Московский Авиационный Институт

(Национальный Исследовательский Университет)

Институт информационных технологий и прикладной математики

Кафедра вычислительной математики и программирования

**Курсовой проект по курсу**

**«Базы данных»**

**Тема: Каталог автомобилей**

Выполнили: Моисеенков И. П.,

Гаптулхаков Р. Р.,

Шавандрин Ф. М.

Группа: М8О – 308Б-19

Преподаватель: Кузнецова Е. В.

Дата: \_\_\_\_\_\_\_\_\_\_\_

Оценка: \_\_\_\_\_\_\_\_\_\_\_

Подпись: \_\_\_\_\_\_\_\_\_\_\_

Москва, 2021

**Техническое задание**

**Технические требования**

Разработать прикладное программное обеспечение для обеспечения работы электронного каталога автомобилей, продающихся в Москве.

В городе располагается несколько автосалонов, в которых продаются различные модели автомобилей. Причем некоторые модели могут встречаться как в одном салоне, так и в нескольких. В каталоге должна содержаться актуальная информация обо всех предложениях.

В приложение можно войти от лица покупателя или от лица продавца (автосалона).

Покупателю должны быть показаны все актуальные позиции каталога. При нажатии на понравившийся автомобиль пользователь должен получить подробную информацию о нем и, при необходимости, оформить заказ. Пользователь может воспользоваться различными фильтрами, чтобы подобрать наиболее подходящий автомобиль.

Пользователь может посмотреть историю своих заказов и оценить каждый из них при желании. По результатам этих оценок рейтинг автосалона должен обновляться. Клиент не может оценить одну и ту же покупку дважды.

Продавец может выставить новый автомобиль на продажу. Для этого ему нужно заполнить все характеристики о машине.

Продавец может посмотреть отчеты о продажах и выручке своего автосалона, а также о наиболее популярных автомобилях в целом.

Требования по функциональности

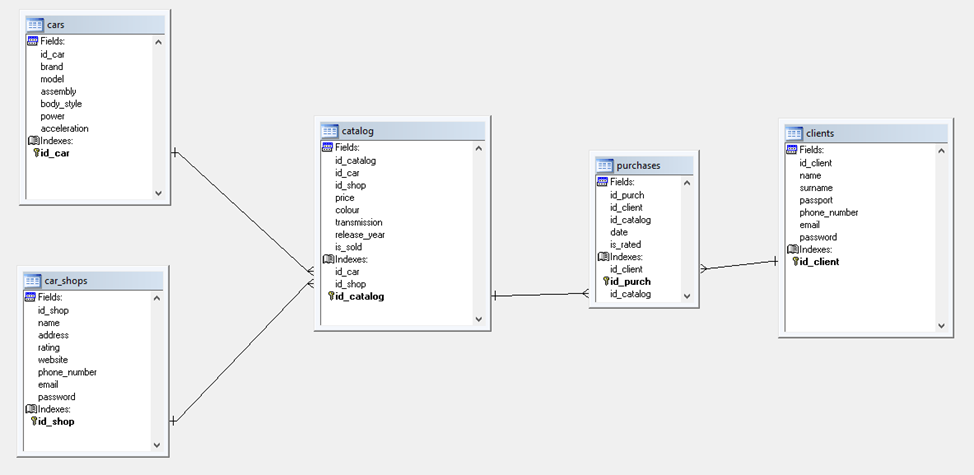
1. Возможность обновления базы данных: добавление сведений о совершенной покупке, добавление новых позиций в каталог, изменение рейтинга автосалона после оценки пользователя.
2. Поиск автомобилей по заданным фильтрам в каталоге.
3. Безопасное удаление информации о купленном автомобиле в базе данных.
4. Возможность создания отчётов:
   1. Сводка по продажам и выручке автосалонов.
   2. Список наиболее популярных автомобилей.

Инструменты

1. CASE-средство - Erwin Data Modeler
2. СУБД - Microsoft Visual FoxPro 9.0

План работы

| № | Задача | Исполнитель | Срок выполнения |
| --- | --- | --- | --- |
| 1 | Составление технического задания | Моисеенков И. П.,  Шавандрин Ф. М.,  Гаптулхаков Р. Р. | 17.11.2021 |
| 2 | Проектирование и заполнение базы данных в FoxPro и ERwin | Моисеенков И. П.,  Шавандрин Ф. М. | 17.11.2021 |
| 3 | Утвердить техническое задание с преподавателем | Моисеенков И. П.,  Шавандрин Ф. М.,  Гаптулхаков Р. Р. | 19.11.2021 |
| 4 | Разработка интерфейса для регистрации и авторизации | Гаптулхаков Р. Р. | 12.12.2021 |
| 5 | Разработка интерфейса взаимодействия с пользователем | Моисеенков И.П. | 12.12.2021 |
| 6 | Разработка интерфейса взаимодействия с автосалоном | Шавандрин Ф. М. | 12.12.2021 |
| 8 | Объединение всех результатов в один проект | Моисеенков И. П.,  Шавандрин Ф. М.,  Гаптулхаков Р. Р. | 13.12.2021 |
| 9 | Оформление отчета | Моисеенков И. П.,  Шавандрин Ф. М.,  Гаптулхаков Р. Р. | 13.12.2021 |
| 10 | Сдача проекта | Моисеенков И. П.,  Шавандрин Ф. М.,  Гаптулхаков Р. Р. | 14.12.2021 |

**База данных в FoxPro**

**Таблицы:**

*Cars (автомобили):* 

* id\_car – идентификатор автомобиля (integer)
* brand – марка автомобиля (character)
* model – модель автомобиля (character)
* assembly – производитель автомобиля (character)
* body\_style – тип кузова (character)
* power – мощность, л.с. (integer)
* acceleration – разгон до 100 км/ч, с (double)

