

Scripting Languages

13.6 Exercises

- 13.19 Modify the XSLT of Figure ©13.26 to do one or more of the following:
- (a) Alter the titles of conference papers so that only first words, words that follow a dash or colon (and thus begin a subtitle), and proper nouns are capitalized. You will need to adopt a convention by which the creator of the document can identify proper nouns.
 - (b) Sort entries by the last name of the first author or editor. You will need to adopt a convention by which the creator of the document can identify compound last names (“von Neumann,” for example, should be alphabetized under ‘v’).
 - (c) Allow bibliographic entries to contain an `abstract` element, which when formatted appears as an indented block of text in a smaller font.
 - (d) In addition to the `book`, `article`, and `inproceedings` elements, add support for other kinds of entries, such as manuals, technical reports, theses, newspaper articles, web sites, and so on. You may want to draw inspiration from the categories supported by BibTeX [Lam94, App. B].
 - (e) Format entries according to some standard style convention (e.g., that of the Chicago Manual of Style [Uni03] or the ACM Transactions [www.acm.org/pubs/submissions/latex_style/index.htm]).
- 13.20 Suppose bibliographic entries in Figure ©13.25 contain a mandatory `key` element, and that other documents can contain matching `cite` elements. Create an XSLT script that imitates the work of BibTeX. Your script should
- (a) read an XML document, find all the `cite` elements, collect the keys they contain, and replace them with `bibref` elements that contain small integers instead.
 - (b) read a separate XML bibliography document, extract the entries with matching keys, and write them, in sorted order, to a new (and probably smaller) bibliography.

The small numbers in the `bibref` elements of the new document from (a) should match the corresponding numbered entries in the new bibliography from (b).

13.21 Write a program that will read an XHTML file and print an outline of its contents, by extracting all `<title>`, `<h1>`, `<h2>`, and `<h3>` elements, and printing them at varying levels of indentation. Write

- (a) in C or Java
- (b) in `sed` or `awk`
- (c) in Perl, Python, Tcl, or Ruby
- (d) in XSLT

Compare and contrast your solutions.