 Софийски Университет

Св. Климент Охридски

Факултет по Математика и Информатика

**Документация**

към проект по „*ASP Програмиране*“ на тема: „*Каталог за компютърен магазин*“

Изготвил:

Димитър Михайлов – 71773 Дата: 08.07.2020г.

Съдържание

1. Описание на курсовата работа.

2. Описание на елементите и атрибутите на XSD

3. XSD

4. Описание на базата от данни

5. Схема на взаимовръзката между таблиците на базата от данни

6. Функционалности работещи с XML файлове и базата

1. Описание на курсовата работа.

Проектът е на тема „Автокъщи“. Сайтът представя информация за различни автокъщи, и колите в тях. XML файловете описват характеристиките на колите в автокъщите.

Проектът е написан на най-новата технология на ASP – Blazor.

2. Описание на елементите и атрибутите на XSD

* *dealership* – основният елемент в XML документите, който трябва да съдържа елементите cars
  + *id –* индентификационен номер
  + *address* - адрес
* *cars* – елемент, обединяващ цялата информация за колите, предлагани в автокъщата
* *car* – елемент, в който се съдържа подробна информация за всяка кола, всички вложени елементи са задължителни
* *manufactureDetails* – елемент, в който се съдържа информация за създателя на колата
  + *manufacturer* – прост елемент, символен низ, енумерация марка(Хонда, Бмв …)
  + *model* – елемент, символен низ - модел на колата,
  + *year* – елемент, реално число - година на регистрация,
* *engineDetails* – комплексен елемент, съдържащ данни за двигателя на колата
  + *engineType* – прост елемент, символен низ - тип двигател (Бензин,Дизел,Електричен)
  + *engineVolume* – елемент, дробно число - обем на двигателя
  + *horsePower* – елемент, реално число - конски сили
* *visualDetails* – комплексен елемент, съдържащ информация относно визуалните характеристики на колата:
  + *color* – елемент, символен низ - цвят
  + *bodyType* – прост елемент, символен низ енумерация (Седан,Купе,Комби)
  + *doorCount* – елемент, цяло число - брой врати
  + *hasWing* – елемент, булева стойност, 1 – има спойлер, 0 – няма спойлер
* *wheels* – комплексен елемент, съдържащ информация за гумите и джантите на колата
  + *rims* – комплексен елемент, съдържащ информация за джантите на колата
    - color – цвят, символен низ
    - brand – прост елемент, символен низ, енумерация (ОЗ,ББС) марка джанти
    - model – символен низ, модел на марката джанти
    - size – цяло число, цола
  + *tires* – комплексен елемент, съдържащ информация за гумите на колата
    - brand – прост елемент, символен низ, енумерация (Мишелин, Гуудийр)
    - season – прост елемент, символен низ, енумерация (Летни, Зимни, Всесезонни)
    - width – елемент, цяло число, ширина на гумите
    - diameter – елемент, цяло число, диаметър на гумите
* *transmission* – комплексен елемент, съдържащ информация за скоростната кутия
  + *type* – прост тип, символен низ, енумерация (Ръчни, Автоматични)
  + *gearCount* – елемент, цяло число, брой предавки
* *price* – елемент, съдържащ символен низ, кратко описание на играта

3. XSD

<?xml version="1.0" encoding="UTF-8" ?>

<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"

    targetNamespace="https://www.w3schools.com"

    xmlns="https://www.w3schools.com"

