

**Group Project Assignment 5**

**Group Name: The Best**



**Group Members: Jinglin Zhao, Shiv Patel, Zeel Shah**

## **Content**

- Summary
- Introduction
- Business Rules + Requirements
- Develop the ER diagram
- Tables with data
- SQL Queries to retrieve data
- Database Application
- References
- Appendix

## **Summary**

Our project is a cumulative research and developmental project that consisted of database retrieval, database development, and database execution. We created a company called EvCars that implemented a website's infrastructure but in more of a database type of format. The organization's goal is to employ a data-driven online platform for operating technology support and effective database management to optimize the online services for searching EV cars. With the capability and expansion of the Internet, web applications have become an increasingly important tool for business. Some examples of the most common uses are communication with customers, collaboration with employees, secure storage of data, and also providing data and information to management. We have deployed a system that not only mocks the database of a prominent car dealer website like Cars.com, but also developed a system in which data can be extracted using simple SQL queries.

With Covid-19 being continuing to plague our world, our team met virtually through Zoom. Team meetings were held on a week to week basis to discuss accessible data for extraction, report findings as discovered and collaborate on deliverables. The team designed a clearly defined project plan that was reviewed and updated based on available and accessible data. This was executed primarily through effective communication and the ability to identify what the business wants to get out of the system.

The Database we developed was built in MySQL in the first few stages, and then the team transitioned to multiuser databases with the use of AWS, MySQL, PHP, and XAMPP. The database applications are used to search, sort, filter, and present information based upon web requests from users. This can also contain code to perform mathematical and statistical calculations on the data to support queries submitted from web browsers. The application server then works with databases by communicating through the intermediary of a database driver. After the driver establishes communication, the query is executed against the database, and a record set is created. The record set is returned to the application server and the data used in the dynamic page.

After performing several SWOT analyses and basic sitemaps, we developed the necessary business rules and requirements for all the entities based on the features and functionalities of the website. Within the Business Rules and Requirements, we implemented relationships like 1:1, 1:M, and M:N that were specifically developed based on the website infrastructure. According to the relationships among various tables, we created Entity Relationship Diagrams(ERD) that were created based on the specific entity table. From a high level, our entity tables include brand, model, body, tire,

camera, seat covers, window tint, stereo systems, DVD players, engine, and battery. With the excess amount of entities in our database, we had to carefully decide and develop the attributes per table and try not to duplicate any unnecessary attributes. Overall, depending on the relevance and correlation of each entity table, important attributes were used and implemented into building tables.

## Introduction

We have \$500,000 from Ms.Smith for building an EV car service website. Our company name is MyEvcars. The name includes 'My' for customers to have a sense of belongingness, 'EVcars' for emphasizing the type of services. Our missions are to provide the best buying experience for consumers, be the trusted resource for people, and give dealers the most effective selling way. We will use a data-driven online platform for operating technology support and employ effective database management for optimizing our online services. Our company values include:

- Asking why.
- Working as a coalition.
- Listening first.
- Serving passionately.
- Building a trusted and market-learning source for EV cars.
- Reducing greenhouse gas emissions.



*Figure 1. Company Logo*

To set up a good platform, we did a SWOT(strengths, weaknesses, opportunities, and threats) analysis for building on what the company does well, address what the company is lacking, minimize risks, and make the most significant possible advantage of success opportunities. After that, we created a basic sitemap to learn information flow from competitors' websites and plan the new website as a map.

We analyzed truecar.com and concluded several functionalities we could use in our website, including testimonials from previous customers, Friendly animations throughout the homepage, bottom-line information, excellent sort features, check availability button, clear comparison page, and a section for building customer community. Based on all the features and functionalities our company wants and the information obtained from interviews with several car website users, we wrote up the

necessary business rules for all the entities. The entities include car, make, model, customer, and car features, which will be included in the database.

### **Business Rules:**

1. A car can have only one manufacturer. One manufacturer can make multiple cars.
2. A car should have a unique VIN id.
3. A car can be assigned to only one type of model. One type of model can be assigned to multiple cars.
4. A car can have one type of battery. The same types of batteries can be installed in multiple cars.
5. A car can have one color, type of car assigned from an Option entity. A one color or type of car from Option entity can be assigned to different cars.
6. A car can be assigned to only one brand. One brand can have multiple cars.
7. A manufacturer can have multiple brands. One brand can have multiple manufacturers.
8. A car with a unique ID can be sold to only one customer. A customer can buy multiple cars.
9. Once the car is sold, that car should be allocated to only one purchase ID.
10. A one purchase should be assigned to one customer. A customer can have multiple purchases.
11. A review should be written by only one customer. A customer should be allowed to write reviews as per the number of purchased cars.

### **Business rules' requirements :**

- Customer entity will be identified by the ID. This will have a first name, last name, email, initials, address.
- Car entity will be identified by the Car ID. This entity will have used/new options.
- Car model will be identified by the Model ID. This entity will have car type, manufacture, option and model types.
- Manufacturer entity will be identified with make id. This entity will have make year, address and phone number.
- Brand entity will be identified with Brand ID. This entity will have brand name, brand initials, brand logo.
- Purchase/Transaction will be identified by purchase ID. The entity will have purchase date, purchase time, and purchase type.
- VIN entity will be identified with VIN ID. This will be unique and will be different and will be associated with each and every type of the car.
- Battery entity will be identified with battery ID. The entity will have battery type, warranty date and price.

- Options entity will be identified with option ID. This will have options like car type to choose, color to choose.
- Review entities will be identified with review ID. This entity will have review date and review content.

All these entities will have many foreign keys assigned to identify/ link other information.

## Requirements

### Website Requirements:

1. Multiple users should be able to use this site at the same time.
2. A user should be able to filter as per his/her own requirements for the car.
3. This website should flash news for any new deals.
4. This website should have all detailed information for each listed car.
5. This website should have live customer chat.

### User Stories (requirements) :

1. A user can visit the site without signing up.
2. A user can login to this site multiple times.
3. A user can filter Cars using the site's dashboard by multiple options
4. A user can perform comparison by selecting two or more types of cars.
5. A user can choose only one brand at a time.
6. After selecting, a user can choose only one model at a time.
7. After choosing a model, the user can make modifications to models if the company allows.
8. A user can add multiple cars in the cart.
9. A user can buy multiple cars.
10. A user shouldn't be able to sell the car.
11. A user can write reviews once purchase is made.

