ГУАП

КАФЕДРА № 43

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ст. преподаватель |  |  |  | Н.А. [Соловьева](http://rasp.guap.ru/?p=257) |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №10 |
| «ГРАФИКА В ПРИЛОЖЕНИЯХ» |
| по курсу: Конструирование информационных систем |
|  |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | 4831 |  |  |  | К.А. Корнющенков |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург 2020

**Задание**

Разработайте форму для отображения данных из одной таблицы, в которой одно из полей является именем (с указанием пути) графического файла, а значения ключевого атрибута формируются автоматически (см. рис. 2).

**Описание таблицы**

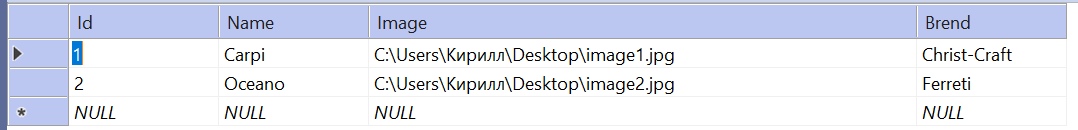
Id – id таблицы

Name – название модели

Image – путь до изображения

Brend – название Бренда

**Данные в таблице**

****

**Листинг кода**

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Data.SqlClient;

namespace WindowsFormsApp12

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

SqlConnection cnn = new SqlConnection("server=DESKTOP-0E0DMO2;Trusted\_Connection=yes;database=kirill;connection timeout=30");

DataSet ds = new DataSet();

BindingSource bind = new BindingSource();

SqlDataAdapter da = new SqlDataAdapter();

private void Form1\_Load(object sender, EventArgs e)

{

da.SelectCommand = new SqlCommand("select \* from ImageTable", cnn);

da.Fill(ds, "ImageTable");

da.DeleteCommand = new SqlCommand("delete from ImageTable where Id = @Id", cnn);

da.DeleteCommand.Parameters.Add("@Id", SqlDbType.Int, 4, "Id");

da.InsertCommand = new SqlCommand("insert into ImageTable (Id,Name,Image,Brend) values (@Id,@Name,@Image,@Brend)", cnn);

da.InsertCommand.Parameters.Add("@Id", SqlDbType.Int, 50, "Id");

da.InsertCommand.Parameters.Add("@Name", SqlDbType.NVarChar, 200, "Name");

da.InsertCommand.Parameters.Add("@Image", SqlDbType.NVarChar, 50, "Image");

da.InsertCommand.Parameters.Add("@Brend", SqlDbType.NVarChar, 200, "Brend");

bind.DataSource = ds.Tables["ImageTable"];

textBox1.DataBindings.Add(new Binding("Text", bind, "Id"));

textBox2.DataBindings.Add(new Binding("Text", bind, "Name"));

textBox3.DataBindings.Add(new Binding("Text", bind, "Image"));

textBox4.DataBindings.Add(new Binding("Text", bind, "Brend"));

pictureBox1.SizeMode = PictureBoxSizeMode.StretchImage;

pictureBox1.Image = Image.FromFile(textBox3.Text);

}

//add

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

{

DataRow newRow = ds.Tables[0].NewRow();

newRow["Id"] = textBox1.Text;

newRow["Name"] = textBox2.Text;

newRow["Image"] = textBox3.Text;

newRow["Brend"] = textBox4.Text;

ds.Tables[0].Rows.Add(newRow);

da.Update(ds.Tables[0]);

ds.Clear();

da.SelectCommand = new SqlCommand("select \* from ImageTable", cnn);

da.Fill(ds, "ImageTable");

}

//delete

private void button4\_Click(object sender, EventArgs e)

{

ds.Tables[0].AcceptChanges();

ds.Tables[0].Rows[bind.Position].Delete();

da.Update(ds.Tables[0]);

}

private void button3\_Click\_1(object sender, EventArgs e)

{

textBox1.Text = "";

textBox2.Text = "";

textBox3.Text = "";

textBox4.Text = "";

}

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

{

bind.Position -= 1;

if (textBox3.Text != "")

{

pictureBox1.SizeMode = PictureBoxSizeMode.StretchImage;

pictureBox1.Image = Image.FromFile(textBox3.Text);

}

}

private void button6\_Click(object sender, EventArgs e)

{

bind.Position += 1;

if (textBox3.Text != "")

{

pictureBox1.SizeMode = PictureBoxSizeMode.StretchImage;

pictureBox1.Image = Image.FromFile(textBox3.Text);

