Week 2 Lecture 5

Applied

What's in this lecture?

- Encoding Data: XML and JSON
- Extending last lecture's introduction to Arrays and Hashes

What's encoding?

- Encoding is the format and structure with which data is stored
- Depending on the type of data and its use, some encoding is preferred over others
- Typical use is when describing an object

Morse Code

- hello world:

.... . .-.. .-.. --- / .-- --- .-. .-.. -..

Health Form

- Name: George Hill
- DOB: 06/01/1966
- Blood Type: A

Problem

- One is ‘machine readable’, other is ‘human readable’

What is XML?

- eXtensible Markup Language
- built with the intent of describing data objects
- easily extendable

XML: Tags

- an open tag:

<name>

- must be closed:

</name>

- and can contain a value:

<name>Kip</name>

XML: Attributes

- Tags can have an attribute:
  <name type="first">Kip</name>
- that only applies to that XML element
- Can also be multiples:
  <name type="first" origin="swedish">Kip</name>

XML: General Format

```
<parent_tag attribute1="value1" attribute2="value2">
  <child_tag>value</child_tag>
  <child_tag>value</child_tag>
</parent_tag>
```

Health Form in XML

```
<health_form>
  <name>George Hill</name>
  <dob>06/01/1966</dob>
  <blood_type>A</blood_type>
</health_form>
```

What does it look like?

A Hash!

```
<hash>
  <key1>value1</key1>
  <key2>value2</key2>
</hash>
```

Multiples of Data:

```
<GetCorporateDetails>
 <SuccessCode status="OK"/>
 <LastModifiedDate>2011-06-09T20:56:15.000Z</LastModifiedDate>
 <CorporateLevel>
  <Photos>
   <MediaImages/>
   <GalleryImages>
    <ImageReference sequenceNumber="0" lastModifiedDate="2011-06-08T13:49:46.000Z">
     <URL>boston_gal01_exterior_view_iphone_med.jpg</URL>
     <ThumbnailUrl>boston_gal01_exterior_view_iphone_thumb.jpg</ThumbnailUrl>
     <LargeUrl>boston_gal01_exterior_view_iphone_med.jpg</LargeUrl>
     <PictureDescription1>Boston</PictureDescription1>
    </ImageReference>
    <ImageReference sequenceNumber="1" lastModifiedDate="2011-06-08T13:50:57.000Z">
     <URL>lasvegas_gal01_exterior_view_dusk_iphone_med.jpg</URL>
     <ThumbnailUrl>lasvegas_gal01_exterior_view_dusk_iphone_thumb.jpg</ThumbnailUrl>
     <LargeUrl>lasvegas_gal01_exterior_view_dusk_iphone_med.jpg</LargeUrl>
     <PictureDescription1>Las Vegas</PictureDescription1>
    </ImageReference>
.....
</GetCorporateDetails>
```

Wait a minute:

Hashes have to have unique keys!

XML also can store data as an array

JSON

What is JSON?

- JavaScript Object Notation
- Lightweight, human readable, JavaScript consumable
- Also maps extremely well to data objects

JSON: Hashes

- Simplified, single key:
  {“key”:“value”}

- Multiple keys:
  {"key":"value","key2":"we win"}

- Strings are keys, values can be:
  numbers; strings; null; true; false

JSON: Arrays

- ["orange","bananana"]
- Looks familiar!

- [“value1”, “value2”, ..., “valueN”]
- Values can be:
  string, number, true, false, null, another object, or another array!

JSON: Putting together

```
{“fruits” : [
        {“name”:“banana”,
          “weight”:“3oz”},
        {“name”:“apple”,
          “weight”:“4oz”},
        {“name”:“orange”,
          “weight”:“8oz”}
]}
```

JSON: Evaluation

```
var simpleJSON = {"key1":"i am an object", "key2":"value2"};
alert(simpleJSON.key1);
```