## Develop the ER diagram

### Brands

-Attributes

- **Brand ID (PK)**
- Brand name
- Brand Type
- Brand address
- Country
- City

- Zip Code
- Phone
- Fax
- Number of employees

## Models

-Attributes :

- **Model ID (PK)**
- Model Name
- Model Number
- Model type
- Model year
- Model Manufacturer

## Body

- **Body\_ID (PK)**
- **Body\_Name**
- **Body\_Number**
- Body\_Maker
- Body\_Year
- Color
  - **Tires type**
    - **Tires\_ID (PK)**
    - **Tires\_Name**
    - **Tires\_Number**
    - **Tires\_Maker**
    - **Tires\_Year**
  - **Camera**
    - **Camera\_ID**
    - **Camera\_Type**
    - **Camera\_Resolution**
    - Camera\_Maker
    - Camera\_Year
  - **Seat Covers**
    - **Seat Covers\_ID (PK)**
    - **Seat Covers\_Name**
    - **Seat Covers\_Number**
    - Seat Covers\_Maker
    - Seat Covers\_Year
  - **Window Tint**

- **Window Tint\_ID (PK)**
- **Window Tint\_Name**
- **Window Tint\_Number**
- Window Tint\_Maker
- Window Tint\_Year

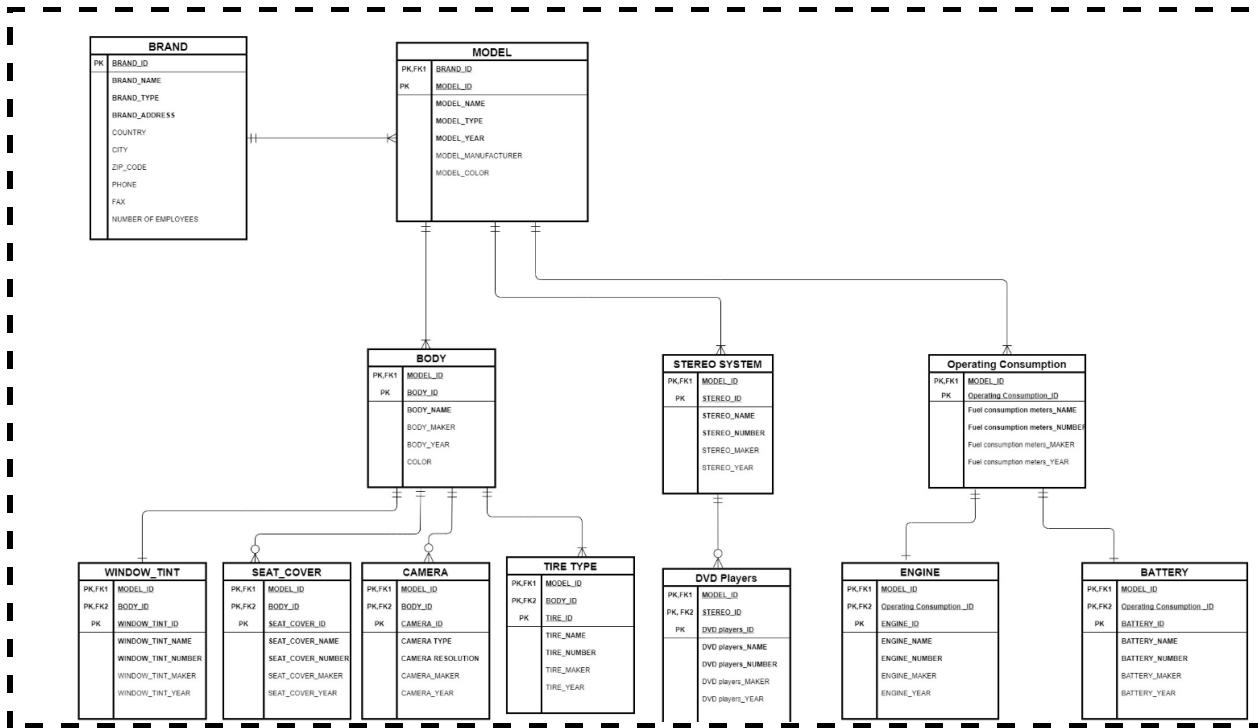
#### **-Stereo System**

- **Stereo System\_ID (PK)**
- **Stereo System\_Name**
- **Stereo System\_Number**
- Stereo System\_Maker
- Stereo System\_Year
  - DVD players
    - **DVD players\_ID (PK)**
    - **DVD players\_Name**
    - **DVD players\_Number**
    - DVD players\_Maker
    - DVD players\_Year

#### **-Operating consumption meters**

- **Operating consumption meters\_ID (PK)**
- **Operating consumption meters\_Name**
- **Operating consumption meters\_Number**
- Operating fuel consumption meters\_Maker
- Operating fuel consumption meters\_Year
  - **Engine**
    - **Engine\_ID (PK)**
    - **Engine\_Name**
    - **Engine\_Number**
    - Engine\_Maker
    - Engine\_Year
  - **Battery**
    - **Battery \_ID (PK)**
    - **Battery \_Name**
    - **Battery \_Number**
    - Battery \_Maker
    - Battery \_Year

#### **ERD for MyEVCars**



## Tables with data

### Brand

```
CREATE TABLE BRAND (
    BRAND_ID CHAR(3) NOT NULL,
    BRAND_NAME VARCHAR(15) NOT NULL,
    BRAND_TYPE VARCHAR(15) NOT NULL,
    BRAND_ADDRESS VARCHAR(255),
    COUNTRY VARCHAR(255),
    CITY VARCHAR(255),
    ZIP_CODE VARCHAR(10),
    PHONE VARCHAR(15),
    FAX VARCHAR(15),
    PRIMARY KEY (BRAND_ID)
);
```

### INSERT INTO BRAND

```
(BRAND_ID, BRAND_NAME, BRAND_TYPE, BRAND_ADDRESS, COUNTRY, CITY, ZIP_CODE, PHONE, FAX)
```

```
VALUES ('001', 'Toyota', 'Japan', 'Toyota Motor Sales, U.S.A.,  
Inc. Plano, TX 75025-9001',  
'US','Plano','75023-9001','800-331-4331', '310-468-7814') ;  
  
INSERT INTO BRAND  
(BRAND_ID,BRAND_NAME,BRAND_TYPE,BRAND_ADDRESS,COUNTRY,CITY,ZIP_C  
ODE,PHONE,FAX)  
  
VALUES ('002', 'Mercedes-Benz', 'Germany', 'BMW of North  
America, LLC. 300 Chestnut Ridge Road. Woodcliff Lake, NJ  
07677-7731', 'US','Woodcliff Lake','07677-7731','800-831-1117',  
'877-281-2087') ;  
  
INSERT INTO BRAND  
(BRAND_ID,BRAND_NAME,BRAND_TYPE,BRAND_ADDRESS,COUNTRY,CITY,ZIP_C  
ODE,PHONE,FAX)  
  
VALUES ('003', 'BMW', 'Germany', 'Mercedes-Benz USA,  
LLC.Customer Assistance Center 1 Mercedes-Benz Drive,Sandy  
Springs, GA 30328', 'US','Sandy  
Springs','30328','800-367-6372','201-307-9286') ;  
  
INSERT INTO BRAND  
(BRAND_ID,BRAND_NAME,BRAND_TYPE,BRAND_ADDRESS,COUNTRY,CITY,ZIP_C  
ODE,PHONE,FAX)  
  
VALUES ('004', 'FORD', 'America', 'Ford Motor Company, Customer  
Relationship Center, P.O. Box 6248, Dearborn, MI 48126',  
'US','Dearborn','48126','800-392-3673', '800- 392-3673') ;  
  
INSERT INTO BRAND  
(BRAND_ID,BRAND_NAME,BRAND_TYPE,BRAND_ADDRESS,COUNTRY,CITY,ZIP_C  
ODE,PHONE,FAX)  
  
VALUES ('005', 'NISSAN', 'Japan', 'P.O. Box 685003 Franklin, TN  
37068-5003', 'US','Franklin','37068-5003','800-647-7261',  
'800-456-6622') ;  
  
INSERT INTO BRAND  
(BRAND_ID,BRAND_NAME,BRAND_TYPE,BRAND_ADDRESS,COUNTRY,CITY,ZIP_C  
ODE,PHONE,FAX)  
  
VALUES ('006', 'TESLA', 'America', '860 Washington st,  
NY,10014', 'US','New York','10014','212-206-1204',  
'800-456-6622') ;
```

