Файл «Users.cs»:

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;

namespace WindowsFormsApp1

{

public partial class Users : Form

{

public Users()

{

InitializeComponent();

}

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

{

// TODO: данная строка кода позволяет загрузить данные в таблицу "moneyDataSet.Users". При необходимости она может быть перемещена или удалена.

this.usersTableAdapter.Fill(this.moneyDataSet.Users);

// TODO: данная строка кода позволяет загрузить данные в таблицу "moneyDataSet.Users". При необходимости она может быть перемещена или удалена.

this.usersTableAdapter.Fill(this.moneyDataSet.Users);

// TODO: данная строка кода позволяет загрузить данные в таблицу "moneyDataSet.Users". При необходимости она может быть перемещена или удалена.

this.usersTableAdapter.Fill(this.moneyDataSet.Users);

}

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

{

this.Validate();

this.tableAdapterManager.UpdateAll(this.moneyDataSet);

}

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

{

this.Validate();

this.usersBindingSource.EndEdit();

this.tableAdapterManager.UpdateAll(this.moneyDataSet);

}

}

}

Файл «Reports.cs»:

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;

namespace WindowsFormsApp1

{

public partial class Reports : Form

{

int currentMonth = DateTime.Now.Month;

string command;

string report;

public Reports()

{

InitializeComponent();

}

public Main Main

{

get => default(Main);

set

{

}

}

public Main Main1

{

get => default(Main);

set

{

}

}

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

{

// TODO: данная строка кода позволяет загрузить данные в таблицу "moneyDataSet.Users". При необходимости она может быть перемещена или удалена.

this.usersTableAdapter.Fill(this.moneyDataSet.Users);

dateTimePicker1.CustomFormat = "MM/dd/yyyy";

dateTimePicker1.Format = DateTimePickerFormat.Custom;

dateTimePicker2.CustomFormat = "MM/dd/yyyy";

dateTimePicker2.Format = DateTimePickerFormat.Custom;

}

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

{

switch (comboBox1.SelectedIndex)

{

case 0: report0(DateTime.Now.Month,DateTime.Now.Year, "Расходы за текущий месяц"); break;

case 1: report0(DateTime.Now.Month-1, DateTime.Now.Year, "Расходы за прошлый месяц"); break;

case 2: report0(dateTimePicker1.Value.Month, dateTimePicker1.Value.Year, "Расходы за указанный месяц"); break;

case 3: report3(); break;

case 4: report4(); break;

case 5: report5(DateTime.Now.Month, DateTime.Now.Year, "Доходы за текущий месяц"); break;

case 6: report5(DateTime.Now.Month-1, DateTime.Now.Year, "Доходы за прошлый месяц"); break;

case 7: report5(dateTimePicker1.Value.Month, dateTimePicker1.Value.Year, "Доходы за период"); break;

case 8: report7(DateTime.Now.Month, DateTime.Now.Year, "Одолжили за текущий месяц"); break;

case 9: report7(DateTime.Now.Month - 1, DateTime.Now.Year, "Одолжили за прошлый месяц"); break;

case 10: report7(dateTimePicker1.Value.Month, dateTimePicker1.Value.Year, "Одолжили за период"); break;

case 11: report6(DateTime.Now.Month, DateTime.Now.Year, "Пользователь одолжил за текущий месяц"); break;

case 12: report6(DateTime.Now.Month - 1, DateTime.Now.Year, "Пользователь одолжил за прошлый месяц"); break;

case 13: report6(dateTimePicker1.Value.Month, dateTimePicker1.Value.Year, "Пользователь одолжил за период"); break;

}

}

//Расходы за текущий месяц

//Расходы за прошлый месяц

//Расходы за указанный месяц

private void report0(int monthInt,int year, string typeRep)

{

//MessageBox.Show($"Месяц {monthInt}, Год {year}");

string month = "MONTH(DateTime)=",

user = "";

if (monthInt == 0) {

monthInt = 12; year -= 1;

}

month += monthInt+" AND YEAR(DateTime)=" + year;

report = "<===== Отчет ====>\r\n" +

$"Тип отчета: {typeRep}\r\n";

Dictionary<string, float> dict = new Dictionary<string, float>();

if (!checkBox1.Checked)

{

user += " AND User=" + comboBox2.SelectedValue;

}

else

report += "По всем пользователям\r\n";

command = $"SELECT Money.id as [№], Category.name as [Категория], SubCategory.name as [Подкатегория], Users.name as [Пользователь], Money.DateTime as [Дата], Money.Cost as [Стоимость], Money.Comment as [Комментарий] FROM Users INNER JOIN((Category INNER JOIN [Money] ON Category.Id = Money.Category) INNER JOIN SubCategory ON(SubCategory.id = Money.SubCategory) AND(Category.Id = SubCategory.categoryId)) ON Users.id = Money.User WHERE(([Category].[Id] =[Money].[Category]) AND([SubCategory].[id] =[Money].[SubCategory]) AND([Users].[id] =[Money].[User]) AND ({month} {user}) );";

Main.DataBase.SelectGridPlus(command, dataGridView1);

string categ;

float sum, all = 0;

try

{

for (int i = 0; i < dataGridView1.RowCount - 1; i++)

{

categ = dataGridView1[1, i].Value.ToString();

sum = float.Parse(dataGridView1[5, i].Value.ToString());

if (dict.ContainsKey(categ)) dict[categ] = dict[categ] + sum;

else

dict.Add(categ, sum);

