XML, DOM, and Element Tree

School of Information Studies
Syracuse University

XML

Extended Markup Language

Format for data interchange

Design your own tag names

Sample XML Document

```xml
<?xml version='1.0' encoding='utf-8'?>
<feed xmlns='http://www.w3.org/2005/Atom' xml:lang='en'>
<CATALOG>
<CD>
<TITLE>Empire Burlesque</TITLE>
<ARTIST>Bob Dylan</ARTIST>
<COUNTRY>USA</COUNTRY>
<COMPANY>Columbia</COMPANY>
<PRICE>10.90</PRICE>
<YEAR>1985</YEAR>
</CD>
<CD>
<TITLE>Hide your heart</TITLE>
<ARTIST>Bonnie Tyler</ARTIST>
<COUNTRY>UK</COUNTRY>
<COMPANY>CBS Records</COMPANY>
<PRICE>9.90</PRICE>
<YEAR>1988</YEAR>
</CD>
</CATALOG>
</feed>
```

Simple Sample

Uses beginning and ending tags

- <foo>
- </foo>

Comments <!-- and -->

Predefined entities

&lt	less than
&gt	greater than
&amp	ampersand &
&apos	apostrophe ‘
&quot	quote “

DOM

Document Object Model

Used to parse the data

Converts entire text to a structure

Produces a node for each tag and its children

DOM Sample

```
>>> import urllib.request
>>> url = "http://feeds.bbci.co.uk/news/rss.xml"
>>> xmlstring = urllib.request.urlopen(url).read().decode('utf8')
>>> len(xmlstring)
35071
>>> xmlstring[:500]
'<?xml version="1.0" encoding="UTF-8"?>\n<?xml-stylesheet
title="XSL_formatting" type="text/xsl"
href="/shared/bsp/xsl/rss/nolsol.xsl"?>\n<rss
xmlns:dc="http://purl.org/dc/elements/1.1/"
xmlns:content="http://purl.org/rss/1.0/modules/content/"
xmlns:atom="http://www.w3.org/2005/Atom" version="2.0"
xmlns:media="http://search.yahoo.com/mrss/">\n    <channel>\n
<title><![CDATA[BBC News - Home]]></title>\n
<description><![CDATA[BBC News - Home]]></description>\n
<link>http://www.bbc.co'
```

Element Tree

Part of Python standard library

Main function is parse ()

Returns the tree structure

Attributes returned as Python dictionary

Element can be treated as a list