	BRAND_ID	BRAND_NAME	BRAND_TYPE	BRAND_ADDRESS	COUNTRY	CITY	ZIP_CODE	PHONE	FAX
▶	001	Toyota	Japan	Toyota Motor Sales, U.S.A., Inc. Plano, TX 750...	US	Plano	75023-9001	800-331-4331	310-468-7814
	002	Mercedes-Benz	Germany	BMW of North America, LLC. 300 Chestnut Ridge...	US	Woodcliff Lake	07677-7731	800-831-1117	877-281-2087
	003	BMW	Germany	Mercedes-Benz USA, LLC. Customer Assistance ...	US	Sandy Springs	30328	800-367-6372	201-307-9286
	004	FORD	America	Ford Motor Company, Customer Relationship C...	US	Dearborn	48126	800-392-3673	800-392-3673
	005	NISSAN	Japan	P.O. Box 685003 Franklin, TN 37068-5003	US	Franklin	37068-5003	800-647-7261	800-456-6622
*	006	TESLA	America	860 Washington st, NY, 10014	US	New York	10014	212-206-1204	800-456-6622
					HULL	HULL	HULL	HULL	HULL

## Model

```

CREATE TABLE MODEL (
    MODEL_ID CHAR(3) NOT NULL UNIQUE,
    MODEL_NAME VARCHAR(15) NOT NULL,
    MODEL_TYPE VARCHAR(15) NOT NULL,
    MODEL_YEAR CHAR(4),
    MODEL_MANUFACTURER VARCHAR(255),
    MODEL_PRICE VARCHAR(30) NOT NULL,
    BRAND_ID CHAR(3) NOT NULL,
    BRAND_NAME VARCHAR(15) NOT NULL,
    PRIMARY KEY (MODEL_ID, MODEL_NAME),
    FOREIGN KEY (BRAND_ID, BRAND_NAME) REFERENCES
    BRAND(BRAND_ID, BRAND_NAME)
);

INSERT INTO MODEL
(MODEL_ID, MODEL_NAME, MODEL_TYPE, MODEL_YEAR, MODEL_MANUFACTURER, MO
DEL_PRICE, BRAND_ID)

VALUES ('001', 'NISSAN LEAF', 'COMPACT', '2010', 'NISSAN',
'50000', '005');

INSERT INTO MODEL
(MODEL_ID, MODEL_NAME, MODEL_TYPE, MODEL_YEAR, MODEL_MANUFACTURER, MO
DEL_PRICE, BRAND_ID)

VALUES ('002', 'NISSAN e-NV200',
'COMPACT', '2013', 'NISSAN', '25000', '005');

INSERT INTO MODEL
(MODEL_ID, MODEL_NAME, MODEL_TYPE, MODEL_YEAR, MODEL_MANUFACTURER, MO
DEL_PRICE, BRAND_ID)

VALUES ('003', 'NISSAN Esflow', 'SPORTS
CAR', '2011', 'NISSAN', '19500', '005');

```

```

INSERT INTO MODEL
(MODEL_ID,MODEL_NAME,MODEL_TYPE,MODEL_YEAR,MODEL_MANUFACTURER,MO
DEL_PRICE,BRAND_ID)

VALUES ('004', 'NISSAN Nuvu', 'COMPACT','2008','NISSAN',
'23500','005');

INSERT INTO MODEL

(MODEL_ID,MODEL_NAME,MODEL_TYPE,MODEL_YEAR,MODEL_MANUFACTURER,MO
DEL_PRICE,BRAND_ID)

VALUES ('005', 'TESLA MODEL-3', 'COMPACT','2008','359000','
TESLA','006');

INSERT INTO MODEL

(MODEL_ID,MODEL_NAME,MODEL_TYPE,MODEL_YEAR,MODEL_MANUFACTURER,MO
DEL_PRICE,BRAND_ID)

VALUES ('006', 'TESLA MODEL-Y', 'MID-SIZE SUV','2008','299000','
TESLA','006');

INSERT INTO MODEL
(MODEL_ID,MODEL_NAME,MODEL_TYPE,MODEL_YEAR,MODEL_MANUFACTURER,MO
DEL_PRICE,BRAND_ID,BRAND_NAME)

VALUES ('007', 'TESLA MODEL-S', '5-door
liftback','2008','TESLA','399000','006','TESLA');

```

	MODEL_ID	MODEL_NAME	MODEL_TYPE	MODEL_YEAR	MODEL_MANUFACTURER	MODEL_PRICE	BRAND_ID	BRAND_NAME
▶	001	NISSAN LEAF	COMPACT	2010	NISSAN	50000	005	NISSAN
	002	NISSAN e-NV200	COMPACT	2013	NISSAN	25000	005	NISSAN
	003	NISSAN Esflow	SPORTS CAR	2011	NISSAN	19500	005	NISSAN
	004	NISSAN Nuvu	COMPACT	2008	NISSAN	23500	005	NISSAN
	005	TESLA MODEL-3	COMPACT	2008	TESLA	359000	006	TESLA
	006	TESLA MODEL-Y	MID-SIZE SUV	2008	TESLA	299000	006	TESLA
*	007	TESLA MODEL-S	5-door liftback	2008	TESLA	399000	006	TESLA
	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

## Body

```

CREATE TABLE BODY (
BODY_ID CHAR(3) NOT NULL UNIQUE,
BODY_NAME VARCHAR(15) NOT NULL,
BODY_MAKER VARCHAR(15) NOT NULL,