}

report += $"Отчет за {currentMonth} {DateTime.Now.Year}\r\n" +

$"Дата/время создания отчета: {DateTime.Now}\r\n\r\n";

foreach (string cat in dict.Keys)

{

report += $"{cat}: {dict[cat]}\r\n";

all += dict[cat];

}

report += "\r\nИТОГО: " + all;

textBox1.Text = report;

}

catch (Exception ex) { }

}

//Расходы за интервал времени

private void report3()

{

string month = "", user = "";

month = $" (DateTime BETWEEN #{dateTimePicker1.Text}# AND #{dateTimePicker2.Text}#) ";

report = "<===== Отчет ====>\r\n" +

"Тип отчета: Расходы за интервал времени\r\n";

Dictionary<string, float> dict = new Dictionary<string, float>();

if (!checkBox1.Checked)

{

user += " AND User=" + comboBox2.SelectedValue;

}

else

report += "По всем пользователям\r\n";

command = $"SELECT Money.id as [№], Category.name as [Категория], SubCategory.name as [Подкатегория], Users.name as [Пользователь], Money.DateTime as [Дата], Money.Cost as [Стоимость], Money.Comment as [Комментарий] FROM Users INNER JOIN((Category INNER JOIN [Money] ON Category.Id = Money.Category) INNER JOIN SubCategory ON(SubCategory.id = Money.SubCategory) AND(Category.Id = SubCategory.categoryId)) ON Users.id = Money.User WHERE(([Category].[Id] =[Money].[Category]) AND([SubCategory].[id] =[Money].[SubCategory]) AND([Users].[id] =[Money].[User]) AND ({month} {user}) );";

Console.WriteLine(command);

Main.DataBase.SelectGridPlus(command, dataGridView1);

string categ;

float sum, all = 0;

try

{

for (int i = 0; i < dataGridView1.RowCount - 1; i++)

{

categ = dataGridView1[1, i].Value.ToString();

sum = float.Parse(dataGridView1[5, i].Value.ToString());

if (dict.ContainsKey(categ)) dict[categ] = dict[categ] + sum;

else

dict.Add(categ, sum);

}

report += $"Отчет за {currentMonth} {DateTime.Now.Year}\r\n" +

$"Дата/время создания отчета: {DateTime.Now}\r\n\r\n";

foreach (string cat in dict.Keys)

{

report += $"{cat}: {dict[cat]}\r\n";

all += dict[cat];

}

report += "\r\nИТОГО: " + all;

textBox1.Text = report;

}

catch (Exception ex) { }

}

//Кто сколько потратил за интервал

private void report4()

{

string month = "", user = "";

month = $" (DateTime BETWEEN #{dateTimePicker1.Text}# AND #{dateTimePicker2.Text}#) ";

report = "<===== Отчет ====>\r\n" +

"Тип отчета: Кто сколько потратил за интервал\r\n";

Dictionary<string, float> dict = new Dictionary<string, float>();

if (!checkBox1.Checked)

{

user += " AND User=" + comboBox2.SelectedValue;

}

else

report += "По всем пользователям\r\n";

command = $"SELECT Money.id as [№], Category.name as [Категория], SubCategory.name as [Подкатегория], Users.name as [Пользователь], Money.DateTime as [Дата], Money.Cost as [Стоимость], Money.Comment as [Комментарий] FROM Users INNER JOIN((Category INNER JOIN [Money] ON Category.Id = Money.Category) INNER JOIN SubCategory ON(SubCategory.id = Money.SubCategory) AND(Category.Id = SubCategory.categoryId)) ON Users.id = Money.User WHERE(([Category].[Id] =[Money].[Category]) AND([SubCategory].[id] =[Money].[SubCategory]) AND([Users].[id] =[Money].[User]) AND ({month} {user}) );";

Console.WriteLine(command);

Main.DataBase.SelectGridPlus(command, dataGridView1);

string categ;

float sum, all = 0;

try

{

for (int i = 0; i < dataGridView1.RowCount - 1; i++)

{

categ = dataGridView1[3, i].Value.ToString();

sum = float.Parse(dataGridView1[5, i].Value.ToString());

if (dict.ContainsKey(categ)) dict[categ] = dict[categ] + sum;

else

dict.Add(categ, sum);

}

report += $"Отчет за {currentMonth} {DateTime.Now.Year}\r\n" +

$"Дата/время создания отчета: {DateTime.Now}\r\n\r\n";

foreach (string cat in dict.Keys)

{

report += $"{cat}: {dict[cat]}\r\n";

all += dict[cat];

}

report += "\r\nИТОГО: " + all;

textBox1.Text = report;

}

catch (Exception ex) { }

}

private void report5(int monthInt, int year, string typeRep)

{

string month = "MONTH(dates)=",

user = "";

if (monthInt == 0)

{

monthInt = 12; year -= 1;

}

month += monthInt + " AND YEAR(dates)=" + year;

report = "<===== Отчет ====>\r\n" +

$"Тип отчета: {typeRep}\r\n";

Dictionary<string, float> dict = new Dictionary<string, float>();

if (!checkBox1.Checked)

{

user += " AND userId=" + comboBox2.SelectedValue;

}

else

report += "По всем пользователям\r\n";

command = $"SELECT [count] as [Количество], [dates] as [Дата], [comment] as [Комментарий], [names] as [Тип заработка] from [Dohod] INNER JOIN [DohodTypes] ON [DohodTypes].[id] = [Dohod].[typesId] where {month} {user};";

