МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний аерокосмічний університет ім. М.Є. Жуковського

“Харківський авіаційний інститут”

Факультет СУЛА

Кафедра математичного моделювання та штучного інтелекту

**Лабораторна робота №10**

з Організація баз даних

на тему: «SQLite»

Виконав: студент 3 курсу групи №335a

напряму підготовки(спеціальності):

122 Комп’ютерні науки

(шифр і назва напряму підготовки(спеціальності))

Гринюк М.О.

(прізвище й ініціали студента)

Прийняв: ст. викл., к.т.н. Коробчинський К.П.

(посада, науковий ступінь, прізвище й ініціали)

Харків – 2020

**Цель работы:** Изучить принципы работы SQLite

**Ход работы:**

1. Пройти тест
2. Использовать в своём проекте класс логирования
3. Подключить проект C# проверить соединение с БД вывести отладочную информацию в лог файл.
4. Создать Windows Form App использующий по Вашему варианту БД на MS SQL
5. Для одной-двух своих таблиц создать БД используя SQLite и реализовать действия:
6. Nuget Pakages
7. Вывод на экран
8. Вставка данных в таблицу
9. Удаление данных из БД
10. Обновление данных
11. Все функции сделать в виде wrapper (www.codeproject.com Using-SQLite)
12. Оформить отчёт и сохранить на STM.

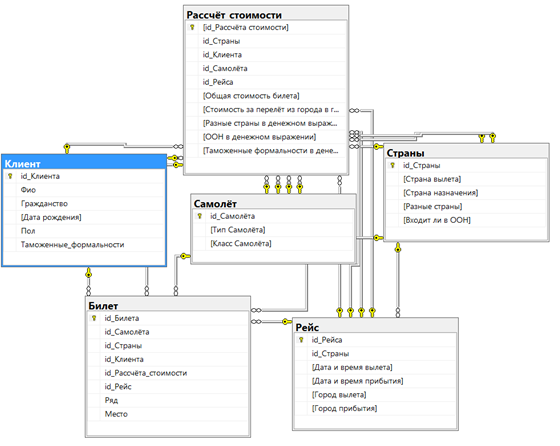
**Содержание работы:**

1. Для своего варианта на основе построенной логической модели своего варианта создать БД (**MS SQL**).
2. Подключить класс для вывода логирования.
3. Выполнить соединение с правильными и не правильными тестовыми данными (Сonnectionstrings sqlite)
4. Создать форму для входа и после успешной авторизации запустить основную, если авторизация не успешная, то главная  форма не должна создаваться. (повышенная сложность на максимальное количество баллов)
5. Использовать**SQLite**по **ходу работы**
6. Для каждого пункта показать программный код и пример работы (скрин)

**Выполнение работы**

1. Для своего варианта на основе построенной логической модели своего

варианта создать БД;

**Логическая модель БД**

**Использовать в своём проекте класс логирования**

class ConnectedLayer

{

public string connectionString = ConfigurationManager.ConnectionStrings["MyConnectionString"].ConnectionString;

Logger logger = LogManager.GetCurrentClassLogger();

SqlConnectionStringBuilder builder;

public ConnectedLayer()

{

builder = new SqlConnectionStringBuilder(connectionString);

//ds = new DataSet();

var oSqlCon = new SqlConnection(connectionString);

using (SqlConnection connection = new SqlConnection(connectionString))

{

logger.Trace(

"Connection to" + Environment.NewLine +

"Data Source: " + oSqlCon.DataSource + Environment.NewLine +

"Database: " + oSqlCon.Database + Environment.NewLine +

"State: " + oSqlCon.State +

"User: " + builder.UserID +

"Catalog: " + builder.InitialCatalog +

"Server: " + builder.DataSource

);

Configuration config = ConfigurationManager.OpenExeConfiguration(ConfigurationUserLevel.None);

config.ConnectionStrings.ConnectionStrings.Add(

new ConnectionStringSettings("ConnectionStr", "SomeConnectionString")

);

config.Save();

ConnectionStringsSection section = config.GetSection("connectionStrings") as ConnectionStringsSection;

if (section.SectionInformation.IsProtected)

{

section.SectionInformation.UnprotectSection();

}

else

{

section.SectionInformation.ProtectSection("DataProtectionConfigurationProvider");