```

```

BODY_YEAR CHAR(4),
COLOR VARCHAR(255),
MODEL_ID CHAR(3) NOT NULL,
MODEL_NAME VARCHAR(15) NOT NULL,
PRIMARY KEY (MODEL_ID,BODY_ID,MODEL_NAME),
FOREIGN KEY (MODEL_ID,MODEL_NAME) REFERENCES
MODEL(MODEL_ID,MODEL_NAME)
);

INSERT INTO BODY
(BODY_ID,BODY_NAME,BODY MAKER,BODY_YEAR,COLOR,MODEL_ID,MODEL_NAME)

VALUES ('001', 'HATCHBACK','NISSAN','2010','BLACK',
'001','NISSAN LEAF');

INSERT INTO BODY
(BODY_ID,BODY_NAME,BODY MAKER,BODY_YEAR,COLOR,MODEL_ID,MODEL_NAME)

VALUES ('002', '4/5-door van','NISSAN','2009','BLACK',
'002','NISSAN e-NV200');

INSERT INTO BODY
(BODY_ID,BODY_NAME,BODY MAKER,BODY_YEAR,COLOR,MODEL_ID,MODEL_NAME)

VALUES ('003', 'coupe','NISSAN','2008','BLACK', '003','NISSAN Esflow');

INSERT INTO BODY
(BODY_ID,BODY_NAME,BODY MAKER,BODY_YEAR,COLOR,MODEL_ID,MODEL_NAME)

VALUES ('004', '2+1 seating','NISSAN','2011','BLACK',
'004','NISSAN Nuvu');

INSERT INTO BODY
(BODY_ID,BODY_NAME,BODY MAKER,BODY_YEAR,COLOR,MODEL_ID,MODEL_NAME)

VALUES ('005', '2+2 seating','TESLA','2011','BLACK',
'005','TESLA MODEL-3');

```

```
INSERT INTO BODY
(BODY_ID,BODY_NAME,BODY MAKER,BODY_YEAR,COLOR,MODEL_ID,MODEL_NAME)

VALUES ('006', '2+2 seating','TESLA','2011','BLACK',
'006','TESLA MODEL-Y');

INSERT INTO BODY
(BODY_ID,BODY_NAME,BODY MAKER,BODY_YEAR,COLOR,MODEL_ID,MODEL_NAME)

VALUES ('007', '2+2 seating','TESLA','2012','BLUE', '006','TESLA
MODEL-Y');

INSERT INTO BODY
(BODY_ID,BODY_NAME,BODY MAKER,BODY_YEAR,COLOR,MODEL_ID,MODEL_NAME)

VALUES ('008', '2+2 seating','TESLA','2013','WHITE',
'006','TESLA MODEL-Y');

INSERT INTO BODY
(BODY_ID,BODY_NAME,BODY MAKER,BODY_YEAR,COLOR,MODEL_ID,MODEL_NAME)

VALUES ('009', '5-door liftback','TESLA','2013','WHITE',
'007','TESLA MODEL-S');

INSERT INTO BODY
(BODY_ID,BODY_NAME,BODY MAKER,BODY_YEAR,COLOR,MODEL_ID,MODEL_NAME)

VALUES ('010', '5-door liftback','TESLA','2013','BLACK',
'007','TESLA MODEL-S');
```

	BODY_ID	BODY_NAME	BODY_MAKER	BODY_YEAR	COLOR	MODEL_ID	MODEL_NAME
▶	001	HATCHBACK	NISSAN	2010	BLACK	001	NISSAN LEAF
	002	4/5-door van	NISSAN	2009	BLACK	002	NISSAN e-NV200
	003	coupe	NISSAN	2008	BLACK	003	NISSAN Esflow
	004	2+1 seating	NISSAN	2011	BLACK	004	NISSAN Nuvu
	005	2+2 seating	TESLA	2011	BLACK	005	TESLA MODEL-3
	006	2+2 seating	TESLA	2011	BLACK	006	TESLA MODEL-Y
	007	2+2 seating	TESLA	2012	BLUE	006	TESLA MODEL-Y
	008	2+2 seating	TESLA	2013	WHITE	006	TESLA MODEL-Y
	009	5-door liftback	TESLA	2013	WHITE	007	TESLA MODEL-S
	010	5-door liftback	TESLA	2013	BLACK	007	TESLA MODEL-S
*	NULl	NULl	NULl	NULl	NULl	NULl	NULl

## Stereo System

```
CREATE TABLE STEREO_SYSTEM (
STEREO_SYSTEM_ID CHAR(3) NOT NULL UNIQUE,
STEREO_SYSTEM_NAME VARCHAR(255) NOT NULL,
STEREO_SYSTEM_NUMBER VARCHAR(30) NOT NULL,
STEREO_SYSTEM MAKER VARCHAR(15) NOT NULL,
STEREO_SYSTEM_YEAR CHAR(4),
MODEL_ID CHAR(3) NOT NULL,
PRIMARY KEY (MODEL_ID,STEREO_SYSTEM_ID),
FOREIGN KEY (MODEL_ID) REFERENCES MODEL(MODEL_ID)
);
```

```
INSERT INTO STEREO_SYSTEM

(STEREO_SYSTEM_ID,STEREO_SYSTEM_NAME,STEREO_SYSTEM_NUMBER,STEREO
_SYSTEM_MAKER,STEREO_SYSTEM_YEAR,MODEL_ID)

VALUES ('001', 'Sony CMT-SBT20 Micro Hi-Fi System with
Bluetooth','03942','sony','2010', '001');

INSERT INTO STEREO_SYSTEM

(STEREO_SYSTEM_ID,STEREO_SYSTEM_NAME,STEREO_SYSTEM_NUMBER,STEREO
_SYSTEM_MAKER,STEREO_SYSTEM_YEAR,MODEL_ID)

VALUES ('002', 'LG CK43 300 Watt Hi-Fi Shelf
System','033495','LG','2010','001');

INSERT INTO STEREO_SYSTEM
```

```

(STEREO_SYSTEM_ID,STEREO_SYSTEM_NAME,STEREO_SYSTEM_NUMBER,STEREO
_SYSTEM_MAKER,STEREO_SYSTEM_YEAR,MODEL_ID)

VALUES ('003', 'Sony MHC-EC619iP Home Stereo
System','06786','Sony','2010','001');

INSERT INTO STEREO_SYSTEM

(STEREO_SYSTEM_ID,STEREO_SYSTEM_NAME,STEREO_SYSTEM_NUMBER,STEREO
_SYSTEM_MAKER,STEREO_SYSTEM_YEAR,MODEL_ID)

VALUES ('004', 'Onkyo CS-265 CD Hi-Fi Mini
System','023457','Onkyo','2010','001');

```

	STEREO_SYSTEM_ID	STEREO_SYSTEM_NAME	STEREO_SYSTEM_NUMBER	STEREO_SYSTEM_MAKER	STEREO_SYSTEM_YEAR	MODEL_ID
▶	001	Sony CMT-SBT20 Micro Hi-Fi System with Bluetoo...	03942	sony	2010	001
	002	LG CK43 300 Watt Hi-Fi Shelf System	033495	LG	2010	001
	003	Sony MHC-EC619iP Home Stereo System	06786	Sony	2010	001
●	004	Onkyo CS-265 CD Hi-Fi Mini System	023457	Onkyo	2010	001
	HULL	HULL	HULL	HULL	HULL	HULL