*Car\_shops (автосалоны):* **

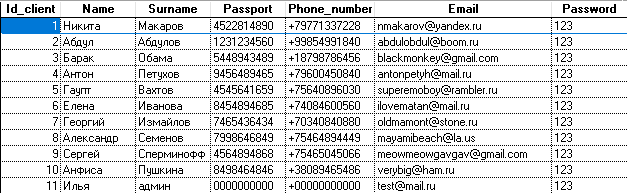
* id\_shop – идентификатор автосалона (integer)
* name – название автосалона (character)
* address – адрес (character)
* rating – рейтинг (double)
* website – веб-сайт (character)
* phone\_number – номер телефона (character)
* email – адрес электронной почты (character)
* password – пароль от аккаунта (character)

*Catalog (каталог):*

**

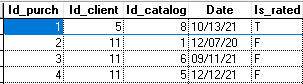
* id\_catalog – идентификатор позиции в каталоге (integer)
* id\_car – идентификатор автомобиля (integer)
* id\_shop – идентификатор автосалона (integer)
* price – цена (integer)
* colour – цвет (character)
* transmission – тип трансмиссии (character)
* release\_year – год выпуска (integer)
* is\_sold – продана ли машина (logical)

*Clients (клиенты):*

**

* id\_client – идентификатор клиента (integer)
* name – имя (character)
* surname – фамилия (character)
* passport – номер паспорта (character)
* phone\_number – номер телефона (character)
* email – адрес электронной почты (character)
* password – пароль от аккаунта (character)

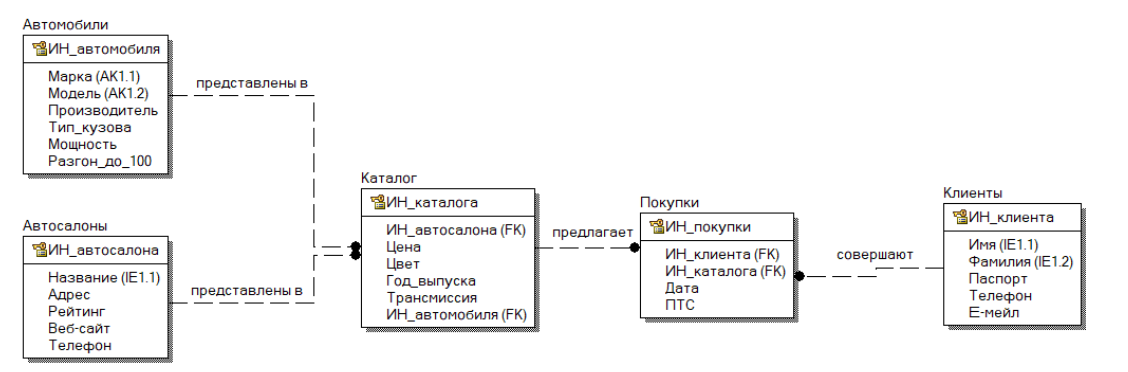
*Purchases (покупки):*

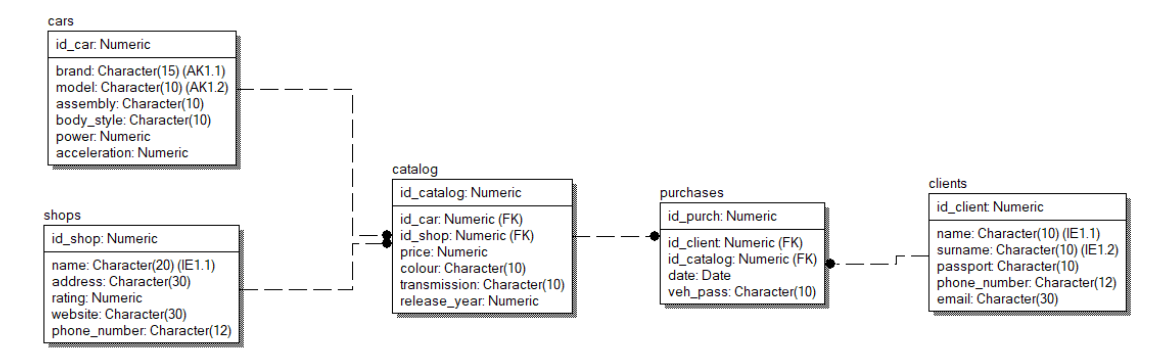
**

* id\_purch – идентификатор покупки (integer)
* id\_client – идентификатор клиента (integer)
* id\_catalog – идентификатор позиции в каталоге (integer)
* date – дата покупки (date)
* is\_rated – оценена ли покупка (logical)

База данных в Erwin

*Логическая модель:*

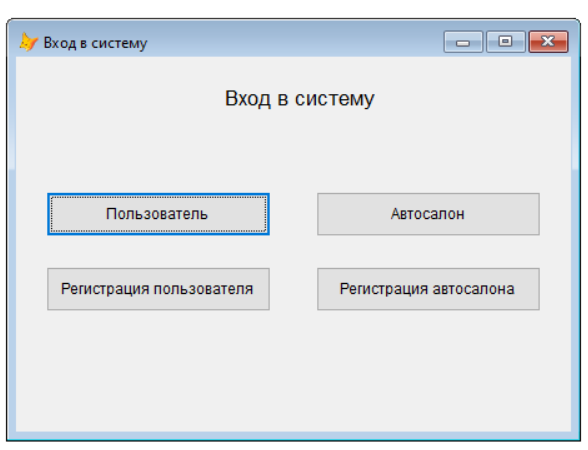


*Физическая модель:*

**Описание форм и** отчетов

start\_form

Форма, в которой пользователь может перейти к форме для входа в свой аккаунт или для регистрации . Также для перехода автосалона к форме регистрации или входа.



*код кнопки “Пользователь”*

DO FORM ".\user authorization.scx"

RELEASE THISFORM

*код кнопки “Автосалон”*

DO FORM ".\seller authorization.scx"

RELEASE THISFORM

*код кнопки “ Регистрация пользователя”*

DO FORM ".\user registration.scx"

RELEASE THISFORM

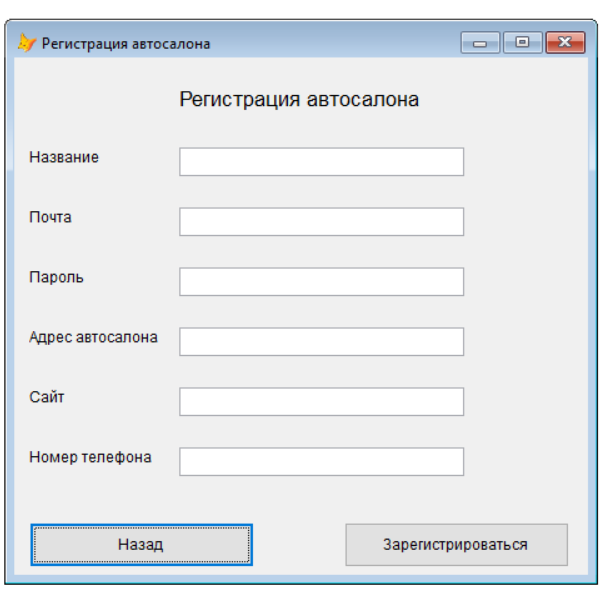
*код кнопки “Регистрация автосалона”*

DO FORM ".\seller registration.scx"

RELEASE THISFORM

user\_registration

Форма, в которой пользователю предлагается ввести свои данные, чтобы зарегистрироваться в приложении. Все поля должны быть заполнены, иначе будет выводиться ошибка. Уникальность пользователя определяется по почте, номеру паспорта и номеру телефона. После регистрации пользователю присваивается уникальный id для дальнейшей удобной работы.



*код кнопки “зарегистрироваться”*

LOCATE FOR alltrim(email) = alltrim(ThisForm.txtEmail.Value) .or.;

alltrim(phone\_number) = alltrim(ThisForm.txtphone.Value) .or.;

alltrim(passport) = alltrim(ThisForm.txtpassport.Value)

IF FOUND()

IF alltrim(email) = alltrim(ThisForm.txtEmail.Value)

MESSAGEBOX('User with this email exists')

thisform.txtemail.SetFocus

RETURN

ENDIF

IF alltrim(phone\_number) = alltrim(ThisForm.txtphone.Value)

MESSAGEBOX('User with this number exists')

thisform.txtphone.SetFocus

RETURN

ENDIF

IF alltrim(passport) = alltrim(ThisForm.txtpassport.Value)

MESSAGEBOX('User with this passport exists')

thisform.txtpassport.SetFocus

RETURN

ENDIF

ELSE

curr\_val = RECCOUNT() + 1

n = ALLTRIM(ThisForm.txtname.Value)

sn = ALLTRIM(ThisForm.txtsurname.Value)

email\_ = ALLTRIM(ThisForm.txtemail.Value)

phone = ALLTRIM(ThisForm.txtphone.Value)

passport\_ = ALLTRIM(ThisForm.txtpassport.Value)

password\_ = ALLTRIM(ThisForm.txtpassword.Value)

INSERT INTO clients (name, surname, email, phone\_number, passport, password, id\_client);

VALUES(n,sn,email\_,phone,passport\_,password\_,curr\_val)

thisform.command4.Enabled = .F.

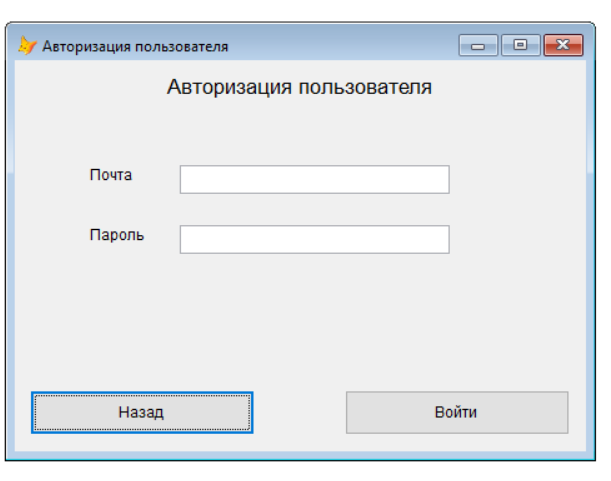
DO FORM '.\start\_form.scx'

Thisform.Release

ENDIF

user\_authorization

Форма для входа пользователя в свой аккаунт. Требуется ввести почту и пароль.



*код кнопки “Войти”*

IF EMPTY(ThisForm.txtpassword.Value)

MESSAGEBOX('Fill in the field password')

RETURN

ENDIF

LOCATE FOR alltrim(email) = alltrim(ThisForm.txtEmail.Value) .and.;

alltrim(password) = alltrim(ThisForm.txtpassword.Value)

IF FOUND()

PUBLIC global\_login\_user

global\_login\_user = alltrim(ThisForm.txtEmail.Value)

ThisForm.command2.Enabled = .F.

DO FORM '.\catalog.scx'

Thisform.Release

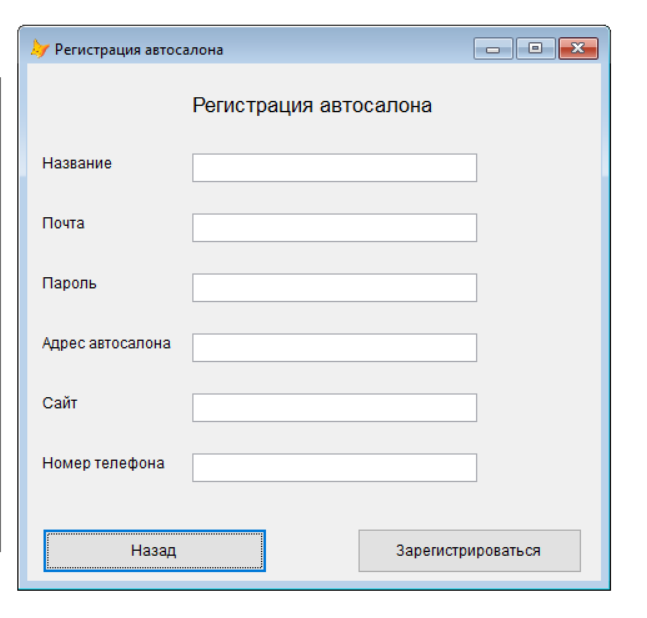
ELSE

MESSAGEBOX('User is not found')

ENDIF

seller\_registration

Форма, в которой продавцу предлагается заполнить поля, чтобы зарегистрировать свой автосалон в приложении. Все поля должны быть заполнены, иначе будет выводиться ошибка. Уникальность автосалона определяется по почте. После регистрации автосалону присваивается уникальный id для дальнейшей удобной работы.



*код кнопки “зарегистрироваться”*

\* обработка пустых строк

IF EMPTY(ThisForm.txtwebsite.Value)

MESSAGEBOX('Fill in the field website')

RETURN

ENDIF

LOCATE FOR alltrim(email) = alltrim(ThisForm.txtemail.Value)

IF FOUND()

MESSAGEBOX('User with this email exists')

thisform.txtemail.SetFocus

RETURN

ELSE

curr\_val = RECCOUNT() + 1

n = ALLTRIM(ThisForm.txtname.Value)

email\_ = ALLTRIM(ThisForm.txtemail.Value)

phone = ALLTRIM(ThisForm.txtphone.Value)

address\_ = ALLTRIM(ThisForm.txtaddress.Value)

password\_ = ALLTRIM(ThisForm.txtpassword.Value)

web\_ = ALLTRIM(ThisForm.txtwebsite.Value)

INSERT INTO car\_shops (id\_shop, name, address, rating, website, phone\_number, email, password);

VALUES(curr\_val, n, address\_, 5, web\_, phone, email\_, password\_)

thisform.command4.Enabled = .F.

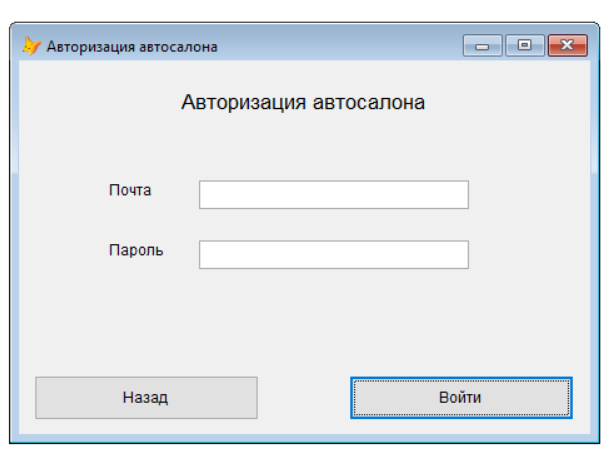
DO FORM '.\start\_form.scx'

Thisform.Release

ENDIF

seller\_authorization

Форма для входа продавца в свой аккаунт. Требуется ввести почту и пароль.



*код кнопки “Войти”*

IF EMPTY(ThisForm.txtpassword.Value)

MESSAGEBOX('Fill in the field password')

RETURN

ENDIF

LOCATE FOR alltrim(email) = alltrim(ThisForm.txtEmail.Value) .and.;

alltrim(password) = alltrim(ThisForm.txtpassword.Value)

IF FOUND()

PUBLIC global\_login\_seller

global\_login\_seller = alltrim(ThisForm.txtemail.Value)

ThisForm.command2.Enabled = .F.

DO FORM '.\start\_form.scx'

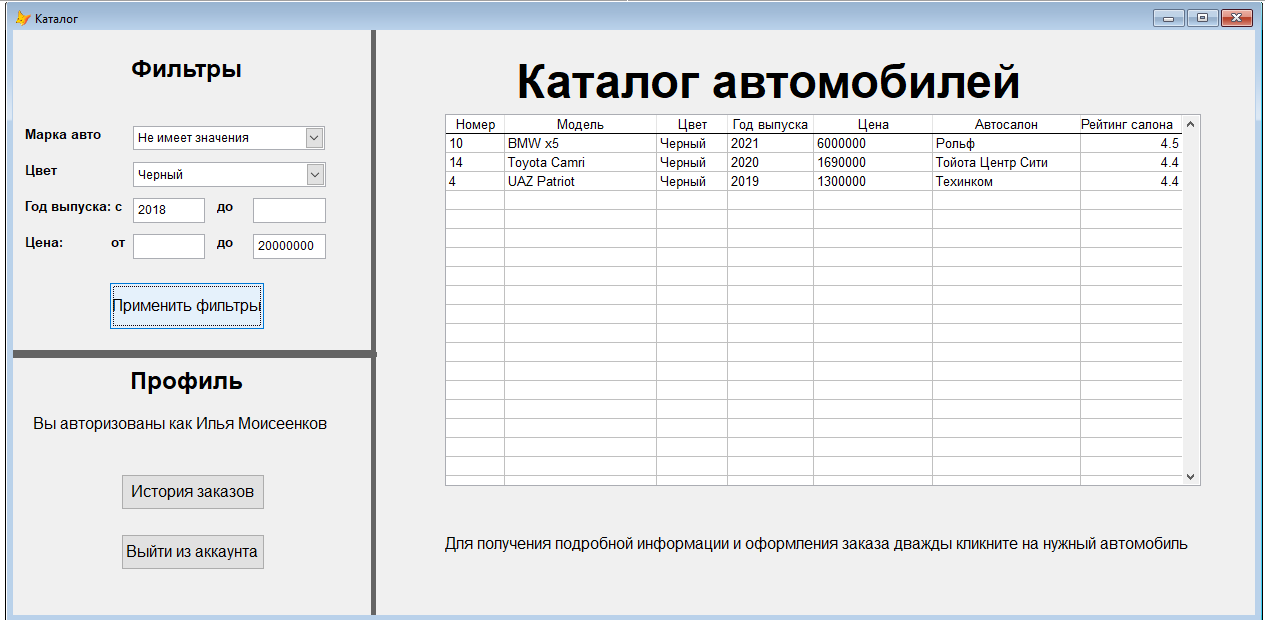
Thisform.Release

ELSE

MESSAGEBOX('User is not found')

ENDIF

**catalog**



Форма для просмотра каталога от лица пользователя. Можно применить различные фильтры и получить подробную информацию, дважды нажав на соответствующую строчку. Из этой формы можно перейти в историю заказов.

*Код init для таблицы*

SELECT catalog.id\_catalog, ALLTRIM(cars.brand) + ' ' + ALLTRIM(cars.model) AS brand\_model, catalog.colour, catalog.release\_year, catalog.price, car\_shops.name, car\_shops.rating;

FROM catalog, cars, car\_shops;

WHERE NOT catalog.is\_sold;

AND catalog.id\_car = cars.id\_car AND catalog.id\_shop = car\_shops.id\_shop;

ORDER BY brand\_model;

INTO CURSOR cars\_for\_sale

\* заполняем таблицу данными из запроса

thisform.Grid1.RecordSourceType = 4

thisform.Grid1.RecordSource = 'SELECT \* FROM cars\_for\_sale INTO CURSOR tmp2'

*Код dblclick для таблицы*

PUBLIC selected\_catalog\_id

ThisForm.Grid1.SetFocus()

selected\_catalog\_id = ThisForm.Grid1.Column1.Text1.Value(ThisForm.Grid1.ActiveRow)

DO FORM ".\buy car.scx"

RELEASE THISFORM

*Код init для выпадающего списка с моделями автомобилей*

SELECT DISTINCT cars.brand;

FROM cars, catalog;

WHERE catalog.id\_car = cars.id\_car;

AND NOT catalog.is\_sold;

UNION;

SELECT "Не имеет значения" FROM cars;

INTO CURSOR available\_cars

Thisform.Combo1.RowSourceType = 3

Thisform.Combo1.RowSource = "SELECT \* FROM available\_cars INTO CURSOR tmp"

*Код click для кнопки применения фильтров*

IF EMPTY(ThisForm.Combo1.Value) OR ThisForm.Combo1.Value = "Не имеет значения" THEN

SELECT cars.brand FROM cars;

INTO CURSOR available\_cars NOFILTER

ELSE

car\_brand = ThisForm.Combo1.Value

SELECT car\_brand AS brand FROM cars;

INTO CURSOR available\_cars NOFILTER

ENDIF

IF EMPTY(ThisForm.Combo2.Value) OR ThisForm.Combo2.Value = "Не имеет значения" THEN

SELECT catalog.colour FROM catalog;

INTO CURSOR available\_colours NOFILTER

ELSE

car\_colour = ThisForm.Combo2.Value

SELECT car\_colour AS colour FROM catalog;

INTO CURSOR available\_colours NOFILTER

ENDIF

IF EMPTY(ThisForm.Text1.Value) THEN

year\_min = 1000

ELSE

year\_min = VAL(ThisForm.Text1.Value)

ENDIF

IF EMPTY(ThisForm.Text2.Value) THEN

year\_max = 3000

ELSE

year\_max = VAL(ThisForm.Text2.Value)

ENDIF

IF EMPTY(ThisForm.Text3.Value) THEN

price\_min = 0

ELSE

price\_min = VAL(ThisForm.Text3.Value)

ENDIF

IF EMPTY(ThisForm.Text4.Value) THEN

price\_max = 100000000000

ELSE

price\_max = VAL(ThisForm.Text4.Value)

ENDIF

SELECT catalog.id\_catalog, ALLTRIM(cars.brand) + ' ' + ALLTRIM(cars.model) AS brand\_model, catalog.colour, catalog.release\_year, catalog.price, car\_shops.name, car\_shops.rating;

FROM catalog, cars, car\_shops;

WHERE NOT catalog.is\_sold;

AND catalog.id\_car = cars.id\_car AND catalog.id\_shop = car\_shops.id\_shop;

AND cars.brand IN (SELECT brand FROM available\_cars);

AND catalog.colour IN (SELECT colour FROM available\_colours);

AND catalog.release\_year BETWEEN year\_min AND year\_max;

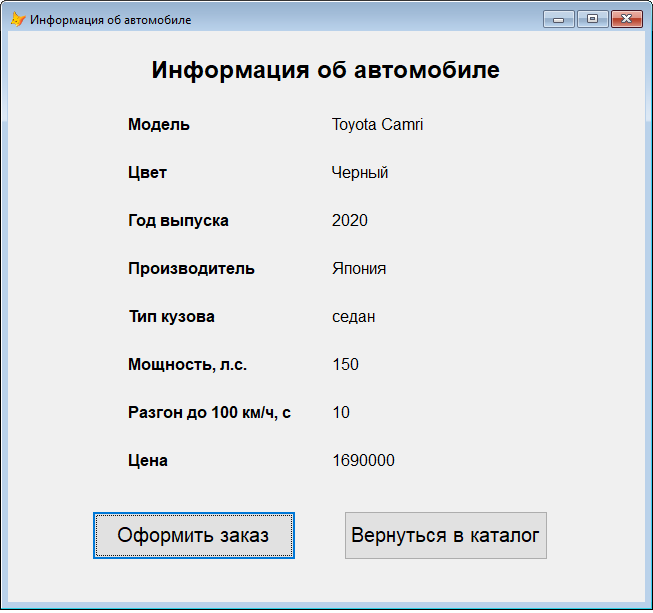
AND catalog.price BETWEEN price\_min AND price\_max;

ORDER BY brand\_model;

INTO CURSOR filtered\_cars

thisform.Grid1.RecordSource = 'SELECT \* FROM filtered\_cars INTO CURSOR tmp3'

buy car



Форма для просмотра подробной информации об автомобиле. Реализованы две кнопки - оформление заказа и возвращение в каталог.

*Код click для кнопки оформления заказа*

PUBLIC global\_login\_user

PUBLIC selected\_catalog\_id

LOCAL current\_client\_id

SELECT id\_client;

FROM clients;

WHERE ALLTRIM(clients.email) = global\_login\_user;

INTO CURSOR id\_client

current\_client\_id = id\_client.id\_client

SELECT purchases

id\_purchase = RECCOUNT() + 1

INSERT INTO purchases VALUES (id\_purchase, current\_client\_id, selected\_catalog\_id, DATE(), .F.)

UPDATE catalog SET is\_sold = .T.;

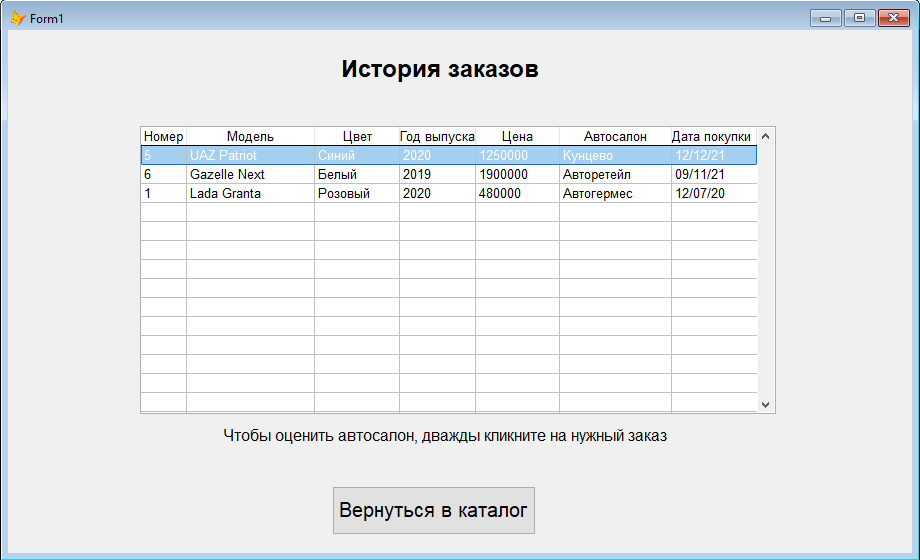
WHERE id\_catalog = selected\_catalog\_id

MESSAGEBOX("Поздравляем с покупкой!")

DO FORM '.\catalog.scx'

RELEASE THISFORM

history



Форма для просмотра истории заказов пользователя. В таблице представлены все покупки в обратном хронологическом порядке. Пользователь может запустить форму оценивания автомобиля, дважды кликнув на соответствующую строчку таблицы.

*Код init для таблицы с историей заказов*

PUBLIC global\_login\_user

SELECT clients.id\_client AS id;

FROM clients;

WHERE ALLTRIM(clients.email) = global\_login\_user;

INTO CURSOR user\_id

SELECT catalog.id\_catalog, ALLTRIM(cars.brand) + ' ' + ALLTRIM(cars.model), catalog.colour, catalog.release\_year, catalog.price, car\_shops.name, purchases.date;

FROM catalog, cars, car\_shops, purchases;

WHERE;

catalog.id\_car = cars.id\_car AND catalog.id\_shop = car\_shops.id\_shop;

AND catalog.id\_catalog = purchases.id\_catalog AND purchases.id\_client = user\_id.id;

ORDER BY purchases.date DESC;

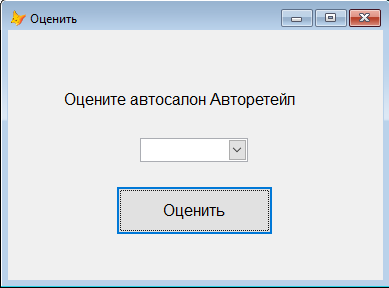
INTO CURSOR users\_cars

\* заполняем таблицу данными из запроса

thisform.Grid1.RecordSourceType = 4

thisform.Grid1.RecordSource = 'SELECT \* FROM users\_cars INTO CURSOR tmp'

rate



Форма для оценивания автосалона. Пользователь может выбрать оценку от 1 до 5. Если пользователь не выбрал оценку или оценивает этот заказ повторно, то ему покажут соответствующее сообщение.

*Код click для кнопки оценки заказа*

PUBLIC selected\_catalog\_id

LOCAL cur\_rate, prev\_rate, new\_rate

IF EMPTY(ThisForm.Combo1.Value)

MESSAGEBOX('Выберите оценку!')

RETURN

ENDIF

SELECT purchases.is\_rated;

FROM purchases;

WHERE purchases.id\_catalog = selected\_catalog\_id;

INTO CURSOR was\_rated\_before

IF was\_rated\_before.is\_rated THEN

MESSAGEBOX("Вы уже оценили этот салон!")

ELSE

cur\_rate = VAL(Thisform.Combo1.Value)

SELECT car\_shops.rating;

FROM car\_shops, purchases, catalog;

WHERE catalog.id\_catalog = selected\_catalog\_id;

AND catalog.id\_shop = car\_shops.id\_shop;

INTO CURSOR prev\_rating

prev\_rate = prev\_rating.rating

new\_rate = (prev\_rate + cur\_rate) / 2

SELECT car\_shops.id\_shop;

FROM catalog, car\_shops;

WHERE catalog.id\_catalog = selected\_catalog\_id;

AND car\_shops.id\_shop = catalog.id\_shop;

INTO CURSOR car\_shop\_id

UPDATE car\_shops SET car\_shops.rating = new\_rate;

WHERE car\_shops.id\_shop = car\_shop\_id.id\_shop

UPDATE purchases SET purchases.is\_rated = .T.;

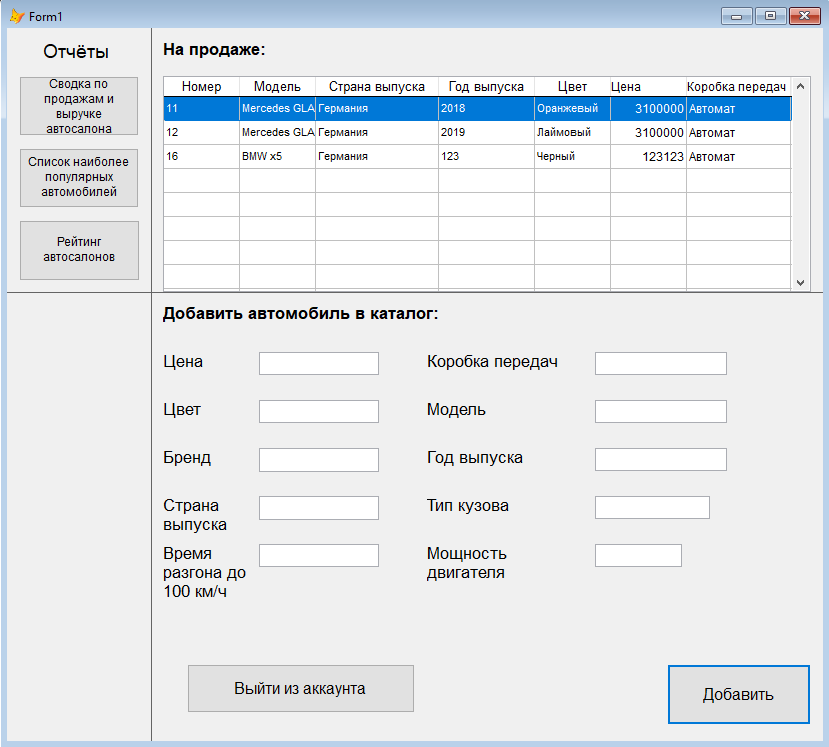
WHERE purchases.id\_catalog = selected\_catalog\_id

MESSAGEBOX("Ваша оценка принята. Спасибо!")

ENDIF

RELEASE THISFORM

cars\_for\_selling



Форма для просмотра доступных для продажи автомобилей в автосалоне, под которым был осуществлен вход в приложение. Есть возможность добавить новый автомобиль в каталог, введя его характеристики в соответствующие поля, а также посмотреть два отчёта.

*Код init для таблицы*

SELECT car\_shops.id\_shop FROM car\_shops WHERE car\_shops.email = global\_login\_seller INTO CURSOR curr\_car\_shop

SELECT catalog.id\_car FROM catalog, curr\_car\_shop WHERE catalog.id\_shop = curr\_car\_shop.id\_shop INTO CURSOR needed\_cars DISTINCT

SELECT catalog.id\_catalog, ALLTRIM(cars.brand) + ' ' + ALLTRIM(cars.model), cars.assembly, catalog.release\_year, catalog.colour, catalog.price, catalog.transmission;

FROM catalog, cars, needed\_cars;

WHERE NOT catalog.is\_sold;

AND catalog.id\_car = cars.id\_car;

AND catalog.id\_shop = car\_shops.id\_shop;

AND catalog.id\_car = needed\_cars.id\_car;

ORDER BY catalog.id\_catalog;

INTO CURSOR cars\_for\_sale

\* заполняем таблицу данными из запроса

thisform.Grid1.RecordSourceType = 4

thisform.Grid1.RecordSource = 'SELECT \* FROM cars\_for\_sale INTO CURSOR tmp'

*Код click для добавления автомобиля в каталог*

IF EMPTY(ThisForm.txtPrice.Value)

MESSAGEBOX('Заполните поле "Цена"')

RETURN

ENDIF

IF EMPTY(ThisForm.txtColour.Value)

MESSAGEBOX('Заполните поле "Цвет"')

RETURN

ENDIF

IF EMPTY(ThisForm.txtBrand.Value)

MESSAGEBOX('Заполните поле "Бренд"')

RETURN

ENDIF

IF EMPTY(ThisForm.txtAssembly.Value)

MESSAGEBOX('Заполните поле "Страна выпуска"')

RETURN

ENDIF

IF EMPTY(ThisForm.txtAcceleration.Value)

MESSAGEBOX('Заполните поле "Время разгона до 100 км/ч"')

RETURN

ENDIF

IF EMPTY(ThisForm.txtTransmission.Value)

MESSAGEBOX('Заполните поле "Коробка передач"')

RETURN

ENDIF

IF EMPTY(ThisForm.txtModel.Value)

MESSAGEBOX('Заполните поле "Модель"')

RETURN

ENDIF

IF EMPTY(ThisForm.txtRelease\_year.Value)

MESSAGEBOX('Заполните поле "Год выпуска"')

RETURN

ENDIF

IF EMPTY(ThisForm.txtBody\_style.Value)

MESSAGEBOX('Заполните поле "Тип кузова"')

RETURN

ENDIF

IF EMPTY(ThisForm.txtPower.Value)

MESSAGEBOX('Заполните поле "Мощность двигателя"')

RETURN

ENDIF

SELECT catalog

catalog\_ = RECCOUNT() + 1

price\_ = VAL(ALLTRIM(ThisForm.txtPrice.Value))

colour\_ = ALLTRIM(ThisForm.txtColour.Value)

brand\_ = ALLTRIM(ThisForm.txtBrand.Value)

assembly\_ = ALLTRIM(ThisForm.txtAssembly.Value)

acceleration\_ = VAL(ALLTRIM(ThisForm.txtAcceleration.Value))

transmission\_ = ALLTRIM(ThisForm.txtTransmission.Value)

model\_ = ALLTRIM(ThisForm.txtModel.Value)

releaseyear\_ = VAL(ALLTRIM(ThisForm.txtRelease\_year.Value))

bodystyle\_ = ALLTRIM(ThisForm.txtBody\_style.Value)

power\_ = VAL(ALLTRIM(ThisForm.txtPower.Value))

SELECT car\_shops.id\_shop FROM car\_shops WHERE ALLTRIM(car\_shops.email) = global\_login\_seller INTO CURSOR id\_shop

idshop\_ = id\_shop.id\_shop

SELECT cars

LOCATE FOR ALLTRIM(assembly) = alltrim(ThisForm.txtAssembly.Value);

AND ALLTRIM(brand) = alltrim(ThisForm.txtBrand.Value);

AND ALLTRIM(model) = alltrim(ThisForm.txtModel.Value);

AND ALLTRIM(acceleration) = alltrim(ThisForm.txtAcceleration.Value);

AND ALLTRIM(body\_style) = alltrim(ThisForm.txtBody\_style.Value);

AND ALLTRIM(power) = alltrim(ThisForm.txtPower.Value)

IF FOUND()

id\_c = cars.id\_car

ELSE

SELECT cars

id\_c = RECCOUNT() + 1

ENDIF

INSERT INTO catalog (id\_car, id\_catalog, id\_shop, price, colour, release\_year, transmission, is\_sold);

VALUES(id\_c, catalog\_, idshop\_, price\_, colour\_, releaseyear\_, transmission\_, .F.)

INSERT INTO cars (id\_car, brand, assembly, model, acceleration, body\_style, power);

VALUES(id\_c, brand\_, assembly\_, model\_, acceleration\_, bodystyle\_, power\_)

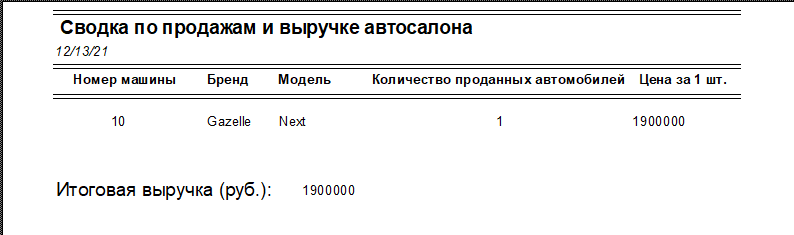
thisform.command1.Enabled = .F.

MESSAGEBOX('Машина успешно добавлена!')

DO FORM './myform.scx'

RELEASE THISFORM

gain



Отчет, предоставляющий информацию о продажах и выручке данного автосалона. Представляет собой таблицу, в которой указаны номер машины, бренд, модель, количество проданных автомобилей одного типа и цена авто за 1 шт. В конце отчета представлена итоговая выручка в рублях.

*Код click для предварительного просмотра отчета*

SELECT car\_shops.id\_shop FROM car\_shops WHERE car\_shops.email = global\_login\_seller INTO CURSOR cur

SELECT catalog.id\_car FROM catalog, cur WHERE catalog.id\_shop = cur.id\_shop INTO CURSOR curr\_cars DISTINCT

SELECT catalog.id\_car, cars.brand, cars.model, COUNT(catalog.price) AS count\_sold, catalog.price;

FROM catalog, curr\_cars, cars;

WHERE catalog.id\_car = curr\_cars.id\_car;

AND catalog.id\_shop = cur.id\_shop;

AND cars.id\_car = curr\_cars.id\_car;

AND catalog.is\_sold;

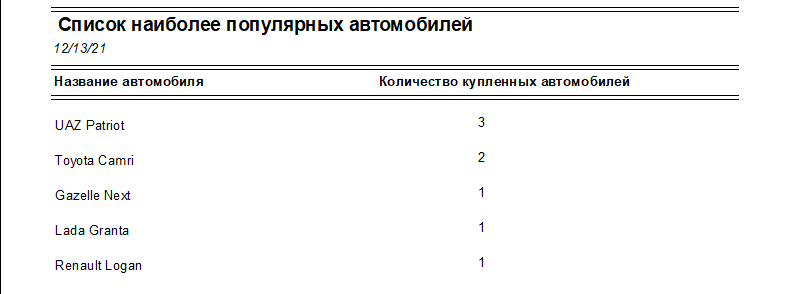
GROUP BY catalog.id\_car, cars.brand, cars.model, catalog.price;

INTO TABLE first\_table

SELECT SUM(first\_table.price) AS total\_sum FROM first\_table INTO CURSOR cur1

REPORT FORM ".\gain.frx" PREVIEW

most\_popular\_cars



Отчет, предоставляющий информацию о наиболее популярных автомобилей среди всех автосалонов, представленных в БД. Представляет собой таблицу, в которой указаны название автомобиля и количество купленных автомобилей одного типа. Данные таблицы отсортированы по убыванию купленных одинаковых машин.

*Код click для предварительного просмотра отчета*

SELECT ALLTRIM(cars.brand) + ' ' + ALLTRIM(cars.model) AS brand\_model, COUNT(purchases.id\_purch) as count;

FROM catalog, purchases, cars;

WHERE purchases.id\_catalog = catalog.id\_catalog;

AND catalog.is\_sold;

AND catalog.id\_car = cars.id\_car;

GROUP BY brand\_model;

ORDER BY count DESC;

INTO TABLE second\_table

REPORT FORM ".\most\_popular\_cars.frx" PREVIEW

car\_shops\_rating



Отчет, предоставляющий рейтинг автосалонов по количеству проданных автомобилей, а также место автосалона, под которым вошли в приложение, в данном рейтинге. Данные отчета отсортированы по итоговой выручке каждого автосалона.

*Код click для предварительного просмотра отчета*

SELECT car\_shops.id\_shop, car\_shops.name, SUM(catalog.price) as gain;

FROM catalog, purchases, car\_shops, cars;

WHERE purchases.id\_catalog = catalog.id\_catalog;

AND catalog.is\_sold;

AND catalog.id\_car = cars.id\_car;

AND car\_shops.id\_shop = catalog.id\_shop;

GROUP BY car\_shops.name, car\_shops.id\_shop;

ORDER BY gain DESC;

INTO TABLE third\_table

SELECT third\_table.id\_shop FROM third\_table, car\_shops;

WHERE car\_shops.email = global\_login\_seller;

AND third\_table.id\_shop = car\_shops.id\_shop;

INTO CURSOR cur

SELECT COUNT(\*) FROM third\_table, cur;

WHERE third\_table.id\_shop != cur.id\_shop;

INTO CURSOR place

SELECT third\_table

REPLACE ALL third\_table.id\_shop WITH RECNO()

REPORT FORM ".\car\_shops\_rating.frx" PREVIEW