Main.DataBase.SelectGridPlus(command, dataGridView1);

string names;

try

{

report += $"Отчет за {currentMonth} {DateTime.Now.Year}\r\n" +

$"Дата/время создания отчета: {DateTime.Now}\r\n\r\n";

for (int i = 0; i < dataGridView1.RowCount - 1; i++)

{

names = dataGridView1[3, i].Value.ToString();

report += dataGridView1[0, i].Value.ToString();

report += " - "+names+"\r\n";

}

textBox1.Text = report;

}

catch (Exception ex) { }

}

private void report6(int monthInt, int year, string typeRep)

{

string month = "MONTH(datePol)=",

user = "";

if (monthInt == 0)

{

monthInt = 12; year -= 1;

}

month += monthInt + " AND YEAR(datePol)=" + year;

report = "<===== Отчет ====>\r\n" +

$"Тип отчета: {typeRep}\r\n";

if (!checkBox1.Checked)

{

user += " AND userId=" + comboBox2.SelectedValue;

}

else

report += "По всем пользователям\r\n";

command = $"SELECT [kto],[datePol],[isOplachen],[count],[name] from [Borrow] inner join [Users] on [Borrow].[userId]=[Users].[id] where {month} {user};";

Main.DataBase.SelectGridPlus(command, dataGridView1);

string names;

try

{

report += $"Отчет за {currentMonth} {DateTime.Now.Year}\r\n" +

$"Дата/время создания отчета: {DateTime.Now}\r\n\r\n";

for (int i = 0; i < dataGridView1.RowCount - 1; i++)

{

names = "Кому должен вернуть: "+ dataGridView1[0, i].Value.ToString()+", Кто: "+ dataGridView1[4, i].Value.ToString();

report +=names+ ", Сколько: " + dataGridView1[3, i].Value.ToString() +" , Вернул ли:"+ dataGridView1[2, i].Value.ToString() + "\r\n";

}

textBox1.Text = report;

}

catch (Exception ex) { }

}

private void report7(int monthInt, int year, string typeRep)

{

string month = "MONTH(dates)=",

user = "";

if (monthInt == 0)

{

monthInt = 12; year -= 1;

}

month += monthInt + " AND YEAR(dates)=" + year;

report = "<===== Отчет ====>\r\n" +

$"Тип отчета: {typeRep}\r\n";

Dictionary<string, float> dict = new Dictionary<string, float>();

if (!checkBox1.Checked)

{

user += " AND userId=" + comboBox2.SelectedValue;

}

else

report += "По всем пользователям\r\n";

command = $"SELECT \* from Lend where {month} {user};";

Main.DataBase.SelectGridPlus(command, dataGridView1);

string names;

try

{

report += $"Отчет за {currentMonth} {DateTime.Now.Year}\r\n" +

$"Дата/время создания отчета: {DateTime.Now}\r\n\r\n";

for (int i = 0; i < dataGridView1.RowCount - 1; i++)

{

names = "Кто должен вернуть: " + dataGridView1[2, i].Value.ToString();

report += names + ", Сколько: " + dataGridView1[6, i].Value.ToString() + " , Вернул ли:" + dataGridView1[4, i].Value.ToString() + "\r\n";

}

textBox1.Text = report;

}

catch (Exception ex) { }

}

}

}

Файл «PlansOperation.cs»:

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;

namespace WindowsFormsApp1

{

public partial class PlansOperation : Form

{

int id;

string command;

public PlansOperation(int id)

{

InitializeComponent();

this.id = id;

}

public Plans Plans

{

get => default(Plans);

set

{

}

}

public Money Money

{

get => default(Money);

set

{

}

}

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

{

// TODO: данная строка кода позволяет загрузить данные в таблицу "moneyDataSet.Category". При необходимости она может быть перемещена или удалена.

this.categoryTableAdapter.Fill(this.moneyDataSet.Category);

if (id >0) loadData();

}

private void loadData() {

command = "SELECT \* FROM [Plan] WHERE id="+id;

Dictionary<int,string> dict = Main.DataBase.getDictionary(command,7);

//comboBox1.SelectedValue = Main.DataBase.ReturnString($"SELECT name FROM [Category] WHERE id={dict[1]}");

comboBox1.SelectedValue = dict[1];

textBox1.Text = dict[2];

textBox3.Text = dict[3];

textBox2.Text = dict[5];

dateTimePicker1.Text = dict[6];

}

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

{

if (textBox1.Text.Length < 1) return;

if (textBox2.Text.Length < 1) return;

int think = int.Parse(textBox1.Text);

int pay = int.Parse(textBox3.Text);

if (id < 1)

command = $"INSERT INTO [Plan] ([Categoty], [Think],[Pay], [Rasn],[Comment], [DateTime]) VALUES({comboBox1.SelectedValue},{think},{pay},{think - pay},\"{textBox2.Text}\",\"{dateTimePicker1.Text}\") ";

else command = $"UPDATE [Plan] SET [Categoty]={comboBox1.SelectedValue}, [Think]={think},[Pay]={pay}, [Rasn]={think - pay},[Comment]=\"{textBox2.Text}\", [DateTime]=\"{dateTimePicker1.Text}\" WHERE id={id}";

Console.WriteLine(command);

try

{

Main.DataBase.SqlCommand(command);

Main.plans.loadData();

