# Exercises: ASP.NET Essentials – Part I

Problems for exercises and homework for the [“C# MVC Frameworks - ASP.NET” course @ SoftUni](https://softuni.bg/courses/asp-net-mvc).

## Car Dealer

You are provided with a ASP.NET Core MVC project with database model as follows.

A car dealer needs information about cars, their parts, parts suppliers, customers and sales.

* **Cars** have **make, model**, travelled distance in kilometers
* **Parts** have **name**, **price** and **quantity**
* Part **supplier** have **name** and info whether he **uses imported parts**
* **Customer** has **name**, **date of birth** and info whether he **is young driver** (Young driver is a driver that has **less than 2 years of experience**. Those customers get **additional 5% off** for the sale.)
* **Sale** has **car**, **customer** and **discount percentage**

A **price of a car** is formed by **total price of its parts**.



Relations between models:

* A **car** has **many parts** and **one part** can be placed **in many cars**
* **One supplier** can supply **many parts** and each **part** can be delivered by **only one supplier**
* In **one sale**, only **one car** can be sold
* **Each sale** has **one customer** and **a customer** can buy **many cars**

## Car Dealer Import Data

Use the provided SQL script to **populate the database** with sample data.

## Queries

Using the provided project skeleton create an application that can show to the user results from different queries described below. The results from the queries should be visualizes **in a table.**

#### Query 1 – Ordered Customers

Get all **customers** ordered by their **birth date ascending or descending**. If two customers are born on the same date **first print those who are not young drivers** (e.g. print experienced drivers first). **Show** their name, birthdate and whether he is young driver or not.

##### Example URLs

{host}/customers/all/ascending

{host}/customers/all/descending

#### Query 2 – Cars from make

Get all **cars** from make **provided in the URL** and **order them by model alphabetically** and by **travelled distance descending**. **Show** them in a table with their **make**, **model** and **travelled distance**.

##### Example URL

{host}/cars/{make}

#### Query 3 – Filter Suppliers

Get a list of all suppliers. There are two types of suppliers

* **Local** – do not import parts from abroad
* **Importers** – import parts from abroad

Get their **id**, **name** and the **number of parts they can offer to supply**.

##### Example URLs

{host}/suppliers/local

{host}/suppliers/importers

#### Query 4 – Cars with Their List of Parts {by id}

Get all **cars along with their list of parts**. For the **car** get only **make, model** and **travelled distance** and for the **parts** get only **name** and **price**.

##### Example URL

{host}/cars/parts

#### Query 5 – Total Sales by Customer

Get a **customer by ID** and show his **name**, **bought cars** **count** and **total spent money** on cars.

##### Example URL

{host}/customers/{id}

#### Query 6 – Sales with Applied Discount

Get all **sales** with information about the **car (make, model, travelled distance)**, **customer** and **price** of the sale **with and without discount** and the **discount percent** itself.

##### Example URLs

{host}/Sales – list of all sales

{host}/Sales/{id} – more details about sale by provided Id (car make, car model, and customer name)

{host}/Sales/discounted – all sales that are discounted by any percentage

{host}/Sales/discounted/{percent} – all sales that are discounted with given percent

## Hyperlinks

Add menu to the website with the following structure

* **Customers** - customers/all/ascending
* **Sales**
  + **All** – /sales
  + **Discounted** - sales/discounted
* **Cars** - cars/all list of all cars with their make, model and travelled distance