}

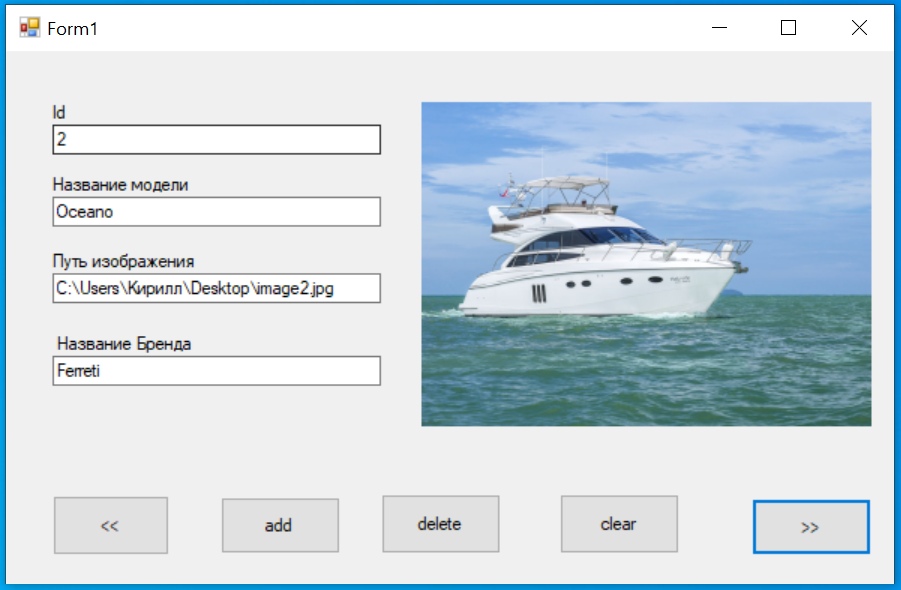
}

}

}

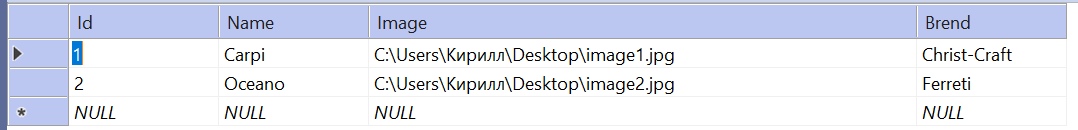
**Пример работы программы**

Перемещение при помощи кнопок << или >>

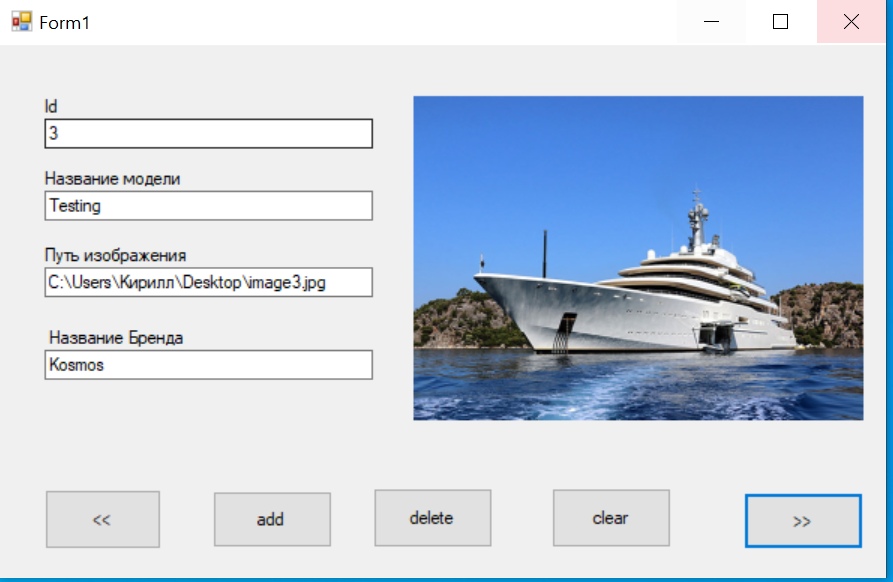
****

**Добавление данных**

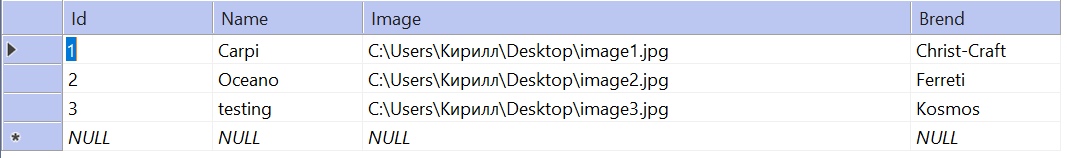
Данные до добавления

****

После добавления данные отображаются

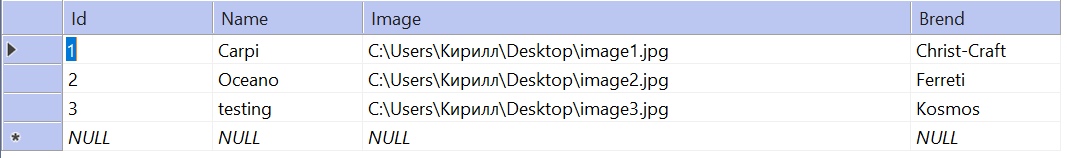


Данные после добавления

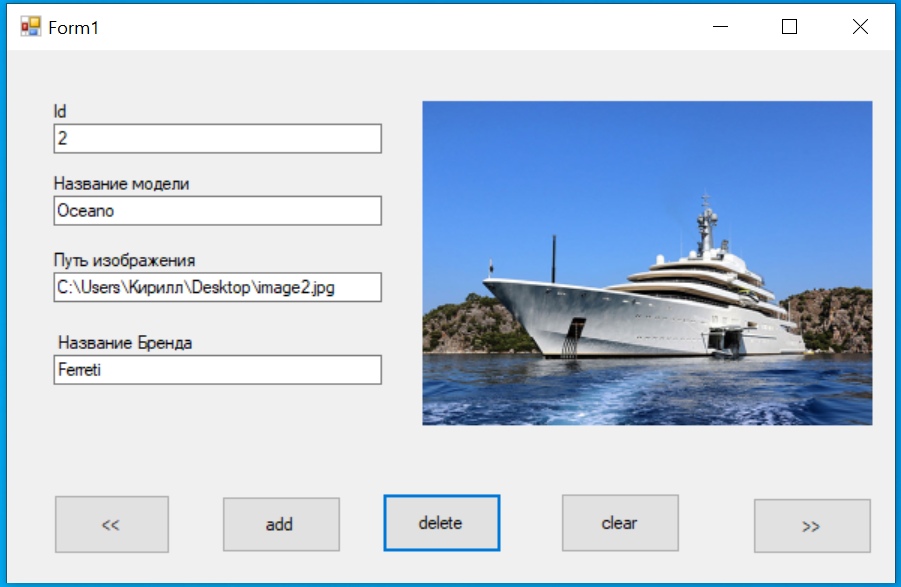
****

**Удаление Записей**

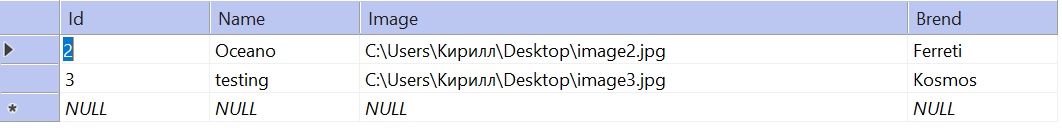
Данные до удаления

****

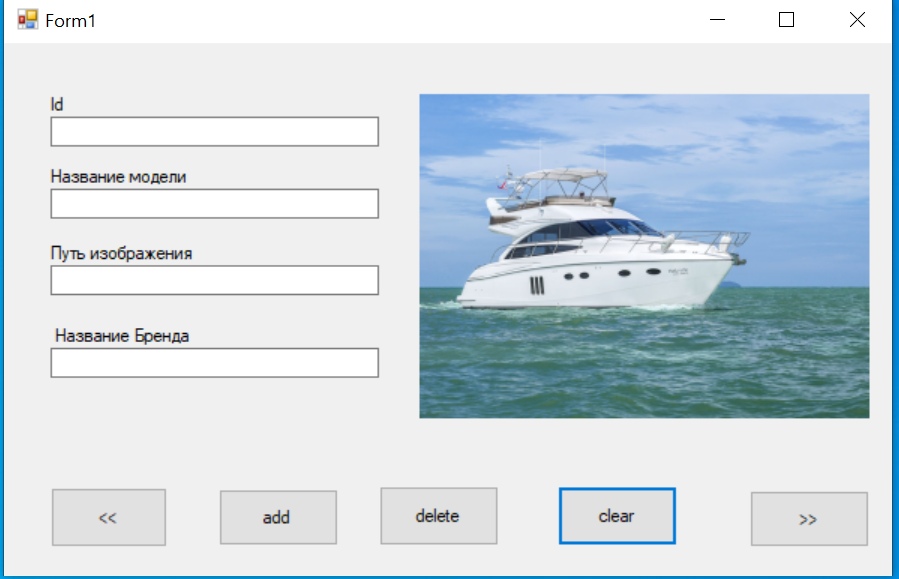
После удаления отображаются данные, которые следуют после удаленных



Данные после удаления



**Очистка полей**



**Вывод**

В ходе лабораторной работы был изучены способы применения графики в приложениях.