Lecture 13

Introduction to JSON

Role of Modeling the Data

Data modeling establishes the logical structure of a database

Data Model determines how data is

- Stored
- Organized
- Manipulated

Data

Information in relational databases is stored as per structure of relational model

Structured Data is highly predictable

Not all data is collected and inserted into carefully designed structured databases

Unstructured Data is highly unpredictable

e.g. Web pages

Data may have a certain structure,but not all the information collected will have the identical structure.

Some attributes may be shared among the various entities,
other attributes may exist only in a few entities.

Semi-structured Data is partially predictable

Data Transfer

Data transfer over the web is an important problem.

How will a web page be displayed when transmitted from server to client?

HTML - HyperText Markup Language

HTML

Basic HTML for **Static Web Pages**

Interactive web pages with customized content are called **Dynamic Web pages**

scripting languages i.e.

PHP, JavaScript, Python

Different formats to separate data from Presentation

JSON - JavaScript Object Notation

JSON is a syntax for storing and exchanging data

JSON is a language-independent data format

JSON grew out of a need for stateless, real-time, server-to-browser communication protocol without using browser plugins

JSON Syntax

➔ Data is in name/value pairs

```
"squadName": "Super hero squad"
```

➔ Data is separated by commas

```
"homeTown": "Metro City",
"formed": 2016,
"secretBase": "Super tower"
```

JSON Syntax

➔ Curly braces hold objects

```
{
  "squadName": "Super hero squad",
  "homeTown": "Metro City",
  "formed": 2016,
  "secretBase": "Super tower"
}
```

JSON Syntax

➔ Square brackets hold arrays

```
"powers": [
  "Radiation resistance",
  "Turning tiny",
  "Radiation blast"
]
```

JSON Example

```
{"employees":[
  { "firstName":"John", "lastName":"Doe" },
  { "firstName":"Anna", "lastName":"Smith" },
  { "firstName":"Peter", "lastName":"Jones" }
]}
```

Notes on JSON

JSON is purely a data format
    it contains only properties, no methods

JSON is self describing

JSON allows hierarchical data
        values within values

A single misplaced comma or colon can cause a JSON file to go wrong, and not work.

Introduction to XML

Minute to Win It !!!

What is the data about?

How many items are listed?

What information for each item is available?

Most expensive item in the data?

```
<breakfast_menu>
<food>
    <name>Belgian Waffles</name>
    <price>$5.95</price>
    <description>
   Two of our famous Belgian Waffles with plenty of real maple syrup
   </description>
    <calories>650</calories>
</food>
<food>
    <name>Strawberry Belgian Waffles</name>
    <price>$7.95</price>
    <description>
    Light Belgian waffles covered with strawberries and whipped cream
    </description>
    <calories>900</calories>
</food>
</breakfast_menu>
```

XML - eXtensible Markup Language

XML was designed to store and transport data.

XML was designed to be both human-readable & Self describing

XML allows hierarchical data (values within values)

XML

XML documents has two key structuring concepts

Elements & Attributes

Elements are identified in a document by

Start tag **<...>** & End tag **</...>**

Schema document

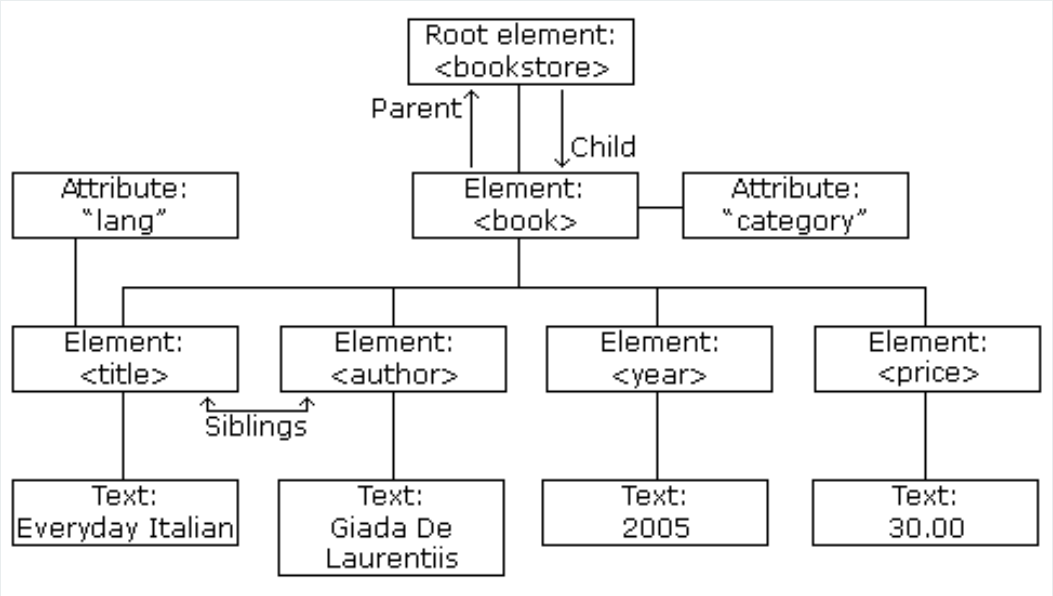
XML tag (element) names can be defined in separate document to give a semantic meaning to the tag

- Can be exchanged among programs and users

XML - Document Structure

XML documents follow a tree structure

- Starts at **the root**
- Branches to **the leaves**



XML Example

```
<employees>
  <employee>
    <firstName>John</firstName> <lastName>Doe</lastName>
  </employee>
  <employee>
    <firstName>Anna</firstName> <lastName>Smith</lastName>
  </employee>
  <employee>
    <firstName>Peter</firstName> <lastName>Jones</lastName>
  </employee>
</employees>
```

Which Documents are Valid?

Document 1

```
<note date="2008-01-10">
  <to>Tove</to>
  <from>Jani</from>
</note>
```

Document 2

```
<note>
  <date>2008-01-10</date>
  <to>Tove</to>
  <from>Jani</from>
</note>
```

Document 3

```
<note>
  <date>
    <year>2008</year>
    <month>01</month>
    <day>10</day>
  </date>
  <to>Tove</to>
  <from>Jani</from>
</note>
```

The three XML documents contain exactly the same

XML Attributes

- Cannot contain multiple values (elements can)
- Cannot contain tree structures (elements can)
- Not easily expandable (for future changes)

```
<note day="10" month="01" year="2008"
to="Tove" from="Jani" heading="Reminder">
</note>
```

XML Attributes

Metadata should be stored as attributes not data itself

```
<messages>
  <note id="501">
    <to>Tove</to>
    <from>Jani</from>
    <heading>Reminder</heading>
  </note>
  <note id="502">
    <to>Jani</to>
    <from>Tove</from>
    <heading>Re: Reminder</heading>
  </note>
</messages>
```

XML DOM - Data Object Model

The DOM defines a standard for accessing and manipulating XML documents

DOM considers an XML document as a set of node objects
Programming interface to the DOM is defined by **properties** and **methods**.

- x.nodeName - the name of x
- x.nodeValue - the value of x
- x.parentNode - the parent node of x
- x.childNodes - the child nodes of x
- x.attributes - the attributes nodes of x

- x.getElementsByTagName(*name*) - get all elements with a specified tag name
- x.appendChild(*node*) - insert a child node to x
- x.removeChild(*node*) - remove a child node from x

XML in Industry

Thousands of XML formats exists, in many different industries, to describe day-to-day data transactions:

- Stocks and Shares
- Financial transactions
- Medical data
- Mathematical data
- Scientific measurements
- News information
- Weather services