Assignment 13: XML Programming

- This assignment will be published on January 30th, 2017.
- You may work on the assignments in groups of up to three people (if possible, keep the same groups as in the previous assignments).
- The groups will present the solution of this assignment on February 7th, 2017 at 11:30. We will meet in seminar room 0.124.
- All members of a group have to show up together for the grading of the assignment. For the grading, each group will need to have the source code ready. Moreover, each group member might be asked questions about the solution.
- If you have questions, send an email to holger.schwarz@ipvs.uni-stuttgart.de.

Task 1 – DOM

The goal of this exercise is to understand and use the main DOM interfaces.

Task 1.1 – Reading elements and attributes from an XML document

Write a Java program that parses the file "scientists.xml" and uses the DOM interfaces to access and print each person's first name, last name, date of birth and date of death.

Refer to http://download.oracle.com/javase/6/docs/api/ for additional information on the API specification. Depending on your implementation, you may have to turn on namespace support so that the parser is able to recognize the xsi:noNamespaceSchemaLocation attribute.

Task 1.2 – Adding elements and attributes to an XML document

Extend your program as follows: In a first step, the program should add a person node to the DOM tree including your first name, last name and year of birth. In the second step, the program should print the entire information for all persons in the XML document (including the new person node and the job information). Make sure that your program works for any valid XML document according to the XML schema "people.xsd".

Task 2 – StAX

The goal of this exercise is to understand and use the Cursor API and the Iterator API of StAX.

Task 2.1 – Cursor API

Write a program that parses the file "scientists_dtd.xml" using the StAX Cursor API and prints out the event types and names of all parsing events, and also additional data (element names, attributes, namespaces, prefixes, comments, ...), if applicable. You do not have to put this data into Java objects for this task.

Task 2.2 – Iterator API

Solve the Task 1.1 using the StAX Iterator API. Compare the solutions.