    elementFormDefault="qualified">

  <xs:simpleType name="manufacturerType">

    <xs:restriction base="xs:string">

      <xs:enumeration value="Ауди"/>

      <xs:enumeration value="Голф"/>

      <xs:enumeration value="БМВ"/>

      <xs:enumeration value="Хонда"/>

      <xs:enumeration value="Хюндай"/>

      <xs:enumeration value="Мазда"/>

      <xs:enumeration value="Субару"/>

      <xs:enumeration value="Нисан"/>

      <xs:enumeration value="Тойота"/>

      <xs:enumeration value="Тесла"/>

    </xs:restriction>

  </xs:simpleType>

  <xs:simpleType name="engineType">

    <xs:restriction base="xs:string">

      <xs:enumeration value="Бензин"/>

      <xs:enumeration value="Дизел"/>

      <xs:enumeration value="Електричен"/>

    </xs:restriction>

  </xs:simpleType>

  <xs:simpleType name="bodyType">

    <xs:restriction base="xs:string">

      <xs:enumeration value="Седан"/>

      <xs:enumeration value="Купе"/>

      <xs:enumeration value="Комби"/>

    </xs:restriction>

  </xs:simpleType>

  <xs:simpleType name="rimBrand">

    <xs:restriction base="xs:string">

      <xs:enumeration value="ОЗ"/>

      <xs:enumeration value="ББС"/>

    </xs:restriction>

  </xs:simpleType>

  <xs:simpleType name="tireBrand">

    <xs:restriction base="xs:string">

      <xs:enumeration value="Мишелин"/>

      <xs:enumeration value="Гуудийр"/>

    </xs:restriction>

  </xs:simpleType>

  <xs:simpleType name="tireSeason">

    <xs:restriction base="xs:string">

      <xs:enumeration value="Летни"/>

      <xs:enumeration value="Зимни"/>

      <xs:enumeration value="Всесезонни"/>

    </xs:restriction>

  </xs:simpleType>

  <xs:simpleType name="transmissionType">

    <xs:restriction base="xs:string">

      <xs:enumeration value="Ръчни"/>

      <xs:enumeration value="Автоматични"/>

    </xs:restriction>

  </xs:simpleType>

  <xs:complexType name="engineDetails">

    <xs:all>

      <xs:element name="engineType" type="engineType"/>

      <xs:element name="engineVolume" type="xs:double"/>

      <xs:element name="horsePower" type="xs:int"/>

    </xs:all>

  </xs:complexType>

  <xs:complexType name="manufactureDetails">

    <xs:sequence>

      <xs:element name="manufacturer" type="manufacturerType"/>

      <xs:element name="model" type="xs:string"/>

      <xs:element name="year" type="xs:int"/>

    </xs:sequence>

  </xs:complexType>

  <xs:complexType name="visualDetails">

    <xs:all>

      <xs:element name="color" type="xs:string"/>

      <xs:element name="bodyType" type="bodyType"/>

      <xs:element name="doorCount" type="xs:int"/>

      <xs:element name="hasWing" type="xs:boolean"/>

    </xs:all>

  </xs:complexType>

  <xs:complexType name="wheels">

    <xs:all>

      <xs:element name="rims">

        <xs:complexType>

          <xs:all>

            <xs:element name="color" type="xs:string"/>

            <xs:element name="brand" type="rimBrand"/>

            <xs:element name="model" type="xs:string"/>

            <xs:element name="size" type="xs:int"/>

          </xs:all>

        </xs:complexType>

      </xs:element>

      <xs:element name="tires">

        <xs:complexType>

          <xs:all>

            <xs:element name="brand" type="tireBrand"/>

            <xs:element name="season" type="tireSeason"/>

            <xs:element name="width" type="xs:int"/>

            <xs:element name="diameter" type="xs:int"/>

          </xs:all>

        </xs:complexType>

      </xs:element>

    </xs:all>

  </xs:complexType>

  <xs:complexType name="transmission">

    <xs:all>

      <xs:element name="type" type="transmissionType"/>

      <xs:element name="gearCount" type="xs:int"/>

    </xs:all>

  </xs:complexType>

  <xs:element name="dealership">

    <xs:complexType>

      <xs:sequence>

        <xs:element name="cars">

          <xs:complexType>

            <xs:sequence>

              <xs:element maxOccurs="unbounded" name="car">

                <xs:complexType>

                  <xs:sequence>

                    <xs:element name="manufactureDetails" type="manufactureDetails"/>

                    <xs:element name="engineDetails" type="engineDetails"/>

                    <xs:element name="visualDetails" type="visualDetails"/>

                    <xs:element name="wheels" type="wheels"/>

                    <xs:element name="transmission" type="transmission"/>

                    <xs:element name="price" type="xs:int"/>

                  </xs:sequence>

                  <xs:attribute name="id" type="xs:int"/>

                </xs:complexType>

              </xs:element>

            </xs:sequence>

          </xs:complexType>

        </xs:element>

      </xs:sequence>

      <xs:attribute name="id" type="xs:int"/>

      <xs:attribute name="address" type="xs:string"/>

    </xs:complexType>

  </xs:element>

</xs:schema>

4. Описание на базата от данни



**Dealerships**

* **Id –** първичен ключ тип **int**
* **Address –** aдрес на автокъщата **string**

