МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ   
РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение

высшего образования

«Российский государственный университет им. А.Н. Косыгина   
(Технологии. Дизайн. Искусство)»

Кафедра автоматизированных систем обработки информации и управления

Отчет по лабораторной работе №3

по дисциплине «Информационные системы и базы данных».

Тема: «Разработка приложения БД. ч.3. Работа с основной таблицей»

Выполнил: Юрков Д.А., гр. МВА-122

Проверил: Адаев Р.Б.

Москва 2025

22. Книга рекламаций: Год, месяц, код модели компьютера, цена, продано, количество рекламаций.

Справочник моделей: Код модели, наименование модели, тип.

Логический вид таблицы представлен на рисунке 1.

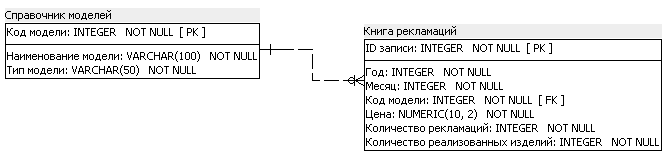


Рис.1

Физический вид таблицы представлен на рисунке 2.

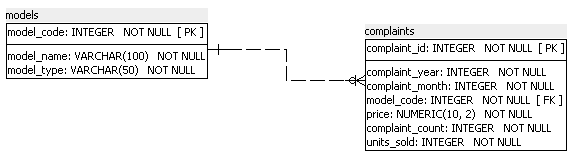


Рис.2

Текст представления соединения таблиц.

CREATE SEQUENCE complaints\_book.models\_seq;

CREATE TABLE complaints\_book.models (

model\_code INTEGER NOT NULL DEFAULT nextval('complaints\_book.models\_seq'),

model\_name VARCHAR(100) NOT NULL,

model\_type VARCHAR(50) NOT NULL,

CONSTRAINT model\_code PRIMARY KEY (model\_code)

);

ALTER SEQUENCE complaints\_book.models\_seq OWNED BY complaints\_book.models.model\_code;

CREATE SEQUENCE complaints\_book.complaints\_complaint\_id\_seq;

CREATE TABLE complaints\_book.complaints (

complaint\_id INTEGER NOT NULL DEFAULT nextval('complaints\_book.complaints\_complaint\_id\_seq'),

complaint\_year INTEGER DEFAULT 2025 NOT NULL,

complaint\_month INTEGER DEFAULT 1 NOT NULL,

model\_code INTEGER NOT NULL,

price NUMERIC(10,2) DEFAULT 0.00 NOT NULL,

complaint\_count INTEGER DEFAULT 0 NOT NULL,

units\_sold INTEGER DEFAULT 0 NOT NULL,

CONSTRAINT complaint\_id PRIMARY KEY (complaint\_id)

);

ALTER SEQUENCE complaints\_book.complaints\_complaint\_id\_seq OWNED BY complaints\_book.complaints.complaint\_id;

ALTER TABLE complaints\_book.complaints ADD CONSTRAINT models\_complaints\_fk

FOREIGN KEY (model\_code)

REFERENCES complaints\_book.models (model\_code)

ON DELETE RESTRICT

ON UPDATE RESTRICT

NOT DEFERRABLE;

программный код метода загрузки данных;

private void LoadComplaints()

{

\_complaints = new List<Complaint>();

string query = "SELECT \* FROM complaints\_book.complaints WHERE 1=1";

if (!string.IsNullOrEmpty(YearFilter1.Text) && int.TryParse(YearFilter1.Text, out int year1))

query += $" AND complaint\_year >= {year1}";

if (!string.IsNullOrEmpty(YearFilter2.Text) && int.TryParse(YearFilter2.Text, out int year2))

query += $" AND complaint\_year <= {year2}";

if (!string.IsNullOrEmpty(MonthFilter1.Text) && int.TryParse(MonthFilter1.Text, out int month1))

{

if (month1 < 1 || month1 > 12)

{

MessageBox.Show("Месяц должен быть от 1 до 12", "Ошибка", MessageBoxButton.OK, MessageBoxImage.Warning);

return;

}

query += $" AND complaint\_month >= {month1}";

}

if (!string.IsNullOrEmpty(MonthFilter2.Text) && int.TryParse(MonthFilter2.Text, out int month2))

{

if (month2 < 1 || month2 > 12)

{

MessageBox.Show("Месяц должен быть от 1 до 12", "Ошибка", MessageBoxButton.OK, MessageBoxImage.Warning);

return;

}

query += $" AND complaint\_month <= {month2}";

}

if (!string.IsNullOrEmpty(ModelCodeFilter.Text) && int.TryParse(ModelCodeFilter.Text, out int modelCode))

query += $" AND model\_code = {modelCode}";

if (!string.IsNullOrEmpty(PriceFilter1.Text) && decimal.TryParse(PriceFilter1.Text, out decimal price1))

query += $" AND price >= {price1}";

if (!string.IsNullOrEmpty(PriceFilter2.Text) && decimal.TryParse(PriceFilter2.Text, out decimal price2))

query += $" AND price <= {price2}";

if (!string.IsNullOrEmpty(ComplaintCountFilter1.Text) && int.TryParse(ComplaintCountFilter1.Text, out int complaintCount1))

query += $" AND complaint\_count >= {complaintCount1}";

if (!string.IsNullOrEmpty(ComplaintCountFilter2.Text) && int.TryParse(ComplaintCountFilter2.Text, out int complaintCount2))

query += $" AND complaint\_count <= {complaintCount2}";

if (!string.IsNullOrEmpty(UnitsSoldFilter1.Text) && int.TryParse(UnitsSoldFilter1.Text, out int unitsSold1))

query += $" AND units\_sold >= {unitsSold1}";

if (!string.IsNullOrEmpty(UnitsSoldFilter2.Text) && int.TryParse(UnitsSoldFilter2.Text, out int unitsSold2))

query += $" AND units\_sold <= {unitsSold2}";

try

{

using (var command = new OdbcCommand(query, \_connection))

using (var reader = command.ExecuteReader())

{

while (reader.Read())

{

\_complaints.Add(new Complaint

{

complaint\_id = reader.GetInt32(0),

complaint\_year = reader.GetInt32(1),

complaint\_month = reader.GetInt32(2),

model\_code = reader.GetInt32(3),

price = reader.GetDecimal(4),

complaint\_count = reader.GetInt32(5),

units\_sold = reader.GetInt32(6)

});

}

}

ComplaintsDataGrid.ItemsSource = \_complaints;

}

catch (Exception ex)

{

MessageBox.Show($"Ошибка загрузки данных: {ex.Message}", "Ошибка", MessageBoxButton.OK, MessageBoxImage.Error);

}

}

программный код формы редактирования основной таблицы;

using System;

using System.Data.Odbc;

using System.Windows;

using ISDBlab3;

namespace ISDBlab3

{

public partial class EditComplaintWindow : Window

{

private OdbcConnection \_connection;

private bool \_isNew;

private ComplaintsWindow.Complaint \_complaint;

public EditComplaintWindow(OdbcConnection connection, bool isNew, ComplaintsWindow.Complaint complaint)

{

InitializeComponent();

\_connection = connection;

\_isNew = isNew;

\_complaint = complaint;

if (!\_isNew && \_complaint != null)

{

edId.Text = \_complaint.complaint\_id.ToString();

edYear.Text = \_complaint.complaint\_year.ToString();

edMonth.Text = \_complaint.complaint\_month.ToString();

edModelCode.Text = \_complaint.model\_code.ToString();

edPrice.Text = \_complaint.price.ToString();

edComplaintCount.Text = \_complaint.complaint\_count.ToString();

edUnitsSold.Text = \_complaint.units\_sold.ToString();

}

else

{

edYear.Text = "2025";

edMonth.Text = "1";

edPrice.Text = "0.00";

edComplaintCount.Text = "0";

edUnitsSold.Text = "0";

}

}

private void bOk\_Click(object sender, RoutedEventArgs e)

{

if (!int.TryParse(edYear.Text, out int year) || !int.TryParse(edMonth.Text, out int month) ||

!int.TryParse(edModelCode.Text, out int modelCode) || !decimal.TryParse(edPrice.Text, System.Globalization.NumberStyles.Any, System.Globalization.CultureInfo.InvariantCulture, out decimal price)

||

!int.TryParse(edComplaintCount.Text, out int complaintCount) || !int.TryParse(edUnitsSold.Text, out int unitsSold))

{

MessageBox.Show("Введите корректные значения", "Ошибка", MessageBoxButton.OK, MessageBoxImage.Error);

return;

}

if (month < 1 || month > 12)

{

MessageBox.Show("Месяц должен быть от 1 до 12", "Ошибка", MessageBoxButton.OK, MessageBoxImage.Error);

return;

}

string strSql;

if (\_isNew)

{

strSql = "INSERT INTO complaints\_book.complaints (complaint\_year, complaint\_month, model\_code, price, complaint\_count, units\_sold) VALUES (?, ?, ?, ?, ?, ?)";

}

else

{

strSql = "UPDATE complaints\_book.complaints SET complaint\_year = ?, complaint\_month = ?, model\_code = ?, price = ?, complaint\_count = ?, units\_sold = ? WHERE complaint\_id = ?";

}

try

{

using (var command = new OdbcCommand(strSql, \_connection))

{

command.Parameters.AddWithValue("?", year);

command.Parameters.AddWithValue("?", month);

command.Parameters.AddWithValue("?", modelCode);

command.Parameters.AddWithValue("?", price);

command.Parameters.AddWithValue("?", complaintCount);

command.Parameters.AddWithValue("?", unitsSold);

if (!\_isNew)

command.Parameters.AddWithValue("?", \_complaint.complaint\_id);

command.ExecuteNonQuery();

}

DialogResult = true;

Close();

}

catch (Exception ex)

{

MessageBox.Show($"Ошибка сохранения: {ex.Message}", "Ошибка", MessageBoxButton.OK, MessageBoxImage.Error);

}

}

private void bCancel\_Click(object sender, RoutedEventArgs e)

{

DialogResult = false;

Close();

}

}

}

скриншот формы просмотра основной таблицы с заполненными данными на рисунке 3.

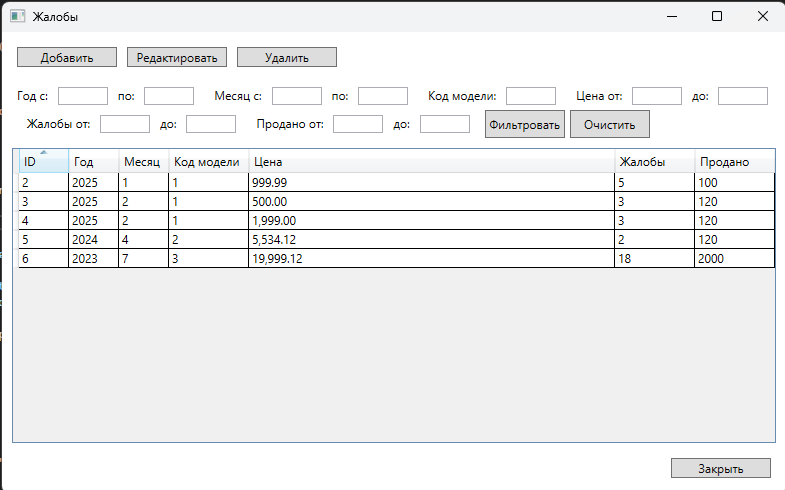


Рис.3

скриншот формы редактирования основной таблицы с заполненными данными на рисунке 4.

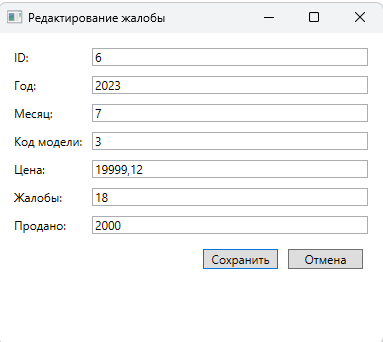


Рис.4

скриншот результатов отбора данных (панель отбора и отобранные строки) на рисунке 5.

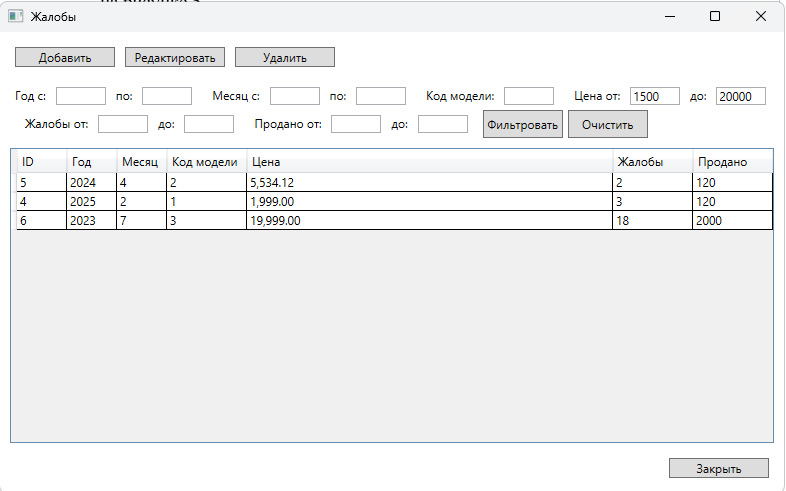


Рис.5