## DVD player

```

CREATE TABLE DVDPLAYER (
D_ID CHAR(2) NOT NULL,
D_NAME VARCHAR(255) NOT NULL,
D_NUMBER VARCHAR(30) NOT NULL,
D_MAKER VARCHAR(15) NOT NULL,
D_YEAR CHAR(4) NOT NULL,
MODEL_ID CHAR(3) NOT NULL,
STEREO_SYSTEM_ID CHAR(3) NOT NULL,
PRIMARY KEY (D_ID, MODEL_ID, STEREO_SYSTEM_ID),
FOREIGN KEY (MODEL_ID) REFERENCES MODEL(MODEL_ID),
FOREIGN KEY (STEREO_SYSTEM_ID) REFERENCES,
STEREO_SYSTEM(STEREO_SYSTEM_ID)
);

INSERT INTO DVDPLAYER VALUES ('76',
'DP_12','12887','SONY','2008','004','001');

INSERT INTO DVDPLAYER VALUES ('43',
'DP_155','87665','PANASONIC','2010','001');

```

	D_ID	D_NAME	D_NUMBER	D MAKER	D_YEAR	MODEL_ID	STEREO_SYSTEM_ID
▶	43	DP_155	87665	PANASONIC	2010	001	002
*	76	DP_12	12887	SONY	2008	004	001
*			NULL	NULL	NULL	NULL	NULL

## Operating consumption

```

CREATE TABLE OPERATING_CONSUMPTION(
METER_ID CHAR(3) NOT NULL,
METER_NAME VARCHAR(255) NOT NULL,
METER_NUMBER VARCHAR(30) NOT NULL,
METER_MAKER VARCHAR(15) NOT NULL,
METER_YEAR CHAR(4) NOT NULL,
MODEL_ID CHAR(3) NOT NULL,
PRIMARY KEY(METER_ID),
INDEX (MODEL_ID),
FOREIGN KEY (MODEL_ID) REFERENCES MODEL(MODEL_ID)
);

INSERT INTO OPERATING_CONSUMPTION VALUES ('111',
'M1','09876','MEGAMETER','2008','004');

INSERT INTO OPERATING_CONSUMPTION VALUES ('078',
'M97','03452','MEGAMETER','2010','003');

INSERT INTO OPERATING_CONSUMPTION VALUES ('156',
'M11','08897','METEROMAKE','2010','001');

INSERT INTO OPERATING_CONSUMPTION VALUES ('098',
'M02','02567','METEROMAKE','2000','002');

```

	METER_ID	METER_NAME	METER_NUMBER	METER_MAKER	METER_YEAR	MODEL_ID
▶	156	M11	08897	METEROMAKE	2010	001
	098	M02	02567	METEROMAKE	2000	002
	078	M97	03452	MEGAMETER	2010	003
	111	M1	09876	MEGAMETER	2008	004
			NULL	NULL	NULL	NULL

## Engine

```

CREATE TABLE ENGINE (
ENGINE_ID CHAR(3) NOT NULL,
ENGINE_NAME VARCHAR(255) NOT NULL,
ENGINE_NUMBER VARCHAR(30) NOT NULL,
ENGINE MAKER VARCHAR(15) NOT NULL,
ENGINE_YEAR CHAR(4) NOT NULL,
MODEL_ID CHAR(3) NOT NULL,
METER_ID CHAR(3) NOT NULL,
PRIMARY KEY (ENGINE_ID),
INDEX (MODEL_ID),
INDEX (METER_ID),
FOREIGN KEY (MODEL_ID) REFERENCES MODEL(MODEL_ID),
FOREIGN KEY (METER_ID) REFERENCES OPERATING_CONSUMPTION(METER_ID)
);

```

```

INSERT INTO ENGINE VALUES ('176',
'MP98','2997','MEGAPOWER','2007','004' , '111');


```

```

INSERT INTO ENGINE VALUES ('188',
'POWER','6743','MEGAPOWER','2012','002' , '098');


```

```

INSERT INTO ENGINE VALUES ('158',
'ENERGY','9987','MEGAPOWER','2011','003' , '078');


```

```

INSERT INTO ENGINE VALUES ('879',
'MP98','2998','MEGAPOWER','2009', '001' , '156');


```

	ENGINE_ID	ENGINE_NAME	ENGINE_NUMBER	ENGINE_MAKER	ENGINE_YEAR	MODEL_ID	METER_ID
▶	158	ENERGY	9987	MEGAPOWER	2011	003	078
	176	MP98	2997	MEGAPOWER	2007	004	111
	188	POWER	6743	MEGAPOWER	2012	002	098
	879	MP98	2998	MEGAPOWER	2009	001	156
	NULL	NULL	NULL	NULL	NULL	NULL	NULL

## Battery

```

CREATE TABLE BATTERY (
BATTERY_ID CHAR(4) NOT NULL,
BATTERY_NAME VARCHAR(255) NOT NULL,
BATTERY_NUMBER VARCHAR(30) NOT NULL,
BATTERY_MAKER VARCHAR(15) NOT NULL,
BATTERY_YEAR CHAR(4) NOT NULL,
MODEL_ID CHAR(3) NOT NULL,

```

```

METER_ID CHAR(3) NOT NULL,
PRIMARY KEY(BATTERY_ID),
INDEX (MODEL_ID),
INDEX (METER_ID),
FOREIGN KEY (MODEL_ID) REFERENCES MODEL(MODEL_ID),
FOREIGN KEY (METER_ID) REFERENCES OPERATING_CONSUMPTION(METER_ID)
);

INSERT INTO BATTERY VALUES ('7854',
'BOOSTER125','B_125','POWER_ENERGY','2006','004' , '111');

INSERT INTO BATTERY VALUES ('9004',
'POWER22','P_22','MEGAPOWER','2011','003' , '078');

INSERT INTO BATTERY VALUES ('4544',
'BOOSTER120','B_120','POWER_ENERGY','2012','002' , '098');

INSERT INTO BATTERY VALUES ('7881',
'POWER120','P_120','POWER_ENERGY','2008', '001' , '156');

```

	BATTERY_ID	BATTERY_NAME	BATTERY_NUMBER	BATTERY MAKER	BATTERY_YEAR	MODEL_ID	METER_ID
▶	4544	BOOSTER120	B_120	POWER_ENERGY	2012	002	098
	7854	BOOSTER125	B_125	POWER_ENERGY	2006	004	111
	7881	POWER120	P_120	POWER_ENERGY	2008	001	156
	9004	POWER22	P_22	MEGAPOWER	2011	003	078
	NULL	NULL	NULL	NULL	NULL	NULL	NULL

## Window\_tint

```

CREATE TABLE WINDOW_TINT (
TINT_ID CHAR(3) NOT NULL,
COLOR VARCHAR(15) NOT NULL,
FILM_TYPE CHAR(3) NOT NULL,
PRICE CHAR(10) NOT NULL,
FILM_MAKER VARCHAR(15) NOT NULL,
PERCENT_VISIBLE CHAR(3) NOT NULL,
PERCENT_UV CHAR(3) NOT NULL,
MODEL_ID CHAR(3) NOT NULL,
BODY_ID CHAR(3) NOT NULL,
PRIMARY KEY (TINT_ID),
INDEX (MODEL_ID),
INDEX (BODY_ID),

```

```

FOREIGN KEY (MODEL_ID) REFERENCES MODEL(MODEL_ID),
FOREIGN KEY (BODY_ID) REFERENCES BODY(BODY_ID)
);

```

```

INSERT INTO WINDOW_TINT VALUES ('001',
'BLACK','20','59','CIR','24' , '99','001' , '001');
INSERT INTO WINDOW_TINT VALUES ('002',
'BLACK','10','79','CIR','16' , '99','002' , '002');
INSERT INTO WINDOW_TINT VALUES ('003',
'BLACK','5','99','CIR','8' , '99','002' , '002');
INSERT INTO WINDOW_TINT VALUES ('004', 'DARK
BLUE','20','99','CXP','26' , '95','003' , '003');
INSERT INTO WINDOW_TINT VALUES ('005', 'DARK
BLUE','10','129','CXP','19' , '95','004' , '004');
INSERT INTO WINDOW_TINT VALUES ('006', 'DARK
BLUE','5','149','CXP','10' , '95','004' , '004');

```

	TINT_ID	COLOR	FILM_TYPE	PRICE	FILM_MAKER	PERCENT_VISIBLE	PERCENT_UV	MODEL_ID	BODY_ID
▶	001	BLACK	20	59	CIR	24	99	001	001
	002	BLACK	10	79	CIR	16	99	002	002
	003	BLACK	5	99	CIR	8	99	002	002
	004	DARK BLUE	20	99	CXP	26	95	003	003
	005	DARK BLUE	10	129	CXP	19	95	004	004
*	006	DARK BLUE	5	149	CXP	10	95	004	004
	HULL	NULL	HULL	NULL	HULL	NULL	NULL	HULL	NULL

## Seat\_cover

```

CREATE TABLE SEAT_COVER (
SEATCOVER_ID CHAR(3) NOT NULL,
COLOR VARCHAR(15) NOT NULL,
FABRIC VARCHAR(15) NOT NULL,
PRICE CHAR(10) NOT NULL,
SEAT_MAKER VARCHAR(15) NOT NULL,
SIZE CHAR(10) NOT NULL,
PIECES CHAR(3) NOT NULL,
WEIGHT CHAR(10) NOT NULL,
MODEL_ID CHAR(3) NOT NULL,
BODY_ID CHAR(3) NOT NULL,
PRIMARY KEY (SEATCOVER_ID),
INDEX (MODEL_ID),
INDEX (BODY_ID),

```

```

FOREIGN KEY (MODEL_ID) REFERENCES MODEL(MODEL_ID),
FOREIGN KEY (BODY_ID) REFERENCES BODY(BODY_ID)
);

INSERT INTO SEAT_COVER VALUES ('001',
'BROWN','LEATHER','49','OASIS','30','4','9','001','001');
INSERT INTO SEAT_COVER VALUES ('002', 'BLACK','COTTON','29','FH
UNIVERSAL', '25','4','5','001','001');
INSERT INTO SEAT_COVER VALUES ('003',
'BLUE','NYLON','39','GORLA', '20','4','7','002','002');
INSERT INTO SEAT_COVER VALUES ('004', 'BLACK','LEATHER','49',
'OASIS', '30','4','9','003','003');
INSERT INTO SEAT_COVER VALUES ('005', 'BLUE','COTTON','29','FH
UNIVERSAL', '30','4','6','003','003');
INSERT INTO SEAT_COVER VALUES ('006',
'PINK','LEATHER','59',GORLA', '20','4','10','002','002');

```

	SEATCOVER_ID	COLOR	FABRIC	PRICE	SEAT_MAKER	SIZE	PIECES	WEIGHT	MODEL_ID	BODY_ID
▶	001	BROWN	LEATHER	49	OASIS	30	4	9	001	001
	002	BLACK	COTTON	29	FH UNIVERSAL	25	4	5	004	004
	003	BLACK	NYLON	39	GORLA	20	4	7	004	004
	004	BLACK	LEATHER	49	OASIS	30	4	9	003	003
	005	BLUE	COTTON	29	FH UNIVERSAL	30	4	6	003	003
	006	PINK	LEATHER	59	GORLA	20	4	10	002	002
*	HULL	HULL	HULL	HULL	HULL	HULL	HULL	HULL	HULL	HULL

## Camera

```

CREATE TABLE CAMERA (
CAMERA_ID CHAR(3) NOT NULL,
CAMERA_MAKER VARCHAR(15) NOT NULL,
CAMERA_RESOLUTION CHAR(15) NOT NULL,
CAMERA_YEAR CHAR(10) NOT NULL,
WEIGHT CHAR(10) NOT NULL,
WARRANTY_TIME CHAR(10) NOT NULL,
PRICE CHAR(10) NOT NULL,
MODEL_ID CHAR(3) NOT NULL,
BODY_ID CHAR(3) NOT NULL,
PRIMARY KEY (CAMERA_ID),
INDEX (MODEL_ID),

```

```

INDEX (BODY_ID),
FOREIGN KEY (MODEL_ID) REFERENCES MODEL(MODEL_ID),
FOREIGN KEY (BODY_ID) REFERENCES OPERATING_CONSUMPTION(BODY_ID)
);

```

```

INSERT INTO CAMERA VALUES ('001', 'SONY', '1980 * 2240', '2020', '7', '2', '50', '001', '001');
INSERT INTO CAMERA VALUES ('002', 'SAMSUNG', '1980 * 2240', '2021', '7', '2', '74', '001', '001');
INSERT INTO CAMERA VALUES ('003', 'SONY', '1440 * 1840', '2017', '13', '1', '29', '004', '004');
INSERT INTO CAMERA VALUES ('004', 'APPLE', '1880 * 2040', '2021', '4', '3', '99', '004', '004');
INSERT INTO CAMERA VALUES ('005', 'ASUS', '1660 * 1980', '2018', '8', '2', '39', '003', '003');
INSERT INTO CAMERA VALUES ('006', 'SAMSUNG', '2040 * 2440', '2021', '5', '1', '129', '002', '002');

```

	CAMERA_ID	CAMERA_MAKER	CAMERA_RESOLUTION	CAMERA_YEAR	WEIGHT	WARRANTY_TIME	PRICE	MODEL_ID	BODY_ID
▶	001	SONY	1980 * 2240	2020	7	2	50	001	001
	002	SAMSUNG	1980 * 2240	2021	7	2	74	001	001
	003	SONY	1440 * 1840	2017	13	1	29	004	004
	004	APPLE	1880 * 2040	2021	4	3	99	004	004
	005	ASUS	1660 * 1980	2018	8	2	39	003	003
	006	SAMSUNG	2040 * 2440	2021	5	1	129	002	002
*	HULL	HULL	HULL	HULL	HULL	HULL	HULL	HULL	HULL

## Tire\_type

```

CREATE TABLE TIRE_TYPE (
TIRE_ID CHAR(3) NOT NULL,
SIZE CHAR(10) NOT NULL,
MAKER VARCHAR(15) NOT NULL,
WIDTH CHAR(10) NOT NULL,
SEASON VARCHAR(15) NOT NULL,
AVG_DURATION CHAR(10) NOT NULL,
TIRE_COUNT CHAR(3) NOT NULL,
WEIGHT CHAR(10) NOT NULL,
PRICE CHAR(10) NOT NULL,
MODEL_ID CHAR(3) NOT NULL,

```

```

BODY_ID CHAR(3) NOT NULL,
PRIMARY KEY (TIRE_ID),
INDEX (MODEL_ID),
INDEX (BODY_ID),
FOREIGN KEY (MODEL_ID) REFERENCES MODEL(MODEL_ID),
FOREIGN KEY (BODY_ID) REFERENCES BODY(BODY_ID)
);

INSERT INTO TIRE_TYPE VALUES ('001',
'225','MICHELIN','49','WINTER','11' , '4' , '60' , '225' , '002'
,'002');
INSERT INTO TIRE_TYPE VALUES ('002',
'220','CHEVY','39','ALL-SEASON','14' , '4' , '55' , '250' , '002'
,'002');
INSERT INTO TIRE_TYPE VALUES ('003',
'225','ASIS','39','SUMMER','8' , '4' , '50' , '200' , '003' , '003');
INSERT INTO TIRE_TYPE VALUES ('004',
'215','GOODYEAR','49','ALL-SEASON','14' , '4' , '60' , '300' , '003'
,'003');
INSERT INTO TIRE_TYPE VALUES ('005',
'215','MICHELIN','39','SUMMER','7' , '4' , '48' , '275' , '004' , '004');
INSERT INTO TIRE_TYPE VALUES ('006',
'230','GOODYEAR','39','SUMMER','10' , '4' , '44' , '250' , '001' ,
'001');

```

	TIRE_ID	SIZE	MAKER	WIDTH	SEASON	AVG_DURATION	TIRE_COUNT	WEIGHT	PRICE	MODEL_ID	BODY_ID
▶	001	225	MICHELIN	49	WINTER	11	4	60	225	002	002
	002	220	CHEVY	39	ALL-SEASON	14	4	55	250	002	002
	003	225	ASIS	39	SUMMER	8	4	50	200	003	003
	004	215	GOODYEAR	49	ALL-SEASON	14	4	60	300	003	003
	005	215	MICHELIN	39	SUMMER	7	4	48	275	004	004
	006	230	GOODYEAR	39	SUMMER	10	4	44	250	001	001
*	HULL	HULL	NULL	NULL	HULL	NULL	NULL	NULL	NULL	NULL	NULL

## SQL Queries to retrieve data

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

```

SELECT MODEL_NAME,BRAND_NAME FROM MODEL
WHERE BRAND_NAME = 'TESLA'

```

	MODEL_NAME	BRAND_NAME
▶	TESLA MODEL-3	TESLA
	TESLA MODEL-Y	TESLA
	TESLA MODEL-S	TESLA

Here, we are using BRAND NAME as TESLA for the table called MODEL to find out all the Model Names we have in our database. Our Model table has the BRAND NAME entered as the foreign key.

- When the user clicks on a certain model (Let's say model : TESLA MODEL-S), then the specific information including the price of that model is displayed

```
SELECT * FROM MODEL
```

```
WHERE MODEL_NAME = 'TESLA MODEL-S'
```

	MODEL_ID	MODEL_NAME	MODEL_TYPE	MODEL_YEAR	MODEL_MANUFACTURER	MODEL_PRICE	BRAND_ID	BRAND_NAME
▶	007	TESLA MODEL-S	5-door liftback	2008	TESLA	399000	006	TESLA
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Here, we are extracting data from the MODEL table for selected model type TESLA MODEL-S.

- When the user clicks on the Exterior of a certain model (Let's say model : TESLA MODEL-S) then all the color options are displayed

```
SELECT MODEL_NAME,COLOR FROM BODY
```

```
WHERE MODEL_NAME = 'TESLA MODEL-S'
```

	MODEL_NAME	COLOR
▶	TESLA MODEL-S	WHITE
	TESLA MODEL-S	BLACK

Here, we are using MODEL\_NAME as TESLA MODEL-S for the table called BODY to find out all the colors available for this model in our database. Our BODY table has the MODEL\_NAME entered as the foreign key.

#### 4. Think of a scenario

- a. The user wants to find a car that has the engine of his likeability. He is looking for an engine made in 2012 with the name of it being Power.

```
SELECT E.ENGINE_NAME,E.ENGINE_YEAR,M.MODEL_NAME
FROM ENGINE E, MODEL M
WHERE E.ENGINE_NAME = 'POWER' AND E.ENGINE_YEAR = '2012' AND
E.MODEL_ID = M.MODEL_ID
```

	ENGINE_NAME	ENGINE_YEAR	MODEL_NAME
▶	POWER	2012	NISSAN e-NV200

Here, we are extracting data from the two different tables to fulfill user's requirements. We are looking for the data two specific requirements year and name from the ENGINE table and getting the associated model name available in our database using comparing the MODEL IDs from ENGINE and MODEL table. And as an output printing the data from the MODEL table.

#### 5. Think of a scenario

- a. When the user clicks on the tire type of his likeability. He is looking for a tire named 'ALL-SEASON' and the price is less than or equals to 300 dollars.

```
SELECT T.MAKER,T.SEASON,T.WEIGHT,M.MODEL_NAME,B.BODY_NAME
FROM TIRE_TYPE T, MODEL M ,BODY B
WHERE T.SEASON = 'ALL-SEASON' AND T.PRICE <= 300 AND T.MODEL_ID =
M.MODEL_ID AND T.MODEL_ID = B.MODEL_ID
```

	MAKER	SEASON	WEIGHT	PRICE	MODEL_NAME	BODY_NAME
▶	CHEVY	ALL-SEASON	55	250	NISSAN e-NV200	4/5-door van
	GOODYEAR	ALL-SEASON	60	300	NISSAN Esflow	coupe

Here, we are extracting data from the three different tables to fulfill user's requirements. We are looking for the data on two specific requirements Tire Season and Price from the TIRE table and getting the associated model name available in our database using comparing the MODEL IDs from TIRE and MODEL table to get an output and print the data from the MODEL table. Also, comparing the MODEL IDs from TIRE and BODY table to get an output and print the BODY\_NAME from the BODY table.

### Database Application Screenshots:

#### MyEvcars

The Best EV cars Website is here for you :).

HERE YOU WILL SEE HOW RETRIEVE EV CARS DATA FROM DATABASE

BRAND Name	BRAND COUNTRY
<a href="#">Toyota</a>	Japan
<a href="#">Mercedes-Benz</a>	Germany
<a href="#">BMW</a>	Germany
<a href="#">FORD</a>	America
<a href="#">NISSAN</a>	Japan
<a href="#">TESLA</a>	America

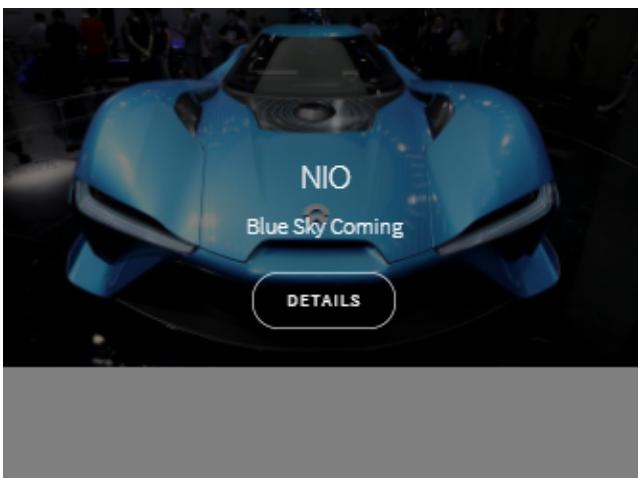
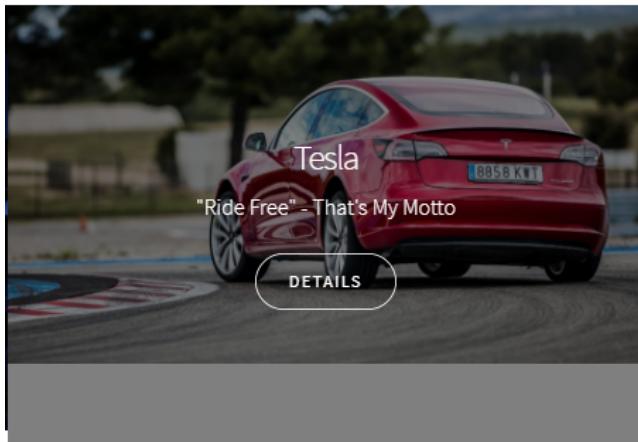
[EXPLORE MORE](#)



#### Gallery

All Amazing Cars Gallery





## Company Information

What we will GIVE U THE BEST SERVICE – THE REAL INFORMATION – THE GOOD PRICE

[LEARN MORE](#)



## Model Section

Model Name	MODEL_MANUFACTURER
<a href="#">NISSAN LEAF</a>	NISSAN
<a href="#">NISSAN e-NV200</a>	NISSAN
<a href="#">NISSAN Esflow</a>	NISSAN
<a href="#">NISSAN Nuvu</a>	NISSAN
<a href="#">TESLA MODEL-3</a>	TESLA
<a href="#">TESLA MODEL-Y</a>	TESLA
<a href="#">TESLA MODEL-S</a>	TESLA

[LEARN MORE](#)

## Body Section

Body Name	Body Maker	Body Year	Color	Model Name
HATCHBACK	NISSAN	2010	BLACK	NISSAN LEAF
4/5-door van	NISSAN	2009	BLACK	NISSAN e-NV200
COUPE	NISSAN	2008	BLACK	NISSAN Esflow
2+1 seating	NISSAN	2011	BLACK	NISSAN Nuvu
2+2 seating	TESLA	2011	BLACK	TESLA MODEL-3
2+2 seating	TESLA	2011	BLACK	TESLA MODEL-Y
2+2 seating	TESLA	2012	BLUE	TESLA MODEL-Y
2+2 seating	TESLA	2013	WHITE	TESLA MODEL-Y
5-door liftback	TESLA	2013	WHITE	TESLA MODEL-S
5-door liftback	TESLA	2013	BLACK	TESLA MODEL-S

[LEARN MORE](#)



**BODY\_NAME BODY MAKER COLOR MODEL\_NAME**  
HATCHBACK NISSAN BLACK NISSAN LEAF



### Battery Section

Battery Name	Battery Maker
BOOSTER120	POWER_ENERGY
BOOSTER125	POWER_ENERGY
POWER120	POWER_ENERGY
POWER22	MEGAPOWER

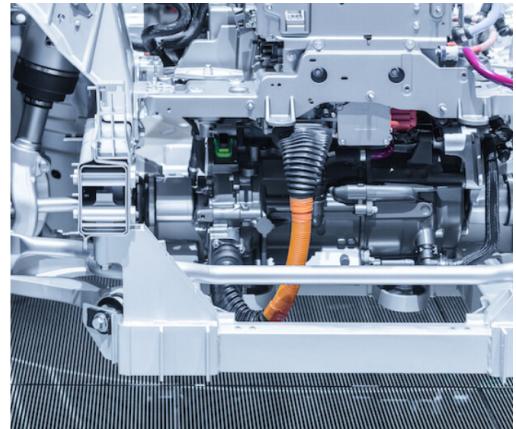
[LEARN MORE](#)

**BATTERY\_NAME BATTERY MAKER BATTERY YEAR**  
BOOSTER120 POWER\_ENERGY 2012

### Engine Section

Engine Name	Engine Maker
ENERGY	MEGAPOWER
MPS8	MEGAPOWER
POWER	MEGAPOWER
MPS8	MEGAPOWER

[LEARN MORE](#)



### Others you want to learn about

Oprating Systems

Window Tint

Camera

Tire Type

Seat Cover

More



With the help of AWS,XAMPP and MySQL, The team exported all the data into the webpage. All the important tables are included in the database application. An application server works with databases. The dynamic page instructs the application server to extract data from a database and insert it into the PHP page. The instruction to extract data from a database is called a database query. A query consists of search criteria expressed in a database language called SQL (Structured Query Language). The SQL query is written into the page's server-side scripts or tags. The application server can communicate only through the intermediary of a database driver. A database driver is software that acts as an interpreter between the application server and the database. After the driver establishes communication, the query is executed against the database, and a recordset is created. The record set is returned to the application server and the data used in the dynamic page. Here is the introduction to all the pages.

- We have Company's name on the top and Company basic information in the middle of the page for increasing brand awareness and building strong customer trust.
- On the top webpage, we have five sections, including brand, model, body, engine, and battery. Informative pictures are employed for guiding people to find the part conveniently and effectively.
- By employing a data-driven online platform for operating technology support and effective database management to optimize the online services, we put the content that customers care most about in priority and consumers can filter various characteristics by clicking different attributes in each section.
- The gallery part displays amazing designed cars, and website visitors can learn more information by clicking the 'Details' button.
- At the bottom, we also have other filter options for the customer to choose from, satisfying various customer needs. We want to be one of the best websites that pushes usability, accessibility, and responsive design forward.
- The team listed all the social media we have for increasing subscriber, better knowing user experience, and expanding global and domestic customer bases.