}

catch (Exception ex) { }

}

private void textBox1\_TextChanged(object sender, EventArgs e)

{

}

private void textBox1\_KeyPress(object sender, KeyPressEventArgs e)

{

if (!(Char.IsDigit(e.KeyChar)) && !((e.KeyChar == '.') &&

(((TextBox)sender).Text.IndexOf(".") == -1) &&

(((TextBox)sender).Text.Length != 0)))

{

if (e.KeyChar != (char)Keys.Back)

{

e.Handled = true;

}

}

}

}

}

Файл «Plans.cs»:

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;

namespace WindowsFormsApp1

{

public partial class Plans : Form

{

string command;

public Plans()

{

InitializeComponent();

}

public Main Main

{

get => default(Main);

set

{

}

}

public Main Main1

{

get => default(Main);

set

{

}

}

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

{

loadData();

domainUpDown1.Text = DateTime.Now.Month + "";

domainUpDown2.Text = DateTime.Now.Year + "";

}

// добавить

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

{

new PlansOperation(-1).Show();

}

//изменить

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

{

try

{

string id = dataGridView1[0, dataGridView1.CurrentRow.Index].Value.ToString();

new PlansOperation(int.Parse(id)).Show();

}

catch (Exception ex) { MessageBox.Show("Нечего редактировать"); };

}

//удалить

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

{

// если нажали ок

if (MessageBox.Show("Действительно удалить ?", "Подтвердите удаление", MessageBoxButtons.YesNo) == DialogResult.Yes)

{

// смотрим какой телефон удалить

int id = int.Parse(dataGridView1[0, dataGridView1.SelectedCells[0].RowIndex].Value.ToString());

// команду удалить

string command = "DELETE FROM [Plan] WHERE id = " + id;

// выполнить удаление

Main.DataBase.SqlCommand(command);

loadData();

}

}

public void loadData(){

command = $"SELECT Plan.id as [№], Category.name as [Категория], Plan.Think as [Предполагалось], Plan.Pay as [Фактичски], Plan.Rasn as [Разница], Plan.Comment as [Комментарий], Plan.DateTime as [Дата] FROM Category INNER JOIN Plan ON Category.Id = Plan.Categoty WHERE(([Plan].[Categoty] =[Category].[Id]) AND (YEAR(DateTime)={domainUpDown2.Text} AND MONTH(DateTime)={domainUpDown1.Text}) ); ";

Main.DataBase.SelectGridPlus(command, dataGridView1);

}

private void domainUpDown1\_SelectedItemChanged(object sender, EventArgs e)

{

loadData();

}

}

}

Файл «MoneyOperation.cs»:

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;

namespace WindowsFormsApp1

{

public partial class PlanOperation : Form

{

int id;

// -1 значит добавить

// остальные значить изменить

public PlanOperation(int id)

{

InitializeComponent();

this.id = id;

}

public Money Money

{

get => default(Money);

set

{

}

}

public Plans Plans

{

get => default(Plans);

set

{

}

}

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

{

// TODO: данная строка кода позволяет загрузить данные в таблицу "moneyDataSet.Users". При необходимости она может быть перемещена или удалена.

this.usersTableAdapter.Fill(this.moneyDataSet.Users);

// TODO: данная строка кода позволяет загрузить данные в таблицу "moneyDataSet.Category". При необходимости она может быть перемещена или удалена.

this.categoryTableAdapter.Fill(this.moneyDataSet.Category);

comboBox2.Items.Clear();

Main.DataBase.LoadBox($"SELECT \* FROM SubCategory WHERE categoryId in (Select id from Category Where Category.name=\"{comboBox1.Text}\")", "name", comboBox2);

if (id > 0) loadData();

}

private void loadData() {

Dictionary<int, string> dict = Main.DataBase.getDictionary($"Select \* from [Money] Where id="+id,7);

//MessageBox.Show(Main.DataBase.ReturnString($"SELECT name FROM Category WHERE id={dict[1]}"));

comboBox1.Text = Main.DataBase.ReturnString($"SELECT name FROM Category WHERE id={dict[1]}");

comboBox2.Text = Main.DataBase.ReturnString($"SELECT name FROM SubCategory WHERE id={dict[2]}");

comboBox3.Text = Main.DataBase.ReturnString($"SELECT name FROM Users WHERE id={dict[3]}");

textBox1.Text = dict[5];

dateTimePicker1.Text = dict[4];

textBox2.Text = dict[6];

}

private void comboBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

comboBox2.Items.Clear();

Main.DataBase.LoadBox($"SELECT \* FROM SubCategory WHERE categoryId in (Select id from Category Where Category.name=\"{comboBox1.Text}\")", "name", comboBox2);

}

// кнопка хорошо

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

{

//проверка на пустые значения

if (comboBox1.Text.Length < 1) return;

if (comboBox2.Text.Length < 1) return;

if (comboBox2.Text.Length < 1) return;

if (textBox1.Text.Length < 1) return;

string command, category, subCategory, user;

category = Main.DataBase.ReturnString($"SELECT id FROM Category WHERE name=\"{comboBox1.Text}\"");

subCategory = Main.DataBase.ReturnString($"SELECT id FROM SubCategory WHERE name=\"{comboBox2.Text}\"");

user = Main.DataBase.ReturnString($"SELECT \* FROM Users WHERE name=\"{comboBox3.Text}\"");

if (id == -1)

command = $"INSERT INTO [Money] ( [Category], [SubCategory], [User], [DateTime], [Cost], [Comment] ) VALUES ({category},{subCategory},{user},\"{dateTimePicker1.Text}\", {textBox1.Text},\"{textBox2.Text}\")";

else

command = $"UPDATE [Money] SET [Category]={category}, [SubCategory]={subCategory}, [User]={user}, [DateTime]=\"{dateTimePicker1.Text}\", [Cost]={textBox1.Text}, [Comment]=\"{textBox2.Text}\" WHERE [id]={id}";

