Course Name: DATABASE AND DATA WAREHOUSING

Course Number and Section: 16:137:538

Experiment: Group Project - 1 – Assignment

Instructor: Dr. Rupa Misra

Date Submitted: 9/ 27/ 2020

Submitted by: Group1 [LU JIN, DAVID KING]

Table of Contents

Summary

This document provides an overview of the design for the Database Management System (DMBS) of an upcoming car sales website. To this end, an in-depth review is provided for the business rules and requirements which serve as the foundation of the database, the coding for construction of this database, and some examples of its application.

As shown within the Entity Relationship Diagram (ERD) on page 8, the database design is composed of multiple interlocking tables for different entities which together create the entity of a car as understood as a real-world object. Each of the tables capture different aspects such as a given vehicle's branding, it’s title, year of manufacture, or the condition of its wear and tear; the logical rules that manage handling of incoming data are also detailed (e.g., each vehicle can have only one brand, however, a given brand can have many vehicles associated).

The remainder of the paper is organized around the practical application of the theoretical design detailed within the business rules. The associated code is provided for each of the tables described within the ERD, as well as sample data to better illustrate how the database would appear in its application. Using the sample data, examples of customer queries are also detailed to illustrate the database functionality handling a variety of circumstances. Finally, a mock-up of a live version of the website hosting this database is provided as a proof of concept.

Introduction

Scenario

Ms. Smith would like to build a website to sell new cars. She has already formed a partnership with a few automobile companies. While searching on the web, she came across truecar.com and she likes this website. What she likes about truecar.com is simple and clean information flow. For example, when the user clicks on a brand (on the left: Shop New > Brand or at the bottom: brand), one can see the models. When the user clicks on any specific model, then one can see the detailed information about that specific model. But she is not totally sold on truecar.com. She is hiring your team to give her some new ideas and design the database including writing all the queries for her.

Background

Advancements within Information Technology (IT) have greatly enhanced various business processes and the communication between companies and their customers. Designing the right database management system (DBMS) is critical for ensuring efficient processing of daily operations and meeting customer demands. As each aspect of a business's operations must have an equivalent representation within an organization's database, and different applications may have unique characteristics, proper implementation of a DBMS represents a complex challenge. Our team will facilitate the selection of appropriate technologies and their implementation to facilitate the success of this new organization. The DBMS is anticipated to help Ms. Smith in leveraging higher sales by providing the right information and tools to effectively track her business and manage inventory.

Benefits of DBMS Services:

1. The database is fully functional and flexible.
2. It is very easy to use. An inquiry is easily done by users in the system.
3. The system helps the administration to manage the data efficiently.
4. A new system provides features like time efficiency to show car details and information, so it will give feedback to the administrator.

Goal

Our objective is the development of a database to handle all storage and processing of car information within a central location.

Business Rules & Requirements

Each car will have a single car brand, car model, trim, year, fuel-type, transmission-type, and condition associated with it.

Each car will have at least one color associated with it.

The “CAR” entity will capture the following attributes: the vehicle identification number, car brand, model, trim, year, color, fuel type, miles per gallon/charge cycle, cylinders, transmission, number of doors, and condition.

CAR	
PK	VIN_NUM
FK	CAR_BRAND_ID
FK	CAR_MODEL_ID
FK	TRIM_ID
FK	YEAR
FK	COLOR_ID
FK	FUEL_ID
FK	TRANSMISSION_ID
FK	CONDT_ID
	NUM_DOORS
	NUM_CYLINDERS
	RANGE

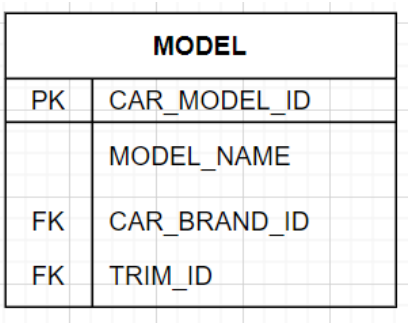
The “MAKE” entity will capture the car brand attribute--upon launch, the site will include an available range of 42 brands which are widely available within the North Am
erican market.

The range of available models are listed below:
Acura, Alfa Romeo, Audi, BMW, Buick, Cadillac, Chevrolet, Chrysler, Dodge, FIAT, Ford, Genesis, GMC, Honda, HUMMER, Hyundai, INFINITI, Jaguar, Jeep, Kia, Land Rover, Lexus, Lincoln, Maserati, Mazda, Mercedes-Benz, Mercury, MINI, Mitsubishi, Nissan, Pontiac, Porsche, Ram, Saturn, Scion, smart, Subaru, Suzuki, Tesla, Toyota, Volkswagen, and Volvo

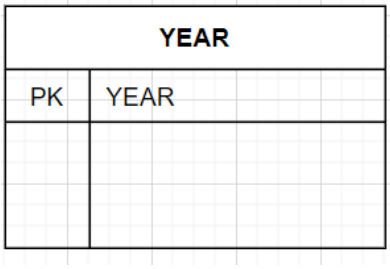
Make	
PK	CAR_BRAND_ID
	BRAND_NAME

The “MODEL” entity will capture the available car models and trim. The “Make” entity is linked to the “Model” entity via the attribute “CAR_BRAND_ID” which acts as a foreign key. There is a 1:M relationship between the entity “Make” and “Model” -- each model requires a make to be

specified; whereas, there can be an infinite number of models for a given make.

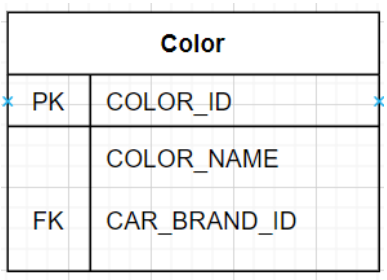


The “YEAR” entity will capture the range of years for available vehicles; it is linked to the model attribute via a M:N or many-to-many relationship: an infinite amount of car models can be a assigned to a given year, and an infinite amount of years can be assigned to a given car model--this reflects that car models are generally updated on a yearly basis.



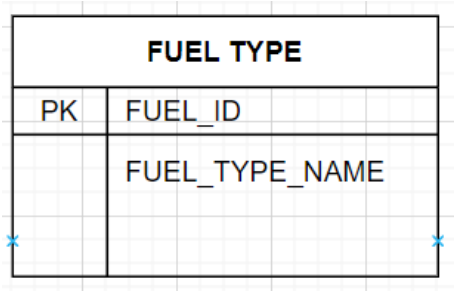
The “COLOR” entity captures the color attribute as well as the car brand under which it is offered. This entity is linked to the “Make” entity via a 1:M relationship -- each “Make” can have an infinite number of colors assigned to it; however, each color can only be assigned to a given Make. While different car manufacturers may use similar colors, they are generally given distinct names (e.g., ‘Coffee Brown’ and ‘Metallic Brown Sugar’ both being used to refer to a brown shade).

