

2. Unit: XSLT

Exercise 2.1 (XML to HTML with XSLT)

Solve Exercise 1.5 in XSLT.

Exercise 2.2 (Recursion in Data)

- (a) Write an XSLT stylesheet which maps the structures of the seas and rivers from Mondial in the following way: Every sea element must contain the name of the sea and a river element for each river flowing into that sea. Each river element, again, must recursively contain a river element for each river flowing into it, and so on:

```
<waters>
  <sea>
    <name>North Sea</name>
    <river>
      <name>Rhein</name>
      <length>...</length>
      <river>
        <name>Main</name>
        <length>...</length>
        <river>
          <name>Tauber</name>
          <length>...</length>
        </river>
      </river>
    </river>
  <river>
    <name>Neckar</name>
    <length>...</length>
  </river>
</waters>
```

- (b) Write another stylesheet (that uses the output of the above one as input) which computes for each river that flows into a sea the total sum of the length of all rivers flowing (directly or transitively) into it, and output the results into a table.
- (c) Write another stylesheet (that uses the original mondial.xml!) as input) which computes for each river that flows into a sea the total sum of the length of all rivers flowing (directly or transitively) into it, and output the results into a table.

Exercise 2.3 (Grouping an Input Sequence) Write an XSLT program that gets an input sequence of simple elements of the form given below (this file is linked as “DuplicateInput.xml” on the Web page): The final result should be the following: a table with columns called d,a,c,b. For the i -th table row, the element of the respective column should be the i -th element of that type in the input sequence, i.e.,

d	a	c	b
d: 31	a: 2	c: 43	b: 47
d: 42	a: 8	c: 65	b: 43
:	:	:	:

Preferably, produce the output as an HTML table.

This exercise illustrates a lot of features and properties of programming language concepts. The task seems to be quite obvious and easy. By hand, writing the result on a paper, this can be done quite easily by traversing the input sequence once. It is recommended to proceed as follows:

