управления базой данных

* 1. Импортирование базы данных

Импорт базы данных предполагает открытие заранее созданной базы данных с указанием адреса её расположения.



Рисунок 2.3 Кнопка открытия файла

* 1. Создание базы данных

При создании новой базы данных пользователю потребуется указать адрес желаемого расположения файла.



Рисунок 2.4 Окно при создании новой базы данных

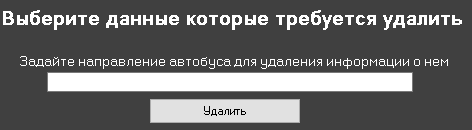
* 1. Управление информацией базы данных
     1. Удаление записей об автобусах в указанном направлении. При выборе этого пункта меню «Удаление данных» производится ввод направления автобуса, который указывает пользователь, после чего данные, которые не соответствуют этому критерию будут удалены.

Рисунок 2.5 Окно запроса удаления данных

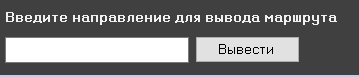
* + 1. Вывод данных маршрутов. При выводе данных маршрута по ключу пользователю предлагаются ключевое поле (рис.2.6 ), после которых откроется окно ввода информации по заданному параметру. Затем производится вывод заданной информации на экран.

Рисунок 2.6 Ключевые поля для поиска конкретного абитуриента

1.6.3 Вывод всех имеющихся данных. При выводе всех данных

маршрута пользователю предлагается заполнить нужные ему поля(рис.2.7)

для редактирования.

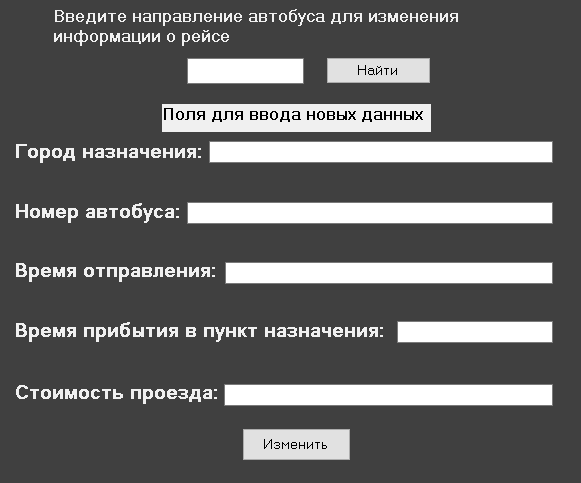


Рисунок 2.6 Ключевые поля для поиска конкретного абитуриента.

ВЫВОДЫ

На основании знаний, приобретенных на курсе «Основи технології .NET», было создано приложение, которое автоматизирует управление базы данных, упрощая поиск, редактирование и удаление информации, и экономя время на ее обработку. Курсовая работа предоставляет практическое применение знаний и повышает уровень профессиональных навыков.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Троелсен, Эндрю Язык программирования C# 5.0 и платформа .NET 4.5 / Эндрю Троелсен. - М.: Вильямс, 2015. - 633 c.
2. Фленов, Михаил Библия C# (+ CD-ROM) / Михаил Фленов. - М.: БХВ-Петербург, 2011. - 560 c.
3. Зиборов, В. В. Visual C# 2012 на примерах / В.В. Зиборов. - М.: БХВ-Петербург, 2013. - 480 c.