There is also a M:1 relationship between the COLOR and CAR entity -- within a given brand, it’s associated colors can be assigned to an infinite number of models; however, each unique car can only be assigned a single color.

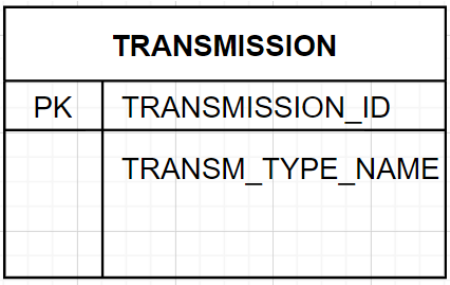


The “FUEL TYPE” entity will capture both the range of available fuel types (e.g., gasoline, diesel, hybrid, electric, hydrogen). There is a 1:1 relationship relationship between this entity and the “CAR” entity as each unique car can only be assigned one type of fuel; moreover, most car models are generally designed for a specific fuel source--this industry trend is starting to change with the introduction of hybrid models of standard cards. As such, for a limited number

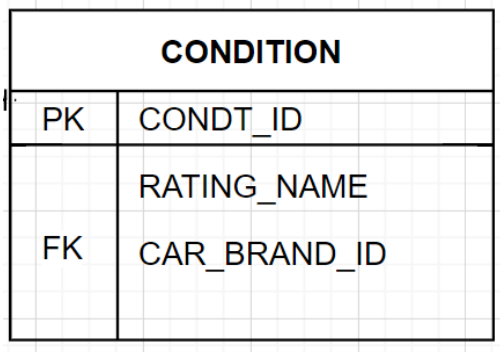
of models there is a M:N or many-to-many relationship between the available number of car models and the type of fuel source which is available (i.e., gas or hybrid).



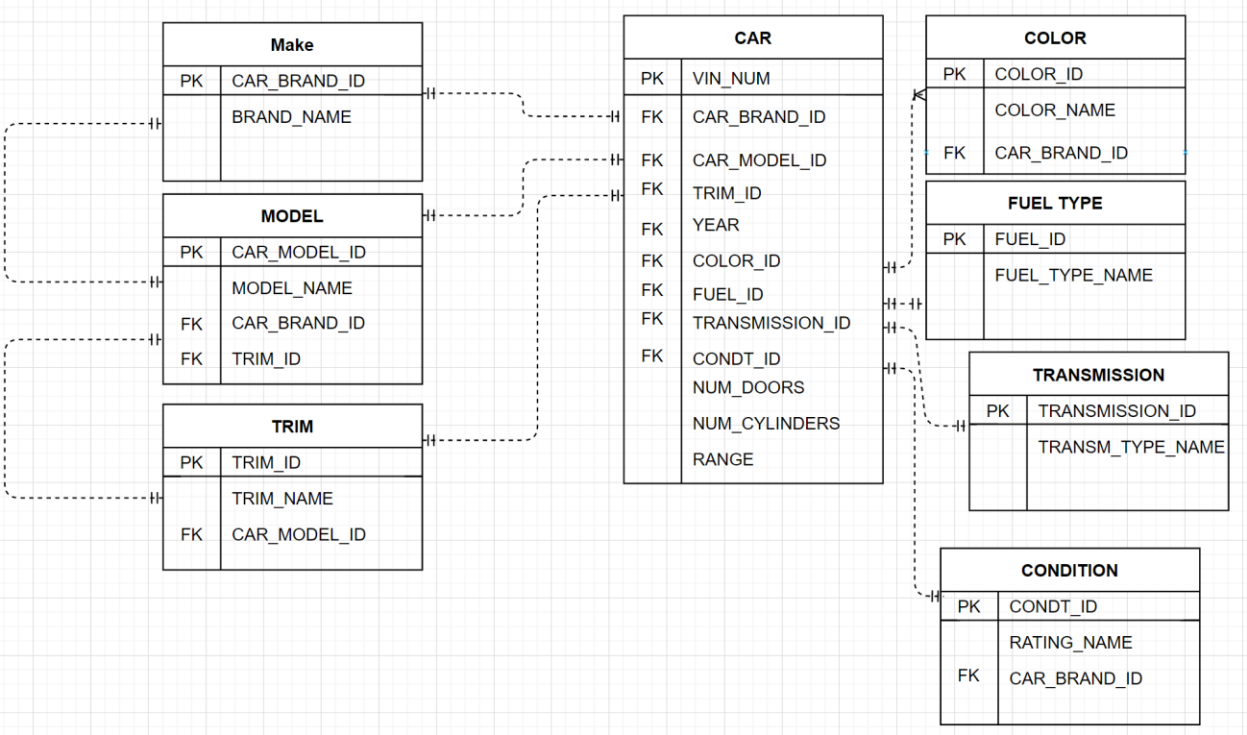
The “TRANSMISSION” entity will capture the available types of transmissions available for selection (e.g., manual or automatic).There is a 1:1 relationship relationship between this entity and the “CAR” entity as each unique car can only be assigned one type of transmission; however, there is a M:N relationship between this entity and the MODEL entity as certain models can be offered in either manual or automatic transmissions.



The “CONDITION” entity will capture generalized tiers of vehicle quality (e.g., perfect, great, fair, poor). These ratings will in turn capture general information about quality such as the amount of scratches or wear and tear a given vehicle has.



An entity relationship diagram (ERD) of the model:



Tables with data

Part(a) Tables

Table for Make

```
CREATE TABLE Make(
CAR_BRAND_ID VARCHAR(4),
BRAND_NAME VARCHAR(20),
PRIMARY KEY(CAR_BRAND_ID)
);
```

Table for TRIM

```
CREATE TABLE `TRIM`(
TRIM_ID VARCHAR(4),
TRIM_NAME VARCHAR(40),
```

```
CAR_BRAND_ID VARCHAR(4),
PRIMARY KEY (TRIM_ID),
FOREIGN KEY (CAR_BRAND_ID) REFERENCES Make(CAR_BRAND_ID)
);
```

Table for MODEL

```
CREATE TABLE MODEL(
CAR_MODEL_ID VARCHAR(4),
MODEL_NAME VARCHAR(20),
CAR_BRAND_ID VARCHAR(4),
TRIM_ID VARCHAR(4),
PRIMARY KEY (CAR_MODEL_ID),
FOREIGN KEY (CAR_BRAND_ID) REFERENCES Make(CAR_BRAND_ID),
FOREIGN KEY (TRIM_ID) REFERENCES `TRIM`(TRIM_ID)
);
```