**Cars**

* **Id -** първичен ключт ип **int**
* **Price –** цена тип **int**
* **DealershipId –** външен ключ към идентификатор на **Dealership**

**EngineDetails**

* **Id -** първичен ключ тип **int,** външен ключ към идентификатор на **Car**
* **EngineType –** тип двигател **string**
* **EngineVolume –** обем на двигател **decimal**
* **HorsePower –** конски сили **int**

**ManufactureDetails**

* **Id -** първичен ключ тип **int,** външен ключ към идентификатор на **Car**
* **Manufacturer –** Марка на колата **string**
* **Model –** Модел на колата **string**
* **Year –** Година на колата **string**

**Rims**

* **Id -** първичен ключ тип **int,** външен ключ към идентификатор на **Car**
* **Color –** цвят **string**
* **RimBrand –** марка **string**
* **Model –** модел **string**
* **Size –** размер **int**

**Tires**

* **Id -** първичен ключ тип **int,** външен ключ към идентификатор на **Car**
* **TireBrand –** марка **string**
* **TireSeason –** сезон **string**
* **Width –** ширина **int**
* **Diameter –** диаметър/ Цол **int**

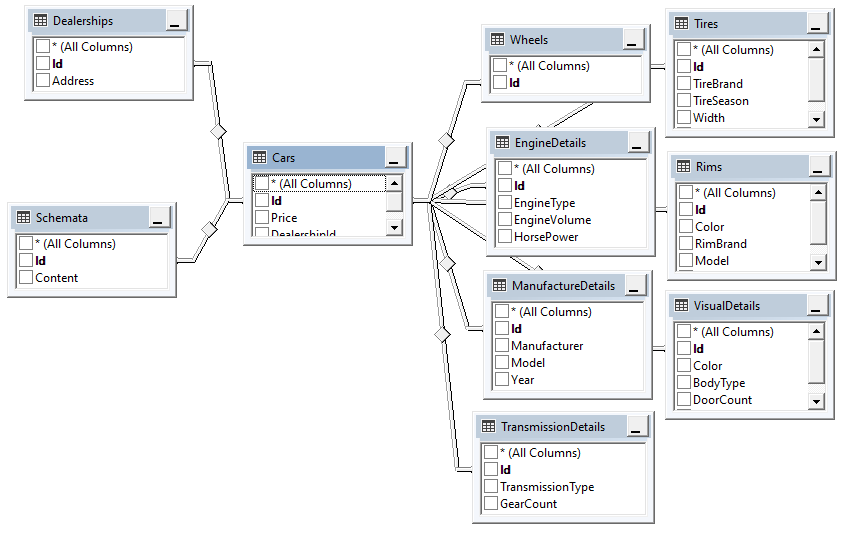
**TransmissionDetails**

* **Id -** първичен ключ тип **int,** външен ключ към идентификатор на **Car**
* **TransmissionType –** Тип скорости **string**
* **GearCount –** брой скорости **int**

**Schemata** (Schemas, но EntityFramework го преименува на Schemata)

* **Id -** първичен ключ тип **int**
* **Content – byte[]**

5. Схема на взаимодействие на базата данни

****

6.Функционалности работещи с XML файлове и базата

**Главни функции:  
CreateEdit.razor: -** Страница, на която може да се попълват или редактират данни на автокъща.

* **Upload() –** Проверява създадения/редактирания модел дали е валиден според качената в базата XSD схема. Ако е валиден се записва в базата.

**Dealerships.razor -**  Страница, на която има списък с автокъщи.

Посредством иконки всяка може да бъде редактирана, изтрита, изтеглена като xml

* **void Download(int id) –** изтегля избраната автокъща като xml.

**Upload.razor –** Страница, на която може да се импортират xml и xsd.

* **async Task UploadSchema(IFileListEntry[] files)**

Метод за импортиране на xsd схема.

* **async Task UploadFile**(**IFileListEntry[] files)**

Метод за импортиране на ВАЛИДЕН XML файл. Ако файлът е невалиден се изписва съобщение на страницата.

* **async Task UploadFile**(**IFileListEntry[] files)**

Метод за импортиране на всички прикачени XML файлове. Прикачва тези, които са валидни, а за другите изписва съобщение с грешките.