}

config.Save();

}

}

public DataSet Read(string command)

{

Logger logger = LogManager.GetCurrentClassLogger();

var oSqlCon = new SqlConnection(connectionString);

builder = new SqlConnectionStringBuilder(connectionString);

oSqlCon.Open();

SqlDataAdapter oSqlDtAdptr = new SqlDataAdapter(command, oSqlCon);

DataSet ds = new DataSet();

oSqlDtAdptr.Fill(ds);

logger.Trace("UserId: " + builder.UserID);

logger.Trace("UserCommand: " + command);

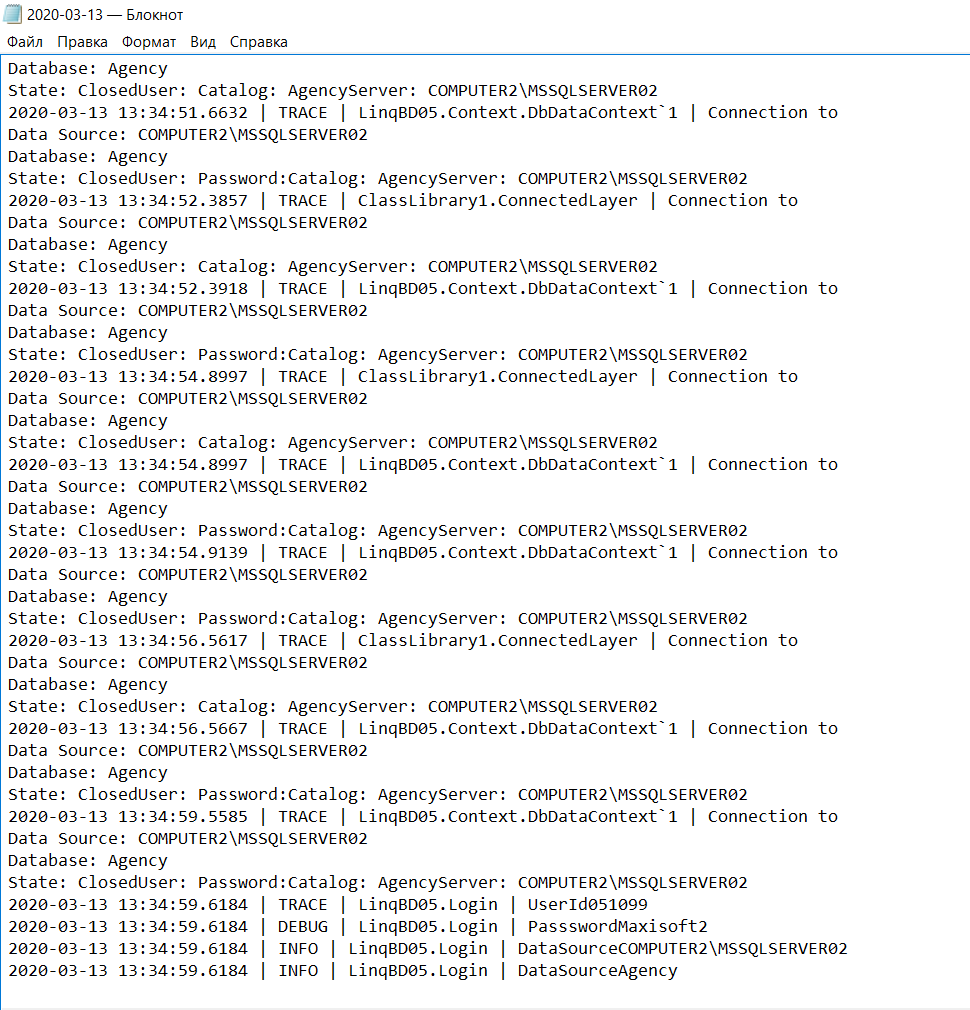
logger.Trace("User's DataSet: " + ds);

oSqlCon.Close();

return ds;