Table for COLOR

```
CREATE TABLE COLOR(
COLOR_ID VARCHAR(4),
COLOR_NAME VARCHAR(20),
CAR_BRAND_ID VARCHAR(4),
PRIMARY KEY (COLOR_ID),
FOREIGN KEY (CAR_BRAND_ID) REFERENCES Make(CAR_BRAND_ID)
);
```

Table for FUEL_TYPE

```
CREATE TABLE FUEL_TYPE(
FUEL_ID VARCHAR(4),
FUEL_TYPE_NAME VARCHAR(20),
PRIMARY KEY(FUEL_ID)
);
```

Table for TRANSMISSION

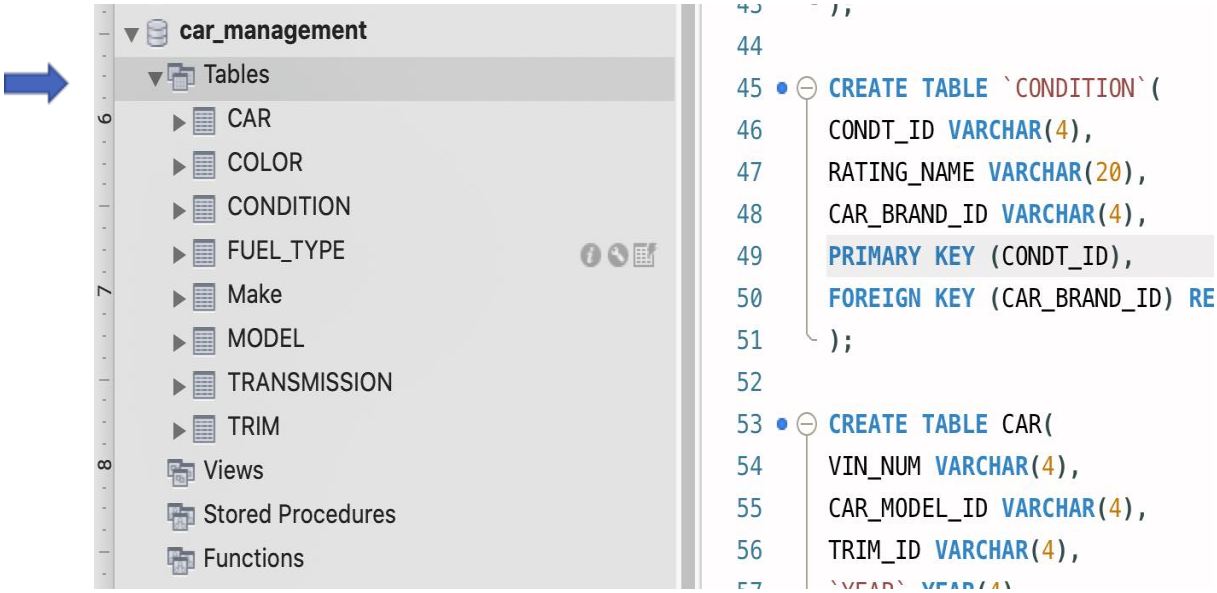
```
CREATE TABLE TRANSMISSION(
TRANSMISSION_ID VARCHAR(4),
TRANSM_TYPE_NAME VARCHAR(20),
PRIMARY KEY(TRANSMISSION_ID)
);
```

Table for CONDITION

```
CREATE TABLE `CONDITION`(
CONDT_ID VARCHAR(4),
RATING_NAME VARCHAR(20),
CAR_BRAND_ID VARCHAR(4),
PRIMARY KEY (CONDT_ID),
FOREIGN KEY (CAR_BRAND_ID) REFERENCES Make(CAR_BRAND_ID)
);
```

Table for CAR

```
CREATE TABLE CAR(
VIN_NUM VARCHAR(4),
CAR_MODEL_ID VARCHAR(4),
TRIM_ID VARCHAR(4),
`YEAR` YEAR(4),
COLOR_ID VARCHAR(4),
FUEL_ID VARCHAR(4),
TRANSMISSION_ID VARCHAR(4),
CONDT_ID VARCHAR(4),
NUM_DOORS INT,
NUM_CYLINDERS INT,
`RANGE` VARCHAR(20),
PRIMARY KEY (VIN_NUM),
FOREIGN KEY (CAR_MODEL_ID) REFERENCES MODEL(CAR_MODEL_ID),
FOREIGN KEY (TRIM_ID) REFERENCES `TRIM`(TRIM_ID),
FOREIGN KEY (COLOR_ID) REFERENCES COLOR(COLOR_ID),
FOREIGN KEY (FUEL_ID) REFERENCES FUEL_TYPE(FUEL_ID),
FOREIGN KEY (TRANSMISSION_ID) REFERENCES TRANSMISSION(TRANSMISSION_ID),
FOREIGN KEY (CONDT_ID) REFERENCES `CONDITION`(CONDT_ID)
);
```

