# Homework: XML Basic Concepts

This document defines the homework assignments from the ["Database Applications" Course @ Software University](https://softuni.bg/trainings/21/Database-Applications-Mar-2015). Please submit as homework a single zip / rar / 7z archive holding the solutions (source code) of all below described problems.

## Catalog of Musical Albums in XML Format

Create a XML file catalog.xml representing a catalog of musical **albums**. For each album you should define name, artist, year, producer, price and a list of songs. Each song should be described by title and duration.

Hint: You can take sample data from <https://gist.github.com/jasonbaldridge/2597611>.

## DOM Parser: Extract Album Names

Write a program that extracts **all album names** from **catalog.xml**. Use the DOM parser.

## DOM Parser: Extract All Artists Alphabetically

Write a program that extracts **all artists** in alphabetical order from **catalog.xml**. Use the DOM parser. Keep the artists in a SortedSet<string> to avoid duplicates and to keep the artist in alphabetical order.

## DOM Parser: Extract Artists and Number of Albums

Write a program that extracts **all different artists**, which are found in the **catalog.xml**. For each artist print the **number of albums** in the catalogue. Use the DOM parser and a Dictionary<string, int> (use the artist name as key and the number of albums as value in the dictionary).

## XPath: Extract Artists and Number of Albums

Implement the previous using XPath.

## DOM Parser: Delete Albums

Using the DOM parser write a program to **delete** from catalog.xml all albums having price > 20. Save the result in a new file cheap-albums-catalog.xml.

## DOM Parser and XPath: Old Albums

Write a program, which extract from the file catalog.xml the titles and prices for all albums, published 5 years ago or earlier. Use **XPath** query.

## LINQ to XML: Old Albums

Write a program, which extract from the file catalog.xml the titles and prices for all albums, published 5 years ago or earlier. Use XDocument and **LINQ to XML** query.

## \* XmlWriter: Directory Contents as XML

Write a program to traverse given directory and write to a **XML file** its contents together with all subdirectories and files. Use tags <file> and <dir> with attributes. For the generation of the XML document use the class XmlWriter. Sample output:

|  |
| --- |
| <?xml version="1.0" ?>  <root-dir path="C:\Example">  <dir name="docs">  <file name="tutorial.pdf" />  <file name="TODO.txt" />  <file name="Presentation.pptx" />  </dir>  <dir name="photos">  <dir name="birthday-4-march">  <file name="friends.jpg" />  <file name="the\_cake.jpg" />  <file name="baloons.jpg" />  </dir>  <dir name="travel">  <file name="IMG24152.jpg" />  </dir>  </dir>  </root-dir> |

\* Hint: search in Internet for "*directory traversal C# recursion*".

## XElement: Directory Contents as XML

Rewrite the previous task (build XML to hold directory contents) with **XDocument**, **XElement** and **XAttribute**.

## \* XML Schema Validation

Using Visual Studio generate an **XSD schema** for the file catalog.xml. Write a C# program that takes an **XML file** and an **XSD file** (schema) and **validates the XML** file against the schema. Test it with valid XML catalogs and invalid XML catalogs.

## \* XML to HTML through XSL Stylesheet

Create an **XSL stylesheet**, which transforms the file catalog.xml into HTML document, formatted for viewing in a standard Web-browser.

Write a C# program to apply the XSLT stylesheet transformation on the file catalog.xml using the class XslTransform.

## \*\*\* XML to HTML through XQuery

Read some tutorial about the **XQuery** language. Implement the XML to HTML transformation with XQuery (instead of XSLT). Download some **open source XQuery library for .NET** and execute the XQuery to transform the catalog.xml to HTML.