//MessageBox.Show(command);

Console.WriteLine(command);

try

{

Main.DataBase.SqlCommand(command);

Main.money.loadData();

}

catch (Exception ex) { }

}

private void textBox1\_KeyPress(object sender, KeyPressEventArgs e)

{

if (!(Char.IsDigit(e.KeyChar)) && !((e.KeyChar == '.') &&

(((TextBox)sender).Text.IndexOf(".") == -1) &&

(((TextBox)sender).Text.Length != 0)))

{

if (e.KeyChar != (char)Keys.Back)

{

e.Handled = true;

}

}

}

}

}

Файл «Money.cs»:

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;

namespace WindowsFormsApp1

{

public partial class Money : Form

{

public Money()

{

InitializeComponent();

}

public Main Main

{

get => default(Main);

set

{

}

}

public Main Main1

{

get => default(Main);

set

{

}

}

private void dataGridView1\_CellContentClick(object sender, DataGridViewCellEventArgs e)

{

}

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

{

// TODO: данная строка кода позволяет загрузить данные в таблицу "moneyDataSet.Users". При необходимости она может быть перемещена или удалена.

this.usersTableAdapter.Fill(this.moneyDataSet.Users);

loadData();

domainUpDown1.Text = DateTime.Now.Month + "";

domainUpDown2.Text = DateTime.Now.Year + "";

}

public void loadData()

{

try

{

string showUsers;

if (checkBox1.Checked)

showUsers = "";

else

showUsers = $" AND User= {comboBox1.SelectedValue.ToString()}";

//showUsers = $" AND User= {Main.DataBase.ReturnString($"SELECT \* FROM Users WHERE name=\"{comboBox1.Text}\"")} ";

string command = $"SELECT Money.id as [№], Category.name as [Категория], SubCategory.name as [Подкатегория], Users.name as [Пользователь], Money.DateTime as [Дата], Money.Cost as [Стоимость], Money.Comment as [Комментарий] FROM Users INNER JOIN((Category INNER JOIN [Money] ON Category.Id = Money.Category) INNER JOIN SubCategory ON(SubCategory.id = Money.SubCategory) AND(Category.Id = SubCategory.categoryId)) ON Users.id = Money.User WHERE(([Category].[Id] =[Money].[Category]) AND([SubCategory].[id] =[Money].[SubCategory]) AND([Users].[id] =[Money].[User]) AND (YEAR(DateTime)={domainUpDown2.Text} AND MONTH(DateTime)={domainUpDown1.Text} {showUsers}) );";

Main.DataBase.SelectGridPlus(command, dataGridView1);

}

catch (Exception ex) { }

}

// кнопка добавить

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

{

new PlanOperation(-1).Show();

}

//кнопка изменить

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

{

try

{

string id = dataGridView1[0, dataGridView1.CurrentRow.Index].Value.ToString();

new PlanOperation(int.Parse(id)).Show();

}

catch (Exception ex) { MessageBox.Show("Нечего редактировать"); };

}

//кнопка удалить

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

{

// если нажали ок

if (MessageBox.Show("Действительно удалить ?", "Подтвердите удаление", MessageBoxButtons.YesNo) == DialogResult.Yes)

{

// смотрим какой телефон удалить

int id = int.Parse(dataGridView1[0, dataGridView1.SelectedCells[0].RowIndex].Value.ToString());

// команду удалить

string command = "DELETE FROM [Money] WHERE id = " + id;

// выполнить удаление

Main.DataBase.SqlCommand(command);

loadData();

}

}

private void domainUpDown1\_SelectedItemChanged(object sender, EventArgs e)

{

loadData();

}

}

}

Файл «Lend.cs»:

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 WindowsFormsApp1.MoneyDataSetTableAdapters;

namespace WindowsFormsApp1

{

public partial class Lend : Form

{

private LendTableAdapter lendTableAdapter;

private BindingSource lendBindingSource;

private MoneyDataSet moneyDataSet;

private DataGridViewRow row;

bool isNew = false;

public Lend(LendTableAdapter lendTableAdapter, BindingSource lendBindingSource, MoneyDataSet moneyDataSet, DataGridViewRow row)

{

InitializeComponent();

this.lendTableAdapter = lendTableAdapter;

this.lendBindingSource = lendBindingSource;

this.moneyDataSet = moneyDataSet;

this.row = row;

usersBindingSource.DataSource = moneyDataSet;

}

public Dolg Dolg

{

get => default(Dolg);

set

{

}

}

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

{

// TODO: данная строка кода позволяет загрузить данные в таблицу "moneyDataSet1.Users". При необходимости она может быть перемещена или удалена.

this.usersTableAdapter.Fill(this.moneyDataSet1.Users);

// TODO: данная строка кода позволяет загрузить данные в таблицу "moneyDataSet1.Users". При необходимости она может быть перемещена или удалена.

this.usersTableAdapter.Fill(this.moneyDataSet1.Users);

// TODO: данная строка кода позволяет загрузить данные в таблицу "moneyDataSet1.Users". При необходимости она может быть перемещена или удалена.

//this.usersTableAdapter.Fill(this.moneyDataSet1.Users);

// TODO: данная строка кода позволяет загрузить данные в таблицу "moneyDataSet1.Users". При необходимости она может быть перемещена или удалена.

this.usersTableAdapter.Fill(this.moneyDataSet.Users);

if (row == null) isNew = true;

else {

comboBox1.SelectedValue = int.Parse(row.Cells[1].Value.ToString());

textBox1.Text = row.Cells[3].Value.ToString();

dateTimePicker1.Text = row.Cells[4].Value.ToString();

checkBox1.Checked = bool.Parse(row.Cells[5].Value.ToString());

textBox2.Text = row.Cells[6].Value.ToString();

textBox3.Text = row.Cells[7].Value.ToString();

}

}

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

{

try {

if(textBox1.Text.Length>0)

if (isNew) insert(); else update();

}

catch (Exception ex) { }

}

private void insert()

{

DataRowView row = (DataRowView)lendBindingSource.AddNew();

row[1] = comboBox1.SelectedValue;

row[2] = textBox1.Text;

row[3] = dateTimePicker1.Value;

row[4] = checkBox1.Checked;

row[5] = textBox2.Text;

row[6] = textBox3.Text;

lendBindingSource.EndEdit();

this.lendTableAdapter.Update(moneyDataSet);

MessageBox.Show("Сохранено");

this.Close();

}

private void update()

{

row.Cells[1].Value = comboBox1.SelectedValue;

row.Cells[2].Value = comboBox1.Text;

row.Cells[3].Value = textBox1.Text;

row.Cells[4].Value = dateTimePicker1.Value;

row.Cells[5].Value = checkBox1.Checked;

row.Cells[6].Value = textBox2.Text;

row.Cells[7].Value = textBox3.Text;

lendBindingSource.EndEdit();

this.lendTableAdapter.Update(((DataRowView)row.DataBoundItem).Row);

this.lendTableAdapter.Update(moneyDataSet);

MessageBox.Show("Сохранено");

this.Close();

}

}

}

Файл «Help.cs»:

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;

namespace WindowsFormsApp1

{

public partial class Help : Form

{

public Help()

{

InitializeComponent();

}

public Main Main

{

get => default(Main);

set

{

}

}

public Main Main1

{

get => default(Main);

set

{

}

}

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

{

// TODO: данная строка кода позволяет загрузить данные в таблицу "moneyDataSet.DohodTypes". При необходимости она может быть перемещена или удалена.

this.dohodTypesTableAdapter.Fill(this.moneyDataSet.DohodTypes);

// TODO: данная строка кода позволяет загрузить данные в таблицу "moneyDataSet.SubCategory". При необходимости она может быть перемещена или удалена.

this.subCategoryTableAdapter.Fill(this.moneyDataSet.SubCategory);

// TODO: данная строка кода позволяет загрузить данные в таблицу "moneyDataSet.Category". При необходимости она может быть перемещена или удалена.

this.categoryTableAdapter.Fill(this.moneyDataSet.Category);

}

// обновить 2 таблицу

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

{

try

{

this.subCategoryTableAdapter.Update(this.moneyDataSet);

}

catch (Exception ex) { MessageBox.Show("Вы не сохранили категорию"); }

}

//удалить в 2 таблице

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

{

try

{

this.categorySubCategoryBindingSource.RemoveCurrent();

}

catch (Exception ex) { MessageBox.Show("Нечего удалять"); }

}

//удалить в первой таблице

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

{

try

{

this.categoryBindingSource.RemoveCurrent();

}

catch (Exception ex) { MessageBox.Show("Нечего удалять"); }

}

//сохранить в первой таблице

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

{

try

{

this.categoryTableAdapter.Update(this.moneyDataSet);

this.categoryTableAdapter.Fill(this.moneyDataSet.Category);

}

catch (Exception ex) { MessageBox.Show("Ошибка сохранения категории. Проверьте данные"); }

}

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

{

dohodTypesTableAdapter.Update(this.moneyDataSet);

}

}

}

Файл «Dolg.cs»:

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;

namespace WindowsFormsApp1

{

public partial class Dolg : Form

{

public Dolg()

{

InitializeComponent();

}

public Main Main

{

get => default(Main);

set

{

}

}

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

{

this.usersTableAdapter.Fill(this.moneyDataSet.Users);

this.borrowTableAdapter.Fill(this.moneyDataSet.Borrow);

this.lendTableAdapter.Fill(this.moneyDataSet.Lend);

find();

}

//add lend

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

{

new Lend(lendTableAdapter, lendBindingSource, moneyDataSet, null).Show();

}

//update lend

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

{

//dataGridView1.BeginEdit(true);

if(dataGridView1.RowCount>0 && dataGridView1.CurrentRow!=null)

new Lend(lendTableAdapter, lendBindingSource, moneyDataSet, dataGridView1.CurrentRow).Show();

}

//delete lend

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

{

try

{

if (dataGridView1.RowCount > 0 && dataGridView1.SelectedCells.Count > 0)

if ((DialogResult = MessageBox.Show("Удалить выбранное?", "Удаление", MessageBoxButtons.YesNo)) == DialogResult.Yes)

{

lendBindingSource.RemoveAt(dataGridView1.CurrentRow.Index);

lendTableAdapter.Update(moneyDataSet.Lend);

}

}

catch (Exception ex) { MessageBox.Show("Ошибка. Возможно вы пытаетесь удалить используемый объект"); }

}

//add borrow

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

{

new Borrow(borrowTableAdapter, borrowBindingSource,null, moneyDataSet).Show();

}

//update borrow

private void button5\_Click(object sender, EventArgs e)

{

//dataGridView2.BeginEdit(true);

if (dataGridView2.RowCount > 0 && dataGridView2.CurrentRow != null)

new Borrow(borrowTableAdapter, borrowBindingSource, dataGridView2.CurrentRow, moneyDataSet).Show();

}

//delete borrow

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

{

try

{

if (dataGridView2.RowCount > 0 && dataGridView2.SelectedCells.Count > 0)

if ((DialogResult = MessageBox.Show("Удалить выбранное?", "Удаление", MessageBoxButtons.YesNo)) == DialogResult.Yes)

{

borrowBindingSource.RemoveAt(dataGridView2.CurrentRow.Index);

borrowTableAdapter.Update(moneyDataSet.Borrow);

}

}

catch (Exception ex) { MessageBox.Show("Ошибка. Возможно вы пытаетесь удалить используемый объект"); }

}

private void comboBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

find();

}

private void checkBox1\_CheckedChanged(object sender, EventArgs e)

{

find();

}

private void find() {

try

{

lendTableAdapter.FillByUserIdAndIsReturn(moneyDataSet.Lend, (int)comboBox1.SelectedValue,!checkBox1.Checked);

borrowTableAdapter.FillByUserIdAndIsOplachen(moneyDataSet.Borrow, (int)comboBox1.SelectedValue, !checkBox2.Checked);

for (int i = 0; i < dataGridView1.RowCount; i++)

{

int index1 = usersBindingSource.Find("id", dataGridView1[1, i].Value.ToString());

dataGridView1[2, i].Value = ((DataRowView)(usersBindingSource[index1]))[1];

}

for (int i = 0; i < dataGridView2.RowCount; i++)

{

int index1 = usersBindingSource.Find("id", dataGridView2[1, i].Value.ToString());

dataGridView2[2, i].Value = ((DataRowView)(usersBindingSource[index1]))[1];

}

}

catch (Exception ex) { }

}

}

}

Файл «DohodOperation.cs»:

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 WindowsFormsApp1.MoneyDataSetTableAdapters;

namespace WindowsFormsApp1

{

public partial class DohodOperation : Form

{

private DohodTableAdapter dohodTableAdapter;

private BindingSource dohodBindingSource;

private MoneyDataSet moneyDataSet;

private DataGridViewRow row;

bool isNew = true;

public DohodOperation(DohodTableAdapter dohodTableAdapter, BindingSource dohodBindingSource, MoneyDataSet moneyDataSet, DataGridViewRow currentRow)

{

InitializeComponent();

this.dohodTableAdapter = dohodTableAdapter;

this.dohodBindingSource = dohodBindingSource;

this.moneyDataSet = moneyDataSet;

this.row = currentRow;

isNew = row == null;

}

public Dohod Dohod

{

get => default(Dohod);

set

{

}

}

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

{

// TODO: данная строка кода позволяет загрузить данные в таблицу "moneyDataSet1.Users". При необходимости она может быть перемещена или удалена.

this.usersTableAdapter.Fill(this.moneyDataSet1.Users);

// TODO: данная строка кода позволяет загрузить данные в таблицу "moneyDataSet1.DohodTypes". При необходимости она может быть перемещена или удалена.

this.dohodTypesTableAdapter.Fill(this.moneyDataSet1.DohodTypes);

if (!isNew)

{

comboBox3.SelectedValue = int.Parse(row.Cells[1].Value.ToString());

comboBox1.SelectedValue = int.Parse(row.Cells[2].Value.ToString());

textBox1.Text = row.Cells[3].Value.ToString();

dateTimePicker1.Text = row.Cells[4].Value.ToString();

textBox2.Text = row.Cells[5].Value.ToString();

}

}

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

{

try

{

if (textBox1.Text.Length > 0)

if (isNew) insert(); else update();

}

catch (Exception ex) { }

}

private void insert()

{

DataRowView row = (DataRowView)dohodBindingSource.AddNew();

row[1] = comboBox3.SelectedValue;

row[2] = comboBox1.SelectedValue;

row[3] = textBox1.Text;

row[4] = dateTimePicker1.Value;

row[5] = textBox2.Text;

dohodBindingSource.EndEdit();

this.dohodTableAdapter.Update(moneyDataSet);

MessageBox.Show("Сохранено");

this.Close();

}

private void update()

{

row.Cells[1].Value = comboBox3.SelectedValue;

row.Cells[2].Value = comboBox1.SelectedValue;

row.Cells[3].Value = textBox1.Text;

row.Cells[4].Value = dateTimePicker1.Value;

row.Cells[5].Value = textBox2.Text;

dohodBindingSource.EndEdit();

this.dohodTableAdapter.Update(((DataRowView)row.DataBoundItem).Row);

MessageBox.Show("Сохранено");

this.Close();

}

}

}

Файл «Dohod.cs»:

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;

namespace WindowsFormsApp1

{

public partial class Dohod : Form

{

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

{

// TODO: данная строка кода позволяет загрузить данные в таблицу "moneyDataSet.DohodTypes". При необходимости она может быть перемещена или удалена.

this.dohodTypesTableAdapter.Fill(this.moneyDataSet.DohodTypes);

// TODO: данная строка кода позволяет загрузить данные в таблицу "moneyDataSet.Dohod". При необходимости она может быть перемещена или удалена.

this.dohodTableAdapter.Fill(this.moneyDataSet.Dohod);

// TODO: данная строка кода позволяет загрузить данные в таблицу "moneyDataSet.Users". При необходимости она может быть перемещена или удалена.

this.usersTableAdapter.Fill(this.moneyDataSet.Users);

find();

domainUpDown1.Text = DateTime.Now.Month+"";

domainUpDown2.Text = DateTime.Now.Year+ "";

}

public Dohod()

{

InitializeComponent();

}

//add

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

{

new DohodOperation(dohodTableAdapter, dohodBindingSource, moneyDataSet, null).Show();

}

//update

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

{

if (dataGridView1.RowCount > 0 && dataGridView1.CurrentRow != null)

new DohodOperation(dohodTableAdapter, dohodBindingSource, moneyDataSet, dataGridView1.CurrentRow).Show();

}

//remove

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

{

try

{

if (dataGridView1.RowCount > 0 && dataGridView1.SelectedCells.Count > 0)

if ((DialogResult = MessageBox.Show("Удалить выбранное?", "Удаление", MessageBoxButtons.YesNo)) == DialogResult.Yes)

{

dohodBindingSource.RemoveAt(dataGridView1.CurrentRow.Index);

dohodTableAdapter.Update(moneyDataSet.Dohod);

}

}

catch (Exception ex) { MessageBox.Show("Ошибка. Возможно вы пытаетесь удалить используемый объект"); }

}

private void domainUpDown1\_TextChanged(object sender, EventArgs e)

{

find();

}

private void find() {

try

{

DateTime date = new DateTime(int.Parse(domainUpDown2.Text), int.Parse(domainUpDown1.Text), 1);

this.dohodTableAdapter.FillBy(this.moneyDataSet.Dohod, date.AddDays(-1), date.AddMonths(1), int.Parse(comboBox1.SelectedValue.ToString()));

for (int i = 0; i < dataGridView1.RowCount; i++)

{

int index1 = dohodTypesBindingSource.Find("id", dataGridView1[2, i].Value.ToString());

dataGridView1[6, i].Value = ((DataRowView)(dohodTypesBindingSource[index1]))[1];

}

}

catch (Exception ex) { }

}

public Main Main

{

get => default(Main);

set

{

}

}

}

}

Файл «Borrow.cs»:

using System;

using System.Data;

using System.Windows.Forms;

using WindowsFormsApp1.MoneyDataSetTableAdapters;

namespace WindowsFormsApp1

{

public partial class Borrow : Form

{

private BorrowTableAdapter borrowTableAdapter;

private BindingSource borrowBindingSource;

private DataGridViewRow row;

private MoneyDataSet moneyDataSet;

bool isNew = false;

public Borrow(BorrowTableAdapter borrowTableAdapter, BindingSource borrowBindingSource, DataGridViewRow row, MoneyDataSet moneyDataSet)

{

InitializeComponent();

this.borrowTableAdapter = borrowTableAdapter;

this.borrowBindingSource = borrowBindingSource;

this.row = row;

this.moneyDataSet = moneyDataSet;

isNew = row == null;

}

public Dolg Dolg

{

get => default(Dolg);

set

{

}

}

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

{

// TODO: данная строка кода позволяет загрузить данные в таблицу "moneyDataSet1.Users". При необходимости она может быть перемещена или удалена.

this.usersTableAdapter.Fill(this.moneyDataSet1.Users);

if (!isNew) {

comboBox1.SelectedValue = int.Parse(row.Cells[1].Value.ToString());

textBox1.Text = row.Cells[5].Value.ToString();

dateTimePicker1.Text = row.Cells[3].Value.ToString();

checkBox1.Checked = bool.Parse(row.Cells[4].Value.ToString());

textBox2.Text = row.Cells[6].Value.ToString();

textBox3.Text = row.Cells[7].Value.ToString();

}

}

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

{

try

{

if (textBox1.Text.Length > 0)

if (isNew) insert(); else update();

}

catch (Exception ex) { }

}

private void insert()

{

DataRowView row = (DataRowView)borrowBindingSource.AddNew();

row["userId"] = comboBox1.SelectedValue;

row["kto"] = textBox1.Text;

row["datePol"] = dateTimePicker1.Value;

row["isOplachen"] = checkBox1.Checked;

row["comment"] = textBox2.Text;

row["count"] = textBox3.Text;

borrowBindingSource.EndEdit();

this.borrowTableAdapter.Update(moneyDataSet);

MessageBox.Show("Сохранено");

this.Close();

}

private void update()

{

row.Cells[1].Value = comboBox1.SelectedValue;

row.Cells[2].Value = comboBox1.Text;

row.Cells[5].Value = textBox1.Text;

row.Cells[3].Value = dateTimePicker1.Value;

row.Cells[4].Value = checkBox1.Checked;

row.Cells[6].Value = textBox2.Text;

row.Cells[7].Value = textBox3.Text;

borrowBindingSource.EndEdit();

this.borrowTableAdapter.Update(((DataRowView)row.DataBoundItem).Row);

MessageBox.Show("Сохранено");

this.Close();

}

}

}