

Platformy Technologiczne .NET C#

Poziom inżynierski, studia niestacjonarne

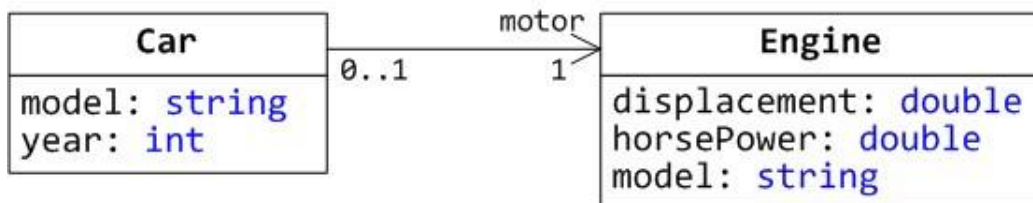
Laboratorium 3. LINQ to XML

Dokumentacja

- [LINQ to XML](#)
- [Controlling XML Serialization Using Attributes](#)
- [XmlRootAttribute](#)
- [Specify an Alternate Element Name for an XML Stream](#)
- [XPath Reference](#)

Zadania wstępne

- Utworzyć klasy implementujące model danych z poniższego diagramu.



- Utworzyć listę obiektów Car:

```
List<Car> myCars = new List<Car>(){
    new Car("E250", new Engine(1.8, 204, "CGI", 2009),
    new Car("E350", new Engine(3.5, 292, "CGI", 2009),
    new Car("A6", new Engine(2.5, 187, "FSI", 2012),
    new Car("A6", new Engine(2.8, 220, "FSI", 2012),
    new Car("A6", new Engine(3.0, 295, "TFSI", 2012),
    new Car("A6", new Engine(2.0, 175, "TDI", 2011),
    new Car("A6", new Engine(3.0, 309, "TDI", 2011),
    new Car("S6", new Engine(4.0, 414, "TFSI", 2012),
    new Car("S8", new Engine(4.0, 513, "TFSI", 2012)
};
```

Zadania główne

1. Napisać 2 zapytania LINQ:

- a) pierwsze (0,5 pkt) dokonuje projekcji elementów kolekcji myCars, dla których model to A6, na typ anonimowy o dwóch własnościach: *engineType* i *hppl*;
 - *engineType* ma wartość "diesel" dla silników "TDI" a "petrol" dla pozostałych;
 - *hppl* = *horsePower* / *displacement*
- b) drugie (0,5 pkt), na podstawie rezultatu zapytania pierwszego, grupuje wartości *hppl* po typie silnika (*engineType*); używając pętli foreach wyświetlić utworzone grupy wraz z przeciętną wartością *hppl*

```
petrol: 83,9015873015873
diesel: 95,25
```

2. Zaimplementować serializację i deserializację kolekcji myCars do formatu XML przedstawionego w załączniku nr 1 (0,6). W trakcie serializacji:
 - a) zmienić nazwę elementu będącego korzeniem dokumentu na "cars" (0,1 pkt)
 - b) zmienić nazwę elementu "motor" na "engine" (0,1 pkt)
 - c) zmienić nazwę elementu "Car" na "car" (0,1 pkt)
 - d) element "model" opisujący silnik ma stać się atrybutem elementu "engine" (0,1 pkt)
3. Napisać i zaprezentować wyrażenie XPath, które na dokumencie z zadania poprzedniego:
 - a) obliczy przeciętną moc samochodów o silnikach innych niż TDI (0,5 pkt)


```

          XElement rootNode = XElement.Load("CarsCollection.xml");
          double avgHP = (double) rootNode.XPathEvaluate("myXPathExpression1");
          
```
 - b) zwróci modele samochodów bez powtórzeń (0,5 pkt)


```

          IEnumerable<XElement> models =
          rootNode.XPathSelectElements("myXPathExpression2");
          
```
4. Napisać zapytanie LINQ do kolekcji myCars, tak aby plik XML wygenerowany przez poniższą metodę miał taką samą strukturę jak plik w załączniku nr 1 (1 pkt)


```

      private void createXmlFromLinq(List<Car> myCars) {
        IEnumerable<XElement> nodes = ...//LINQ query expressions
        XElement rootNode = new XElement("cars", nodes); //create a root node to contain
        the query results  rootNode.Save("CarsFromLinq.xml");
      }
      
```
5. Korzystając z LINQ to XML wygenerować na podstawie kolekcji myCars dokument XHTML zawierający tabelę, której wiersze reprezentują kolejne elementy kolekcji (1 pkt). Można ułatwić sobie zadanie poprzez załadowanie pustego dokumentu XHTML (patrz załącznik nr 2) i dołączenie wygenerowanego elementu table.

E250	CGI	1.8	204	2009
E350	CGI	3.5	292	2009
A6	FSI	2.5	187	2012
A6	FSI	2.8	220	2012
A6	TFSI	3	295	2012
A6	TDI	2	175	2011
A6	TDI	3	309	2011
S6	TFSI	4	414	2012
S8	TFSI	4	513	2012

Załącznik 1

```
<?xml version="1.0" encoding="utf-8" ?>
<cars>
  <car>
    <model>E250</model>
    <engine model="CGI">
      <displacement>1.8</displacement>
      <horsePower>204</horsePower>
    </engine>
    <year>2009</year>
  </car>
  <car>
    <model>E350</model>
    <engine model="CGI">
      <displacement>3.5</displacement>
      <horsePower>292</horsePower>
    </engine>
    <year>2009</year>
  </car>
  <car>
    <model>A6</model>
    <engine model="FSI">
      <displacement>2.5</displacement>
      <horsePower>187</horsePower>
    </engine>
    <year>2012</year>
  </car>
  <car>
    <model>A6</model>
    <engine model="FSI">
      <displacement>2.8</displacement>
      <horsePower>220</horsePower>
    </engine>
    <year>2012</year>
  </car>
  <car>
    <model>A6</model>
    <engine model="TFSI">
      <displacement>3</displacement>
      <horsePower>295</horsePower>
    </engine>
    <year>2012</year>
  </car>
  <car>
    <model>A6</model>
    <engine model="TDI">
      <displacement>2</displacement>
      <horsePower>175</horsePower>
    </engine>
    <year>2011</year>
  </car>
  <car>
    <model>A6</model>
    <engine model="TDI">
      <displacement>3</displacement>
      <horsePower>309</horsePower>
    </engine>
    <year>2011</year>
  </car>
  <car>
    <model>S6</model>
    <engine model="TFSI">
```

```

        <displacement>4</displacement>
        <horsePower>414</horsePower>
    </engine>
    <year>2012</year>
</car>
<car>
    <model>S8</model>
    <engine model="TFSI">
        <displacement>4</displacement>
        <horsePower>513</horsePower>
    </engine>
    <year>2012</year>
</car>
</cars>

```

Załącznik 2

```

<!--?xml version="1.0" encoding="utf-8"?-->
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="pl" lang="pl"><head>
    <title>Cars</title>
    <meta http-equiv="content-type" content="text/html; charset=UTF-8">
</head>
<body>

    </body>
</html>

```