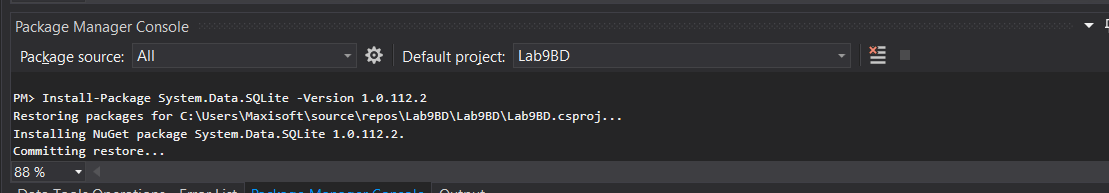
}}

**Подключить проект C# проверить соединение с БД вывести отладочную информацию в лог файл.**



**Для одной-двух своих таблиц создать БД используя SQLite и реализовать действия:**

**Nuget Pakages:**



* **Вставка данных в таблицу**

**Model:**

public class Ticket : INotifyPropertyChanged

{

private string arrival;

private string departure;

private int price;

public int Id { get; set; }

public string Arrival

{

get { return arrival; }

set

{

arrival = value;

OnPropertyChanged("Title");

}

}

public string Departure

{

get { return departure; }

set

{

departure = value;

OnPropertyChanged("Company");

}

}

public int Price

{

get { return price; }

set

{

price = value;

OnPropertyChanged("Price");

}

}

public event PropertyChangedEventHandler PropertyChanged;

public void OnPropertyChanged([CallerMemberName]string prop = "")

{

if (PropertyChanged != null)

PropertyChanged(this, new PropertyChangedEventArgs(prop));

}

}

**EF:**

public class TicketContext : DbContext

{

public TicketContext() : base("DefaultConnection")

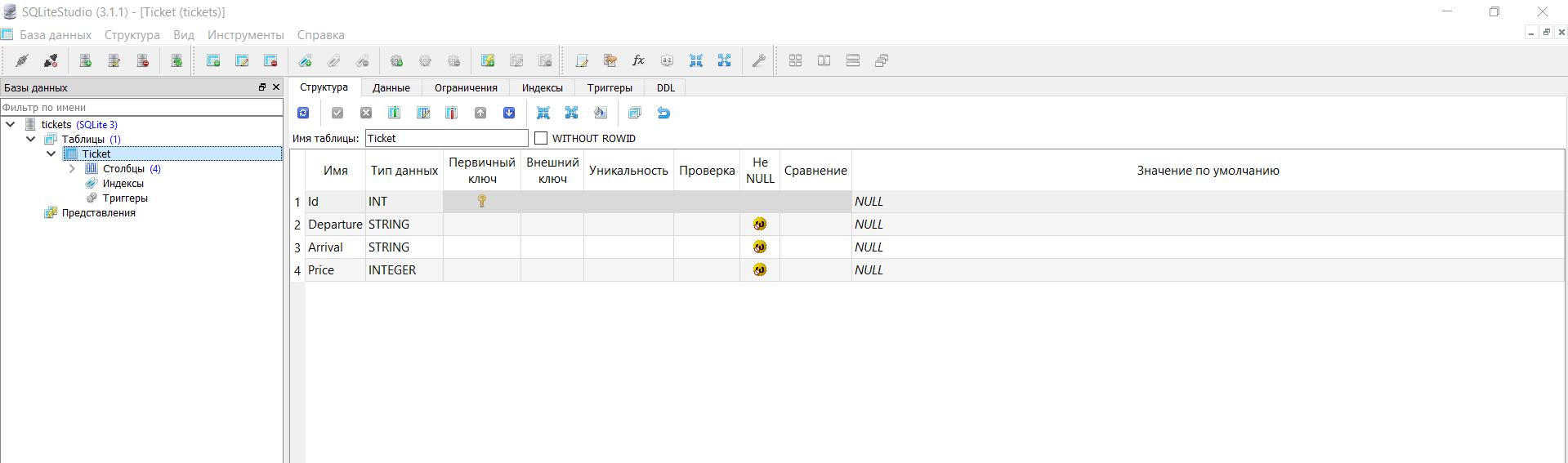
{

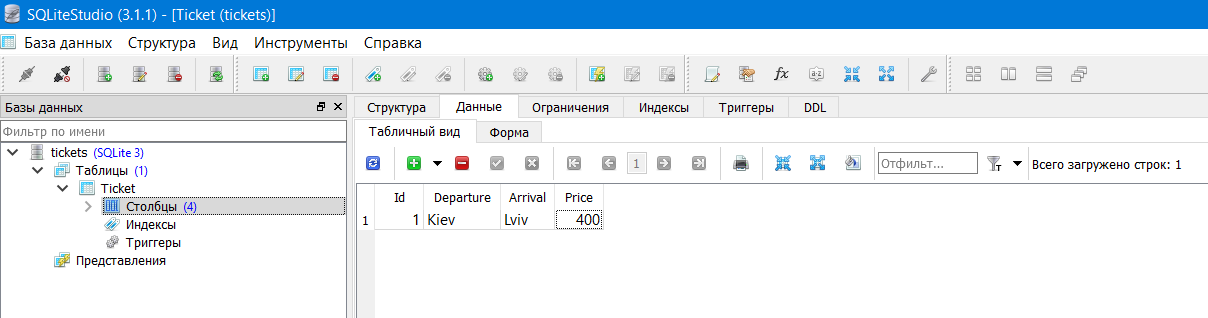
}

public DbSet<Ticket> Tickets { get; set; }

}

**SQLiteStudio база данных:**





**Добавление:**

private void button1\_Click(object sender, EventArgs e)

{

Ticket ticket = new Ticket()

{

Id = 1,

Arrival = textBox1.Text,

Departure = textBox2.Text,

Price = Convert.ToInt32(textBox3.Text)

};

db.Tickets.Add(ticket);

db.SaveChanges();

}

**Удаление:**

private void button2\_Click(object sender, EventArgs e)

{

Ticket ticket = db.Tickets.Where(t => t.Price == Convert.ToInt32(textBox3.Text)).ToList().First();

db.Tickets.Remove(ticket);

db.SaveChanges();

}

**Обновление:**

private void button1\_Click\_1(object sender, EventArgs e)

{

Ticket ticket = db.Tickets.Where(t => t.Id == Convert.ToInt32(textBox1.Text)).ToList().First();

ticket.Price = Convert.ToInt32(textBox2.Text);

ticket.Arrival = textBox3.Text;

ticket.Departure = textBox4.Text;

db.Entry(ticket).State = EntityState.Modified;

db.SaveChanges();

}

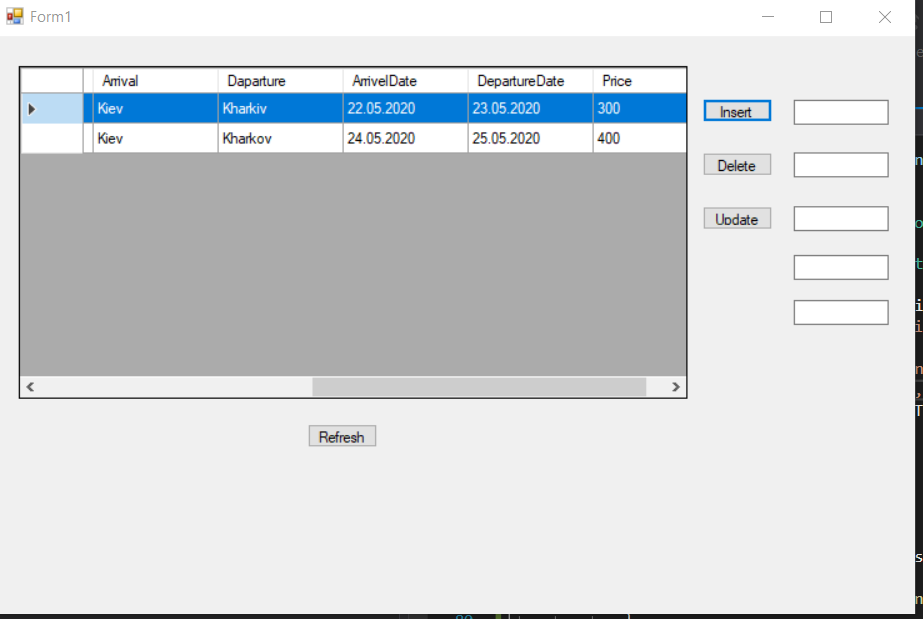
**Строка подлкючения:**

<connectionStrings>

<add name="DefaultConnection" connectionString="Data Source=.\tickets.db" providerName="System.Data.SQLite" />

</connectionStrings>

**Cкриншоты работы:**



**Вывод:** Изучил принципы работы SQLite.