Example from XSLT, Tidwell, with DTD omitted:

```xml
<?xml version="1.0"?>
<?xml-stylesheet href="sonnet.xsl" type="text/xsl"?>
<?cocoon-process type="xslt"?>

<!-- Default sonnet type is Shakespearean, the other allowable -->
<!-- type is "Petrarchan." -->
<sonnet type="Shakespearean">
  <auth:author xmlns:auth="http://www.authors.com/"
                last-name="Shakespeare" first-name="William" nationality="British"
                year-of-birth="1564" year-of-death="1616">
    <!-- Is there an official title for this sonnet? They're sometimes named after the first line. -->
    <title>Sonnet 130</title>
    <lines>
      <line>My mistress' eyes are nothing like the sun,</line>
      <line>Coral is far more red than her lips red.</line>
      <line>If snow be white, why then her breasts are dun,</line>
      <line>If hairs be wires, black wires grow on her head.</line>
      <line>I have seen roses damasked, red and white,</line>
      <line>But no such roses see I in her cheeks.</line>
      <line>And in some perfumes is there more delight</line>
      <line>Than in the breath that from my mistress reeks.</line>
      <line>I love to hear her speak, yet well I know</line>
      <line>That music hath a far more pleasing sound.</line>
      <line>I grant I never saw a goddess go,</line>
      <line>My mistress when she walks, treads on the ground.</line>
      <line>And yet, by Heaven, I think my love as rare</line>
      <line>As any she belied with false compare.</line>
    </lines>
  </auth:author>
</sonnet>
<!-- The title of Sting's 1987 album "Nothing like the sun" is -->
<!-- from line 1 of this sonnet. -->
Class exercise:

What is selected by the following path expressions (assuming the context node is root):

/sonnet/lines/line
   a set containing all of the lines elements
line
   empty set – there are no child elements of root named line (the only child element of root is sonnet)
//line
   a set containing all of the lines elements
/sonnet/line
   a set containing all of the lines elements
/sonnet/@type
   the type attribute (and its value) of the sonnet element
/sonnet/auth:author/text()
   the spaces after the finishing “>” of auth:author and before the starting “<” of lastname
/sonnet/auth:author/last-name/text()
   the text “Shakespeare”
/processing-instruction()
   a set containing two processing instruction nodes (xml-stylesheet and cocoon-process -- <?xml version=… is not considered a PI)
/auth:*
   empty set – no children of root are in the auth: namespace
//auth:*
   returns only the auth:author element
/@auth:*
   empty set – there are no attributes in the auth: namespace
/sonnet[@type="Shakespearean"]
   the sonnet element is selected
XSLT Lecture

XSLT Example, to show how XSLT programs are processed.

In XSLT processing, the source XML document is processed using a preorder list traversal (each XML element is examined before its children) until the first matching template. Once the first template match has occurred, the sequence of XSLT statements inside that template is executed once for every node matching the template. Any output text/XML within the template is added to the output stream. Further processing is now controlled by additional apply-templates directives.

Input File:

```xml
<?xml version="1.0" encoding="utf-8"?>
<chapter name="Chapters One">
  <section>
    <title>An Obvious Vision</title>
    <page>1/2</page>
  </section>
  <section>
    <title>The Sense of Wonderful Developments</title>
    <page>1/6</page>
  </section>
  <section>
    <title>Two Cultures Face the Future</title>
    <page>1/11</page>
  </section>
  <section>
    <title>Hypertext</title>
    <page>1/14</page>
  </section>
</chapter>
```
XSLT Stylesheet