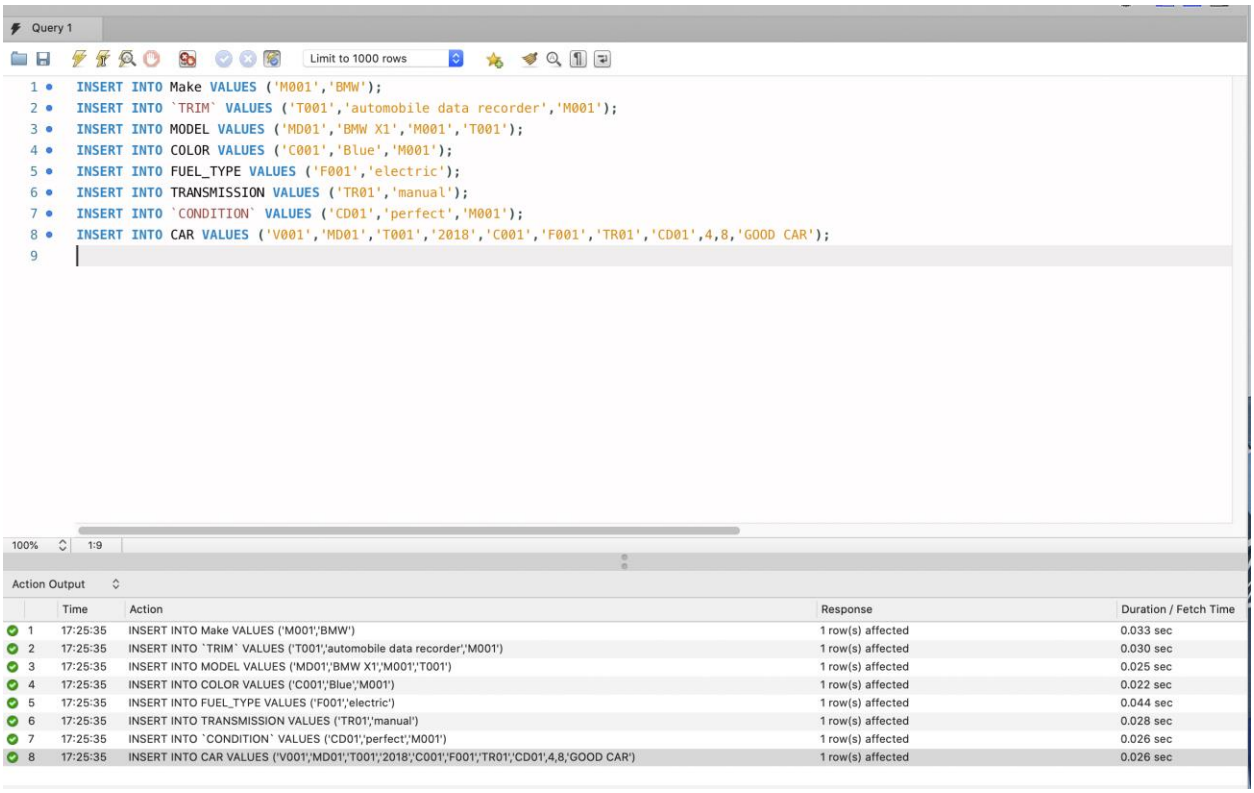


Screenshot in MySQL

Part(b) Datas

One example:

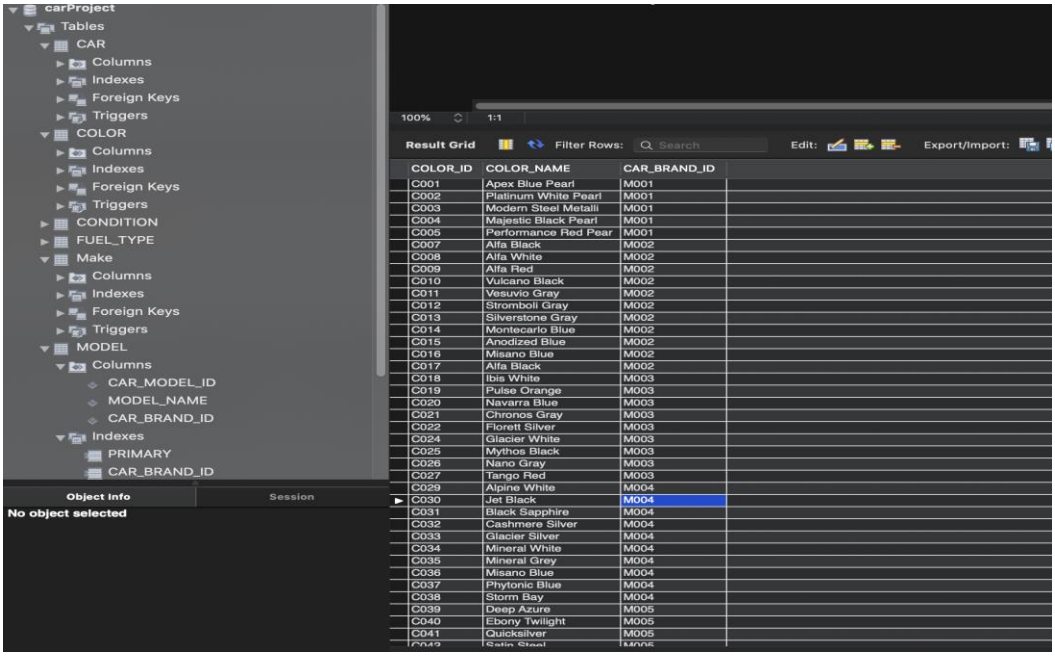
```
INSERT INTO Make VALUES ('M001','BMW');
INSERT INTO `TRIM` VALUES ('T001','automobile data recorder','M001');
INSERT INTO MODEL VALUES ('MD01','BMW X1','M001','T001');
INSERT INTO COLOR VALUES ('C001','Blue','M001');
INSERT INTO FUEL_TYPE VALUES ('F001','electric');
INSERT INTO TRANSMISSION VALUES ('TR01','manual');
INSERT INTO `CONDITION` VALUES ('CD01','perfect','M001');
INSERT INTO CAR VALUES
('V001','MD01','T001','2018','C001','F001','TR01','CD01',4,8,'GOOD CAR');
```



Screenshot in MySQL

VIN_NUM	CAR_MODEL_ID	TRIM_ID	YEAR	COLOR_ID	FUEL_ID	TRANSMISSION_ID	CONDT_ID	NUM_DOORS	NUM_CYLINDERS	RANGE
5J8TB4H35DL017718	MD04	T001	2013	C003	F001	TR01	CD02	4	6	20 City_28 HWY
ZARFANBN5L7625704	MD01	T001	2018	C001	F001	TR01	CD01	4	8	23 City_31 HWY

Screenshot for CAR table



Screenshot for COLOR table

	CONDT_ID	RATING_NAME
▶	CD01	New
	CD02	Used
	CD03	Certified
	CD04	All
*	NULL	NULL

Screenshot for CONDITION table

	FUEL_ID	FUEL_TYPE_NAME
▶	F001	Gas
	F002	Hybrid
	F003	Electric
	F004	Other

Screenshot for FUEL_TYPE table

	CAR_BRAND_ID	BRAND_NAME
▶	M001	Acura
	M002	Alfa Romeo
	M003	Audi
	M004	BMW
	M005	Buick

Screenshot for Make table

	CAR_MODEL_ID	MODEL_NAME	CAR_BRAND_ID
▶	MD01	ILX	M001
	MD02	TLX	M001
	MD03	RLX	M001
	MD04	RDX	M001
	MD05	MDX	M001
	MD06	NSX	M001
	MD07	STELVIO	M002
	MD08	GIULIA	M002
	MD09	4C SPIDER	M002
	MD10	STELVIO QUADRIFOLGIO	M002
	MD11	GIULIA QADRIFOLGIO	M002
	MD12	X1	M004
	MD13	X2	M004
	MD14	X3	M004
	MD15	X4	M004
	MD16	X5	M004
	MD17	X6	M004
	MD18	X7	M004
	MD19	2	M004
	MD20	3	M004
	MD21	4	M004
	MD22	5	M004
	MD23	7	M004
	MD24	8	M004
	MD25	Z4	M004
	MD26	i3	M004
	MD27	i8	M004
	MD28	Q3	M003
	MD29	O5	M003

Screenshot for Make table

	TRANSMISSION_ID	TRANSM_TYPE_NAME
▶	TR01	Manual
	TR02	Automatic

Screenshot for TRANSMISSION table

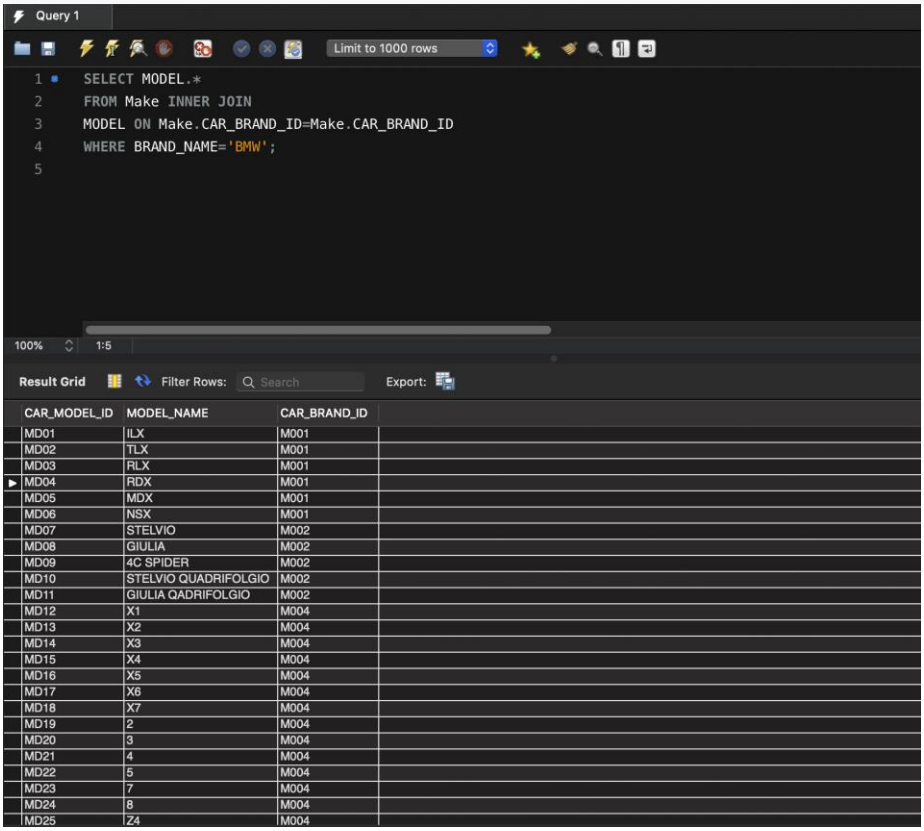
SQL Queries to retrieve data

---

1. When the user clicks on a certain brand (let's say BMW) then the models for that specific brand is displayed

```
SELECT MODEL.*
```

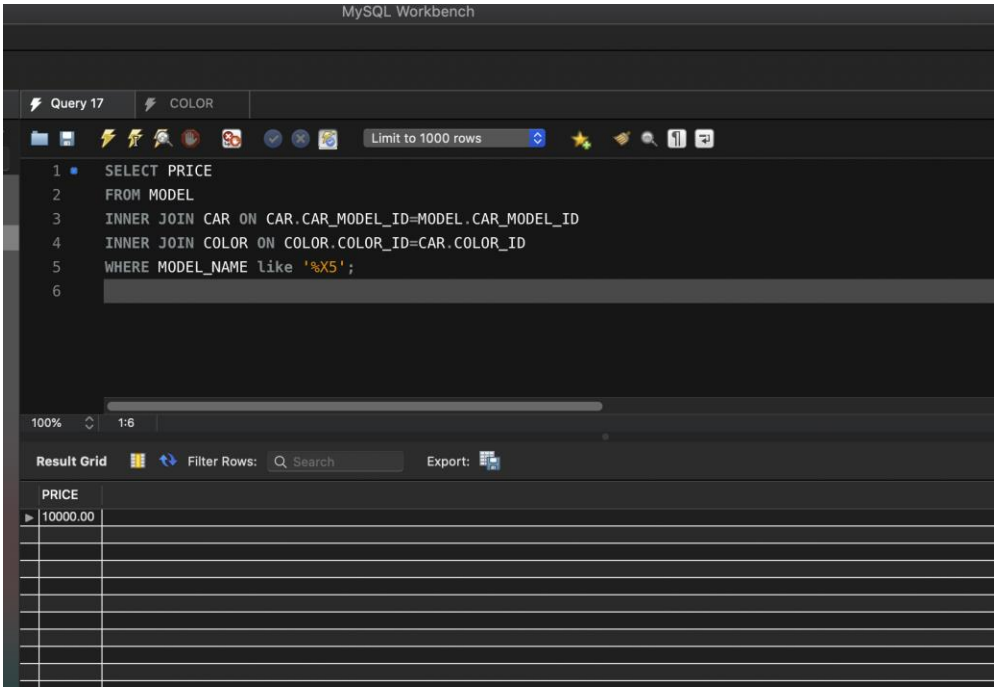
```
FROM Make INNER JOIN
MODEL ON Make.CAR_BRAND_ID=Make.CAR_BRAND_ID
WHERE BRAND_NAME='BMW';
```



CAR_MODEL_ID	MODEL_NAME	CAR_BRAND_ID
MD01	ILX	M001
MD02	TLX	M001
MD03	RLX	M001
MD04	RDX	M001
MD05	MDX	M001
MD06	NSX	M001
MD07	STELVIO	M002
MD08	GIULIA	M002
MD09	4C SPIDER	M002
MD10	STELVIO QUADRIFOLGIO	M002
MD11	GIULIA QADRIFOLGIO	M002
MD12	X1	M004
MD13	X2	M004
MD14	X3	M004
MD15	X4	M004
MD16	X5	M004
MD17	X6	M004
MD18	X7	M004
MD19	2	M004
MD20	3	M004
MD21	4	M004
MD22	5	M004
MD23	7	M004
MD24	8	M004
MD25	Z4	M004

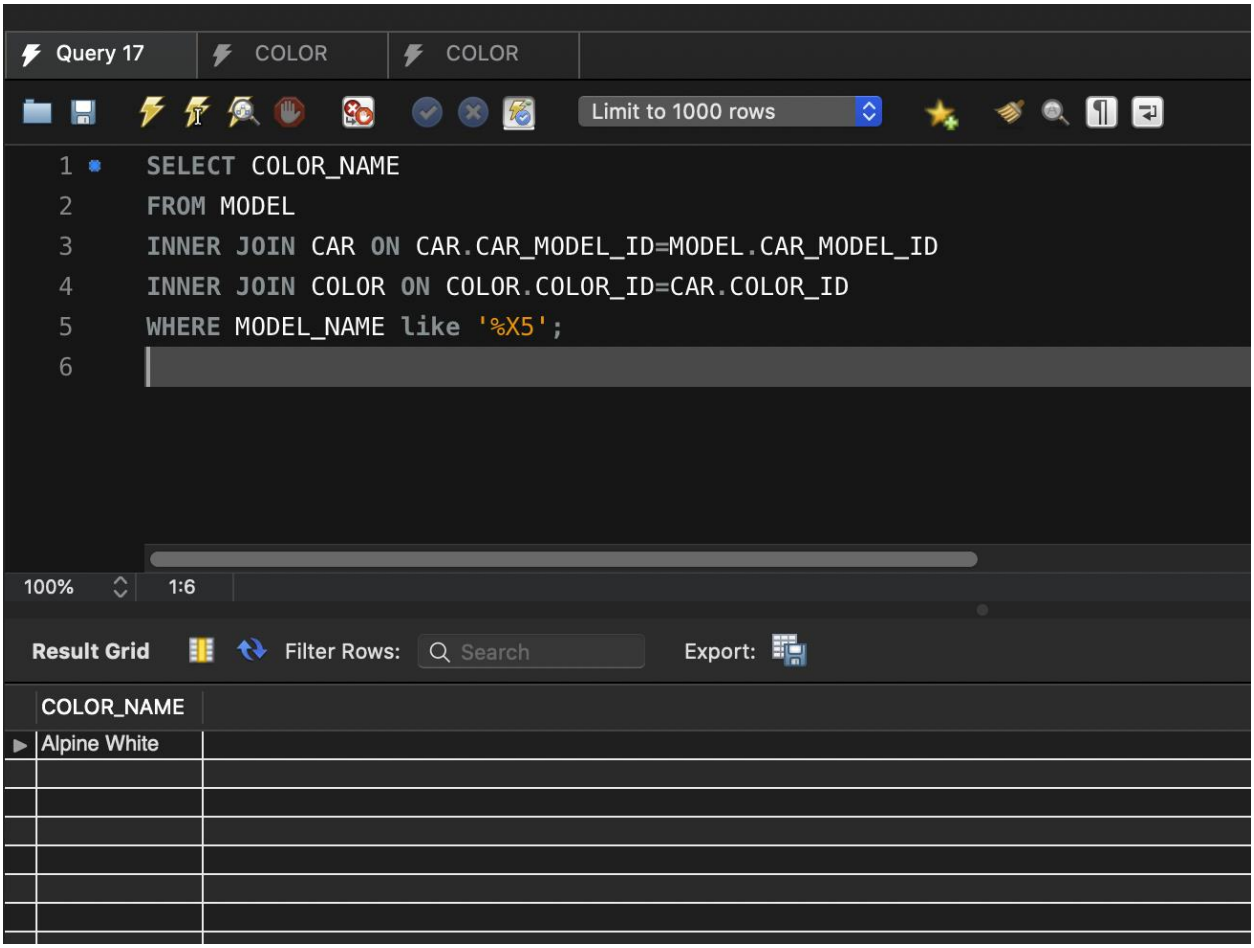
2. When the user clicks on a certain model (Let's say X5), then the specific information including the price regarding that model is displayed

```
SELECT PRICE
FROM MODEL
INNER JOIN CAR ON CAR.CAR_MODEL_ID=MODEL.CAR_MODEL_ID
INNER JOIN COLOR ON COLOR.COLOR_ID=CAR.COLOR_ID
WHERE MODEL_NAME like '%X5';
```



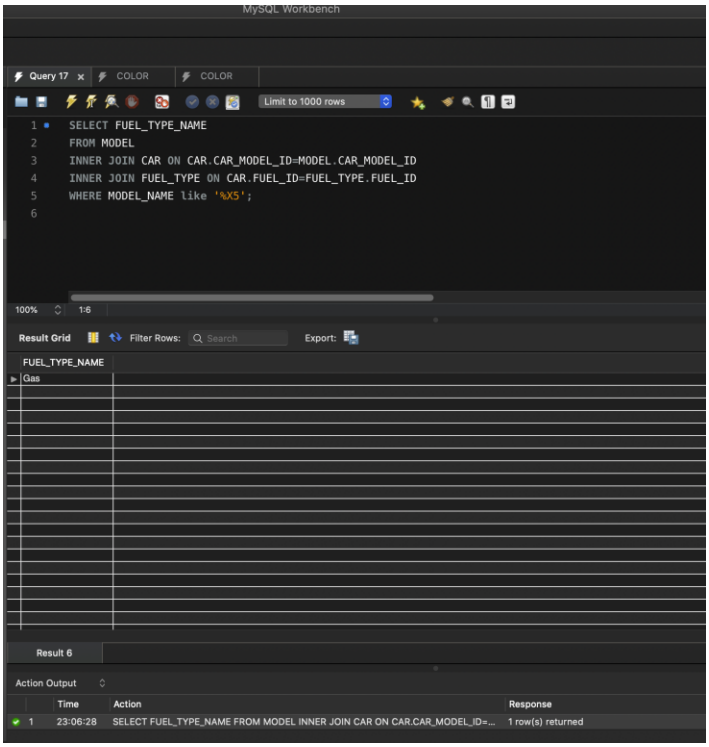
3. When the user clicks on the Exterior of a certain model (Let's say X5) then all the color options are displayed

```
SELECT COLOR_NAME
FROM MODEL
INNER JOIN CAR ON CAR.CAR_MODEL_ID=MODEL.CAR_MODEL_ID
INNER JOIN COLOR ON COLOR.COLOR_ID=CAR.COLOR_ID
WHERE MODEL_NAME like '%X5';
```



4. When a user clicks on the “Fuel” option for a specific model the available fuel options are displayed.

```
SELECT FUEL_TYPE_NAME
FROM MODEL
INNER JOIN CAR ON CAR.CAR_MODEL_ID=MODEL.CAR_MODEL_ID
INNER JOIN FUEL_TYPE ON CAR.FUEL_ID=FUEL_TYPE.FUEL_ID
WHERE MODEL_NAME like '%X5';
```



5. When a user searches a specific model and filters for condition, the available inventory is displayed.

Without Filtering

```
SELECT * FROM CAR
WHERE CAR_MODEL_ID = (SELECT CAR_MODEL_ID FROM MODEL WHERE
MODEL_NAME = "ILX");
```

VIN_NUM	CAR_MODEL_ID	TRIM_ID	YEAR	COLOR_ID	FUEL_ID	TRANSMISSION_ID	CONDT_ID	NUM_DOORS	NUM_CYLINDERS	RANGE	PRICE
JTDEBRBE6LJ015002	MD01	T001	2020	C001	F002	TR02	CD03	4	6	53CITY_52HWY	20000.00
V001	MD01	T001	2018	C001	F001	TR01	CD01	4	8	GOOD CAR	1000.00

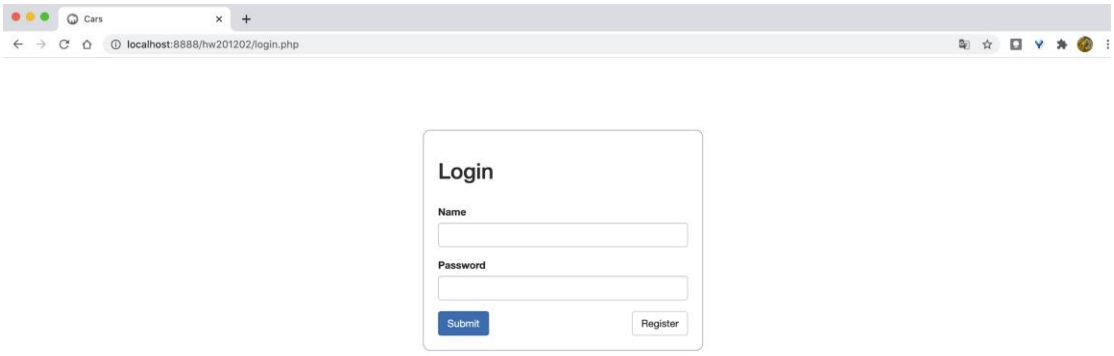
With Filtering

```
SELECT * FROM CAR
WHERE CAR_MODEL_ID = (SELECT CAR_MODEL_ID FROM MODEL WHERE
MODEL_NAME = "ILX") AND CAR.CONDT_ID = "CD03";
```

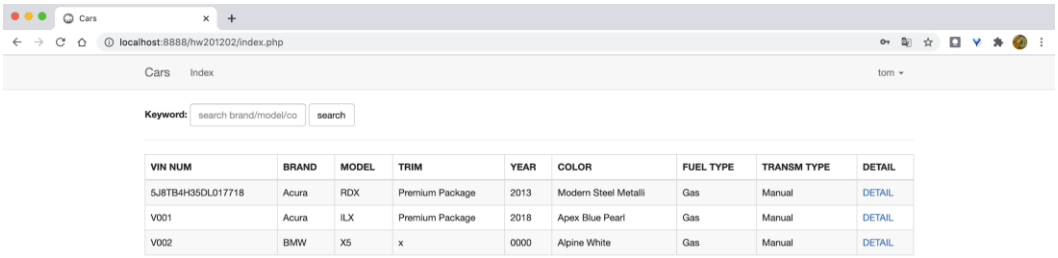
VIN_NUM	CAR_MODEL_ID	TRIM_ID	YEAR	COLOR_ID	FUEL_ID	TRANSMISSION_ID	CONDT_ID	NUM_DOORS	NUM_CYLINDERS	RANGE	PRICE
JTDEBRBE6LJ015002	MD01	T001	2020	C001	F002	TR02	CD03	4	6	53CITY_52HWY	20000.00

Database application

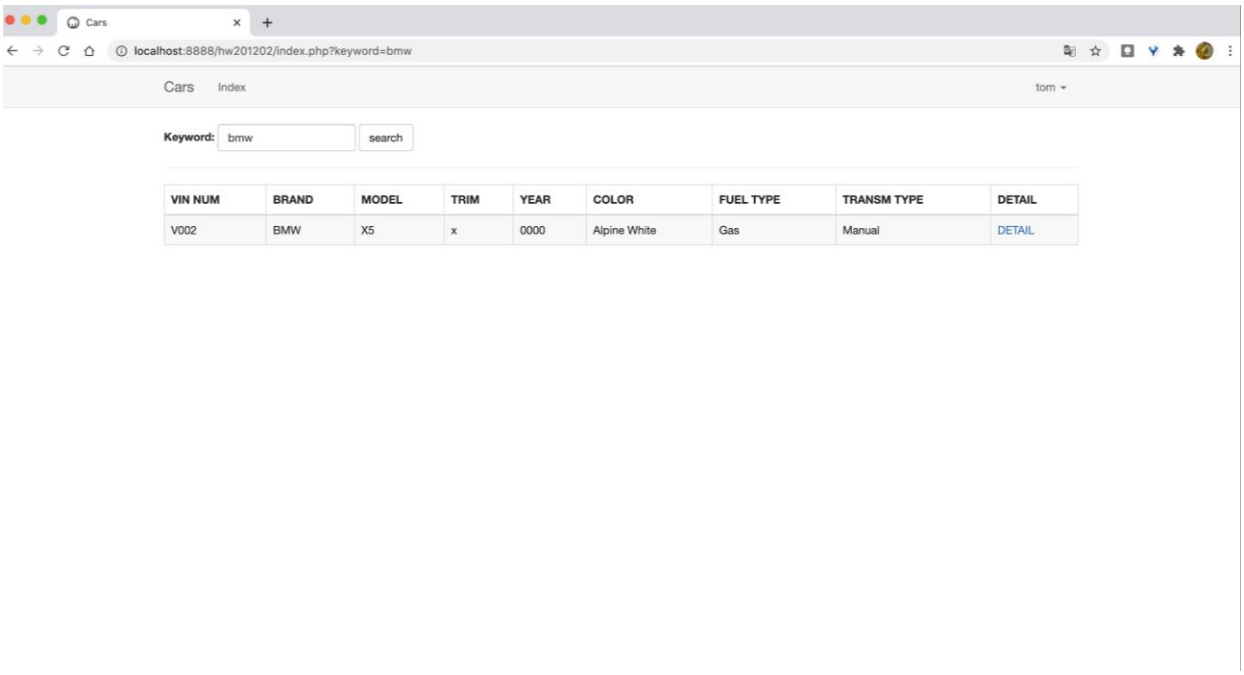
Login in page:



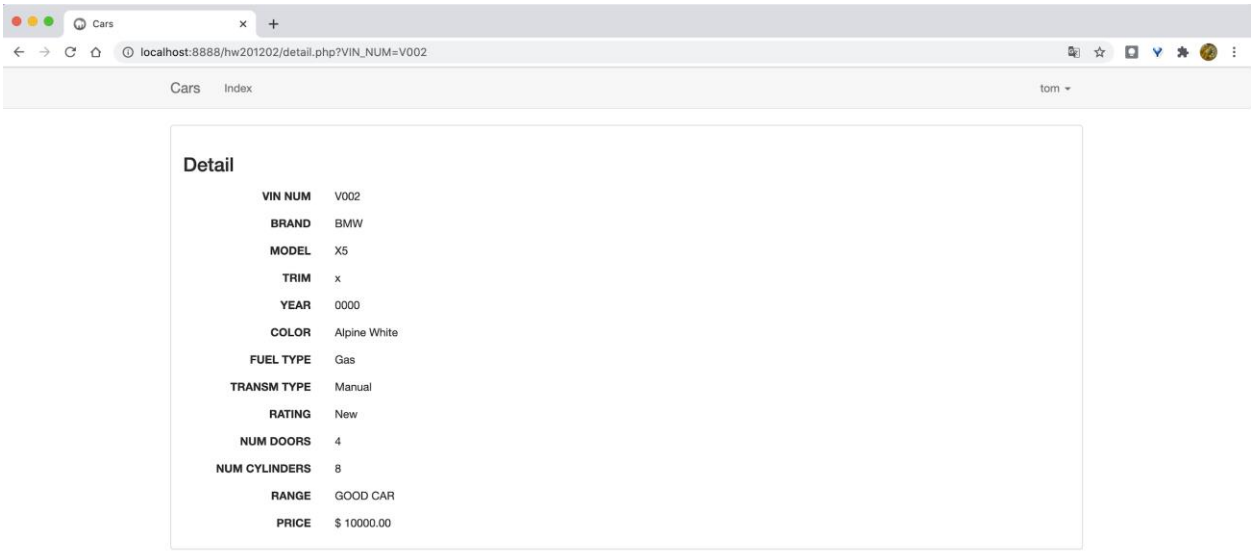
Dashboard:



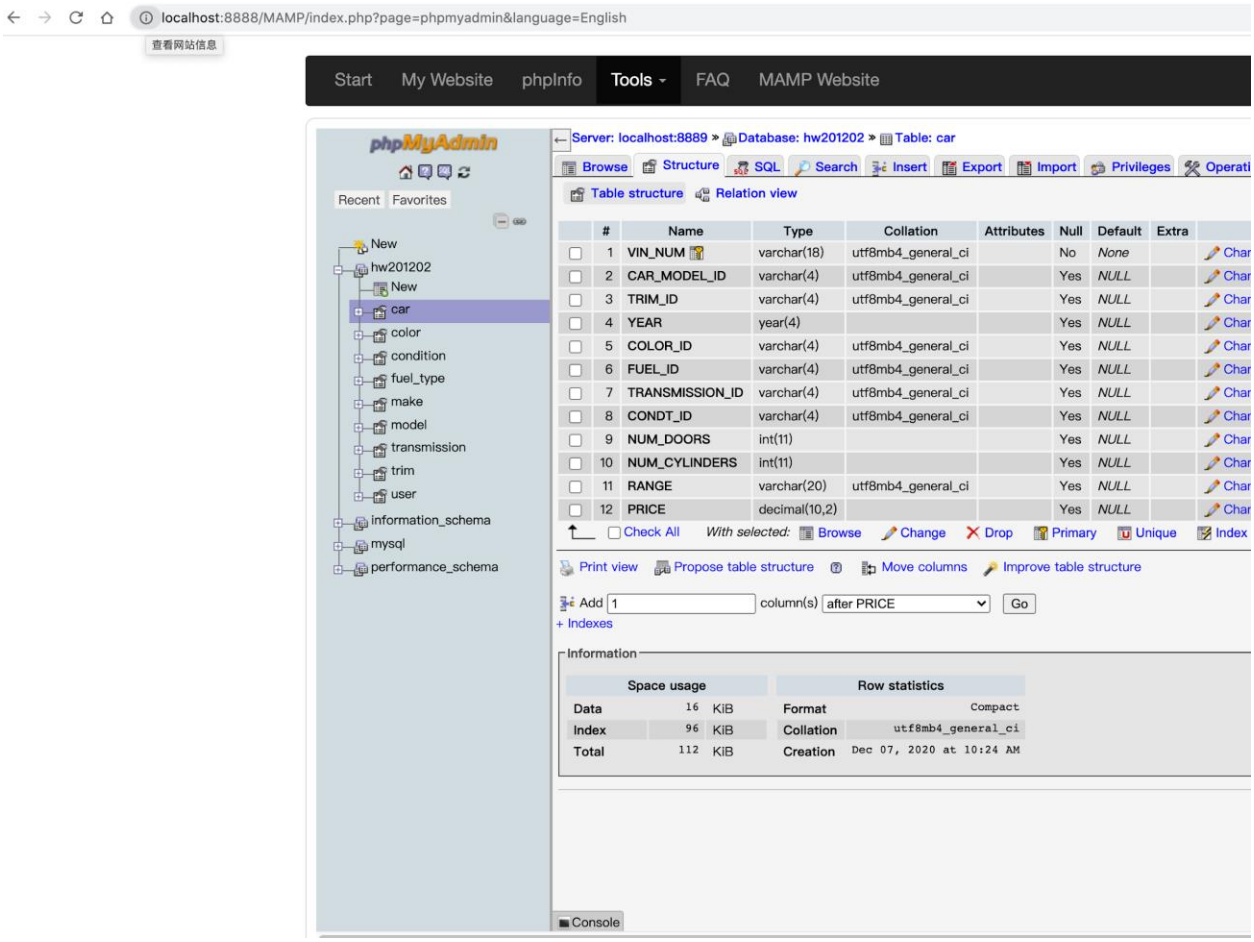
Search:



Information:



PHP-admin page


localhost:8888/MAMP/index.php?page=phpmyadmin&language=English
查看网站信息
Start
My Website
phpInfo
Tools
FAQ
MAMP Website
phpMyAdmin
Recent
Favorites
New
hw201202
New
car
color
condition
fuel_type
make
model
transmission
trim
user
information_schema
mysql
performance_schema
Server: localhost:8889 » Database: hw201202 » Table: car
Browse
Structure
SQL
Search
Insert
Export
Import
Privileges
Table structure
Relation view
Name Type Collation Attributes Null Default Extra
1 VIN_NUM varchar(18) utf8mb4_general_ci No None
2 CAR_MODEL_ID varchar(4) utf8mb4_general_ci Yes NULL
3 TRIM_ID varchar(4) utf8mb4_general_ci Yes NULL
4 YEAR year(4) Yes NULL
5 COLOR_ID varchar(4) utf8mb4_general_ci Yes NULL
6 FUEL_ID varchar(4) utf8mb4_general_ci Yes NULL
7 TRANSMISSION_ID varchar(4) utf8mb4_general_ci Yes NULL
8 CONDT_ID varchar(4) utf8mb4_general_ci Yes NULL
9 NUM_DOORS int(11) Yes NULL
10 NUM_CYLINDERS int(11) Yes NULL
11 RANGE varchar(20) utf8mb4_general_ci Yes NULL
12 PRICE decimal(10,2) Yes NULL
Check All
With selected:
Browse
Change
Drop
Primary
Unique
Index
Print view
Propose table structure
Move columns
Improve table structure
Add 1 column(s) after PRICE Go
+ Indexes
Information
Space usage
Data 16 KiB
Index 96 KiB
Total 112 KiB
Row statistics
Format Compact
Collation utf8mb4_general_ci
Creation Dec 07, 2020 at 10:24 AM
Console

References

1. https://medium.com/@chrismore/vehicle-database-spreadsheet-6f248229aae4

Appendix