МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ   
РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования «Самарский национальный исследовательский университет имени академика С.П. Королева»  
(Самарский университет)

Институт информатики и кибернетики\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Кафедра программных систем\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**ОТЧЕТ**  
  
по лабораторной работе №4

«Получение информации из реляционной базы данных»

по дисциплине «Логическое программирование»

Обучающаяся в группе 6301-020302D Соколова Алёна

Преподаватель Лобанков Антон Алексеевич

Самара 2025

# ВВЕДЕНИЕ

Целью работы является в процессе выполнения заданий ознакомиться с возможностью получения информации из реляционной базы данных и отображения этой информации в виде HTML страниц на прологе.

Задания:

1. на основе 3 лабораторной работы создать базу данных о заданной предметной области в виде таблиц. Таблицы должны создаваться используя соответствующие предикаты;
2. разработать набор sql запросов, при помощи которых можно реализовать не менее двух запросов, приведенных в варианте задания к лабораторной 3;
3. используя HTTP server libraries написать CRUD (Create/Read/Update/Delete) для всех таблиц базы данных. Предоставить пользователю возможность сбросить базу данных к исходному (тестовому) варианту заполнения;
4. реализовать вывод результатов поисковых запросов к базе данных на html странице.

Вариант 26:

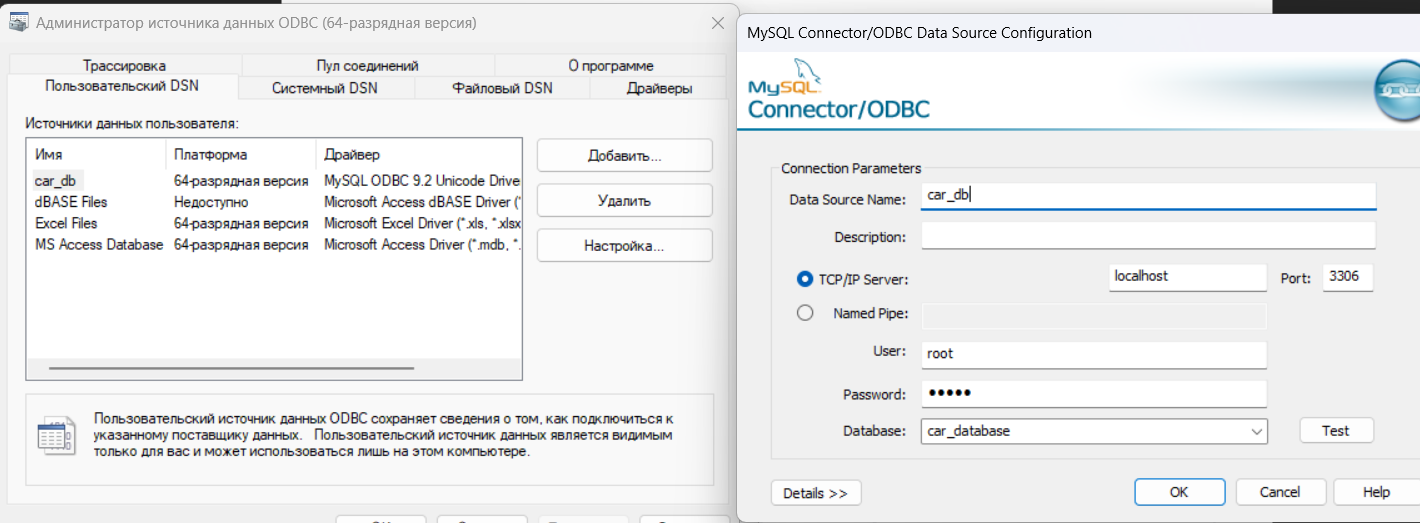
Предметная область – база данных продажи автомобилей. Каждый автомобиль может быть описана структурой: марка автомобиля, страна фирмы-изготовителя, список фирм-продавцов. Фирма-продавец может быть описана структурой: название фирмы, страна, список имеющихся моделей. Модель может быть описана структурой: наименование модели, цена, список имеющихся расцветок.

В этот раз будем реализовывать несколько запросов:

1. найти марку и модель автомобиля, у которой минимальная цена;
2. найти все фирмы, которые продают автомобили заданной расцветки.

Пролог-сервер содержит базу данных автомобилей, где есть возможность просматривать список всех машин на продажу, добавлять, удалять строки, выполнять запросы поиска всех автомобилей определенной расцветки и поиска самой дешевой машины. При необходимости есть возможность отката базы данных (БД) к первоначальному состоянию.

В 4 лабораторной нам нужны SQL запросы, то есть должна быть работа с реляционной БД через ODBC драйвер. Для этого необходимо было установить драйвер, добавить пользовательский DNS и установить настройки, отображенные на рисунке 1.

  
Рисунок 1 – Настройка ODBS

Подключение к базе данных осуществляется через следующий код:

connect\_db :-

odbc\_connect('car\_db', \_Connection,

[ user('root'),

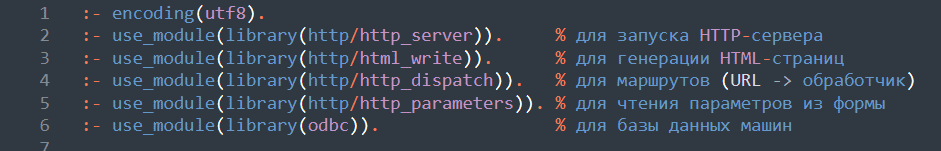
password('admin'),

alias(cardb),

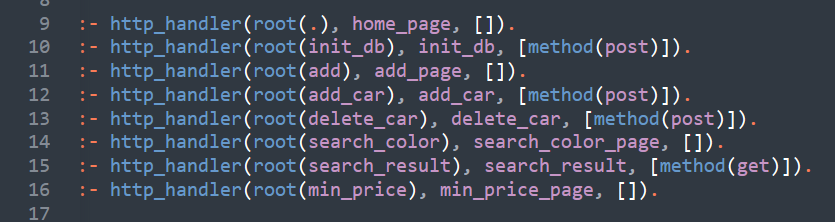
open(once)

]).

На рисунке 2 представлены используемые модули для реализации сервера:

  
Рисунок 2 – Используемые модули

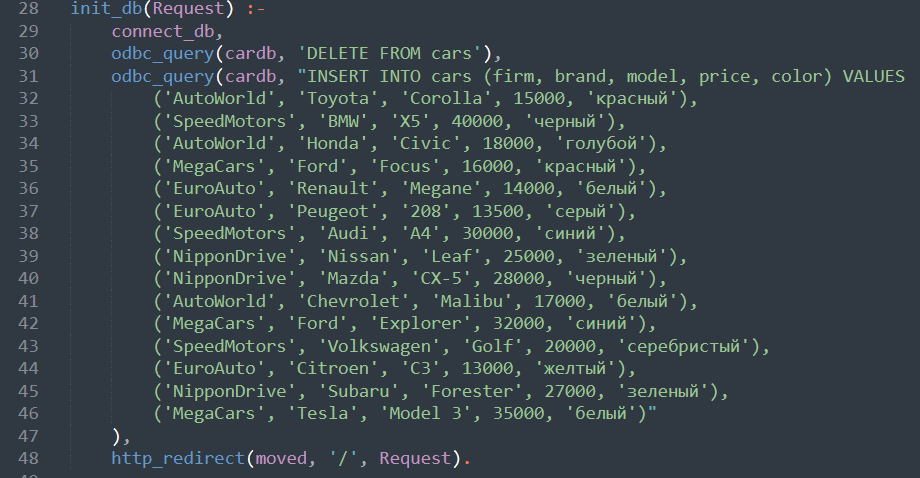
Строки, отображенные на рисунке 3, задают, какие предикаты вызываются при переходе по разным страницам:

  
Рисунок 3 – Обработчики URL

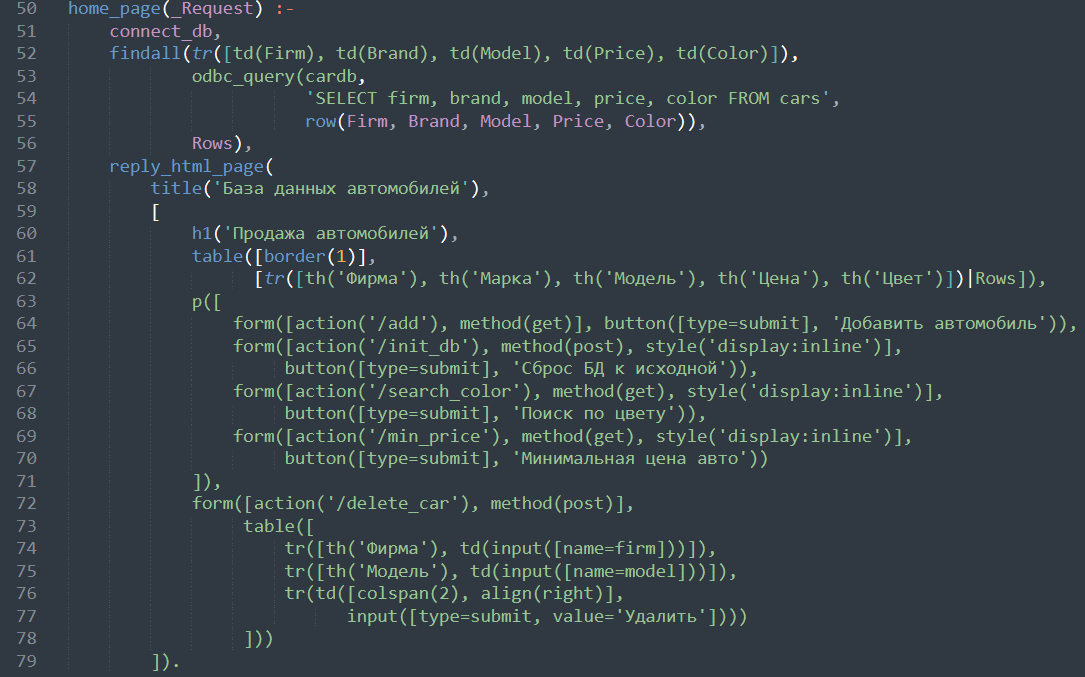
Запуск сервера описан кодом:

server :- http\_server(http\_dispatch, [port(8080)]).

Строки кода, отображенные на рисунке 4, реализуют удаление всех записей из БД и добавление стартовых данных в БД. Вызывается при нажатии **"Сброс БД к исходной".**

  
Рисунок 4 – База данных предметной области

На рисунке 5 представлена главная страница. Показывает таблицу всех машин, кнопки добавить, сброс, поиск по цвету, минимальная цена, удалить по фирме + модели.

  
Рисунок 5 – Главная страница

На рисунке 6 отображен код, где add\_page/1 реализует отображение HTML-формы для добавления новой машины, а add\_car/1 обрабатывает добавление машины из формы.

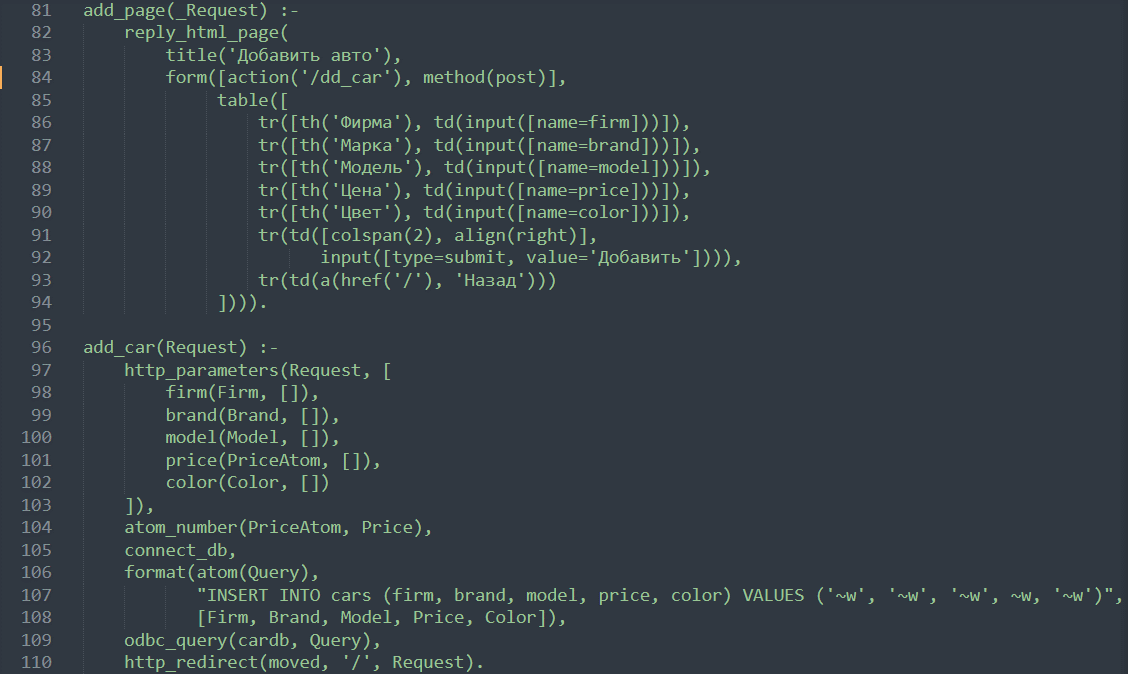


Рисунок 6 – Код для отображения и реализации добавления строк

На рисунке 7 отображены функции:

* delete\_car/1 - удаляет все машины с заданной Firm и Model;
* search\_color\_page/1 -Показывает форму поиска по цвету;
* search\_result/1 -находит все фирмы, продающие машины заданного цвета. Убирает повторы через list\_to\_set;
* min\_price\_page/1 - находит машину с минимальной ценой:

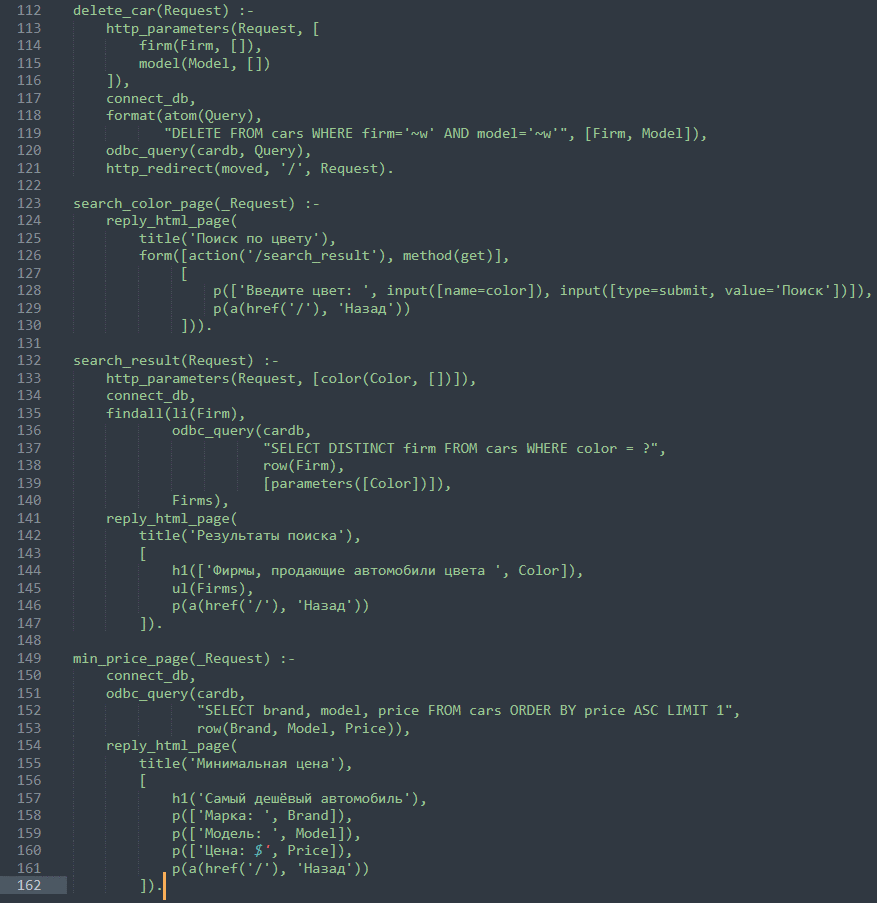
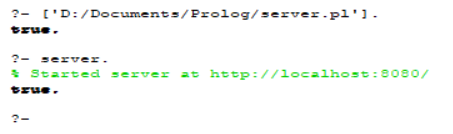
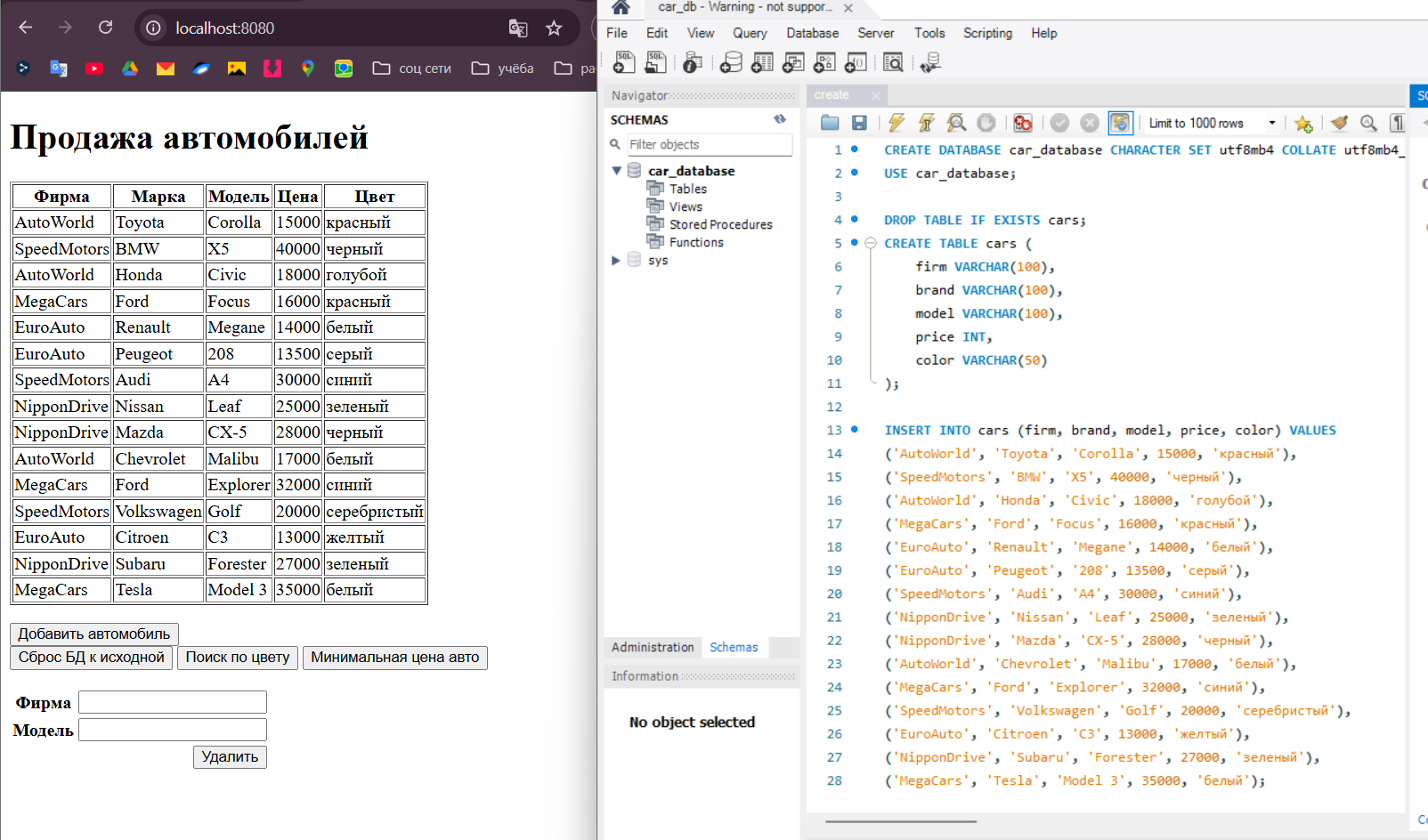


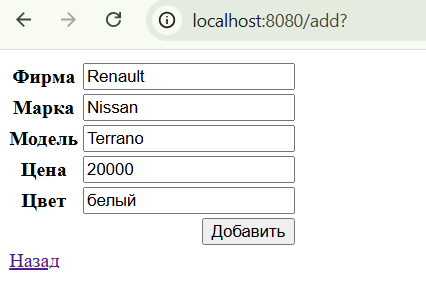
Рисунок 7 – Код с реализацией удаления строк и выводом запросов

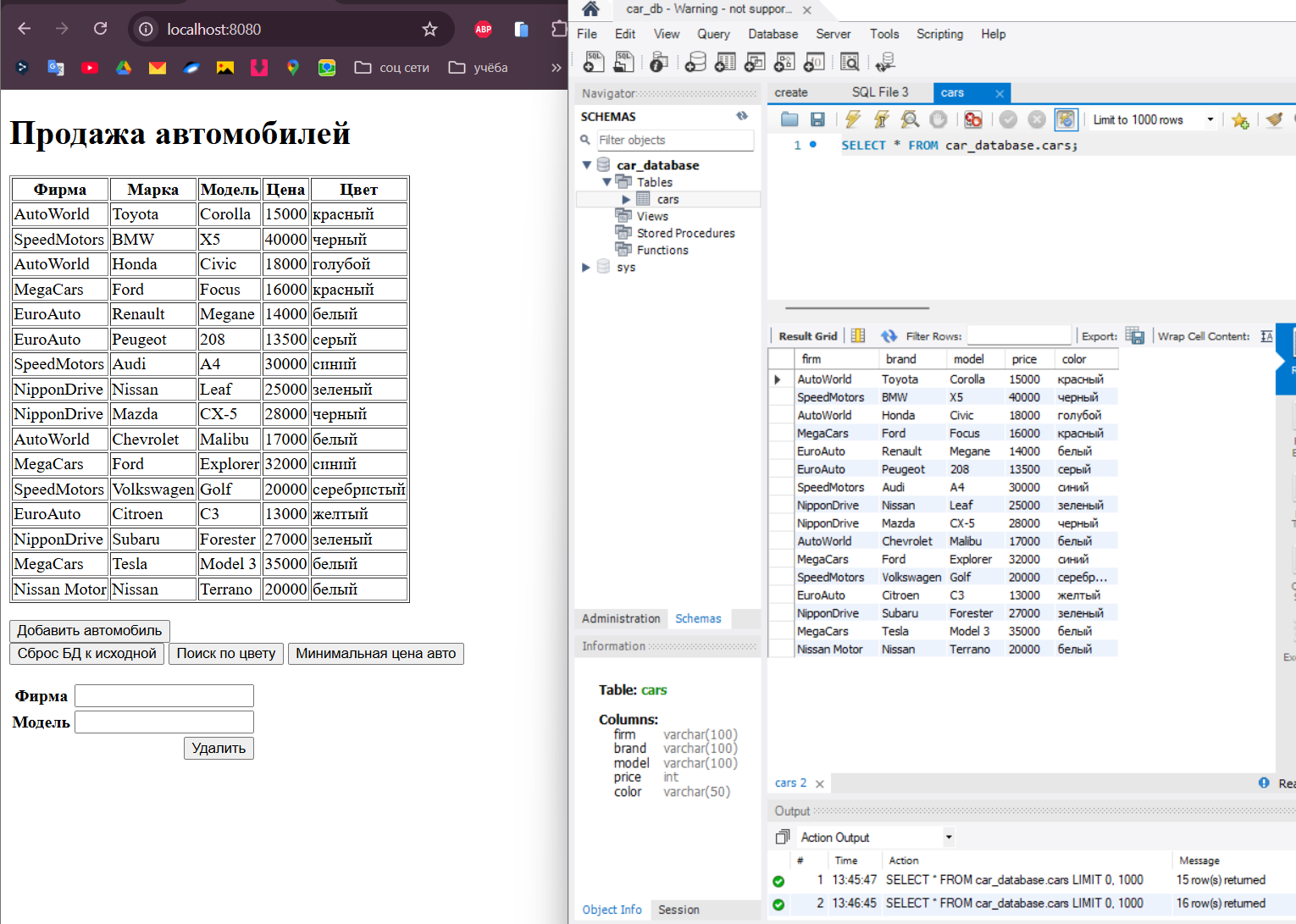
После написания кода, загружаем файл в SWI-Prolog и запускаем сервер (см. рисунок 8). Зайдя на http://localhost:8080 видим страничку, отображенную на рисунке 9. Здесь отображается стартовая таблица с БД, прописанной в скрипте на MySql.

  
Рисунок 8 – Запуск на сервер через SWI-Prolog

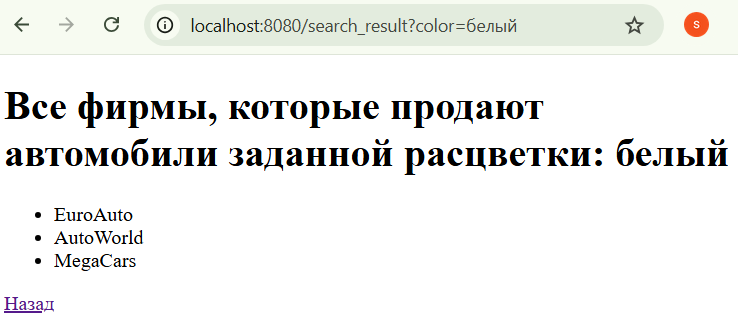
  
Рисунок 9 – Изначальная страничка на HTML

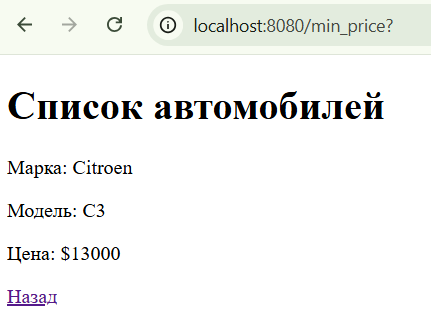
Добавим один автомобиль в базу данных (см. рисунок 10) и удалим некоторые строки таблиц, после увидим обновленную версию БД на HTML страничке и в MySql (рисунок 11).

  
Рисунок 10 – Форма для добавления нового автомобиля в таблицу

  
Рисунок 11 – Обновленная версия БД

Выполним запрос по нахождению автомобилей с определенной расцветкой (рисунок 12). Далее выполним следующий запрос для вывода списка автомобилей с минимальной ценой (рисунок 13).

  
Рисунок 12 – Выполнение первого запроса

  
Рисунок 13 – Выполнение второго запроса

После выполненных изменений вернем всё к первоначальной БД и увидим таблицу, отображенную на рисунке 14.

  
Рисунок 14 – Отображение странички после нажатия кнопки  
«Сброс к исходной базе данных»

# ЗАКЛЮЧЕНИЕ

В ходе работы изучены методы получения информации из реляционных баз данных и отображения этой информации в виде HTML страниц на языке Prolog.

На основе предыдущей лабораторной работы создана база данных, описывающая предметную область. Разработан набор SQL запросов. Написан CRUD для таблицы базы данных, предоставляющий возможность добавить и удалить определенные строки из таблицы базы данных. Также есть возможность сбросить базу данных к исходному (тестовому) варианту заполнения. Реализован вывод результатов поисковых запросов к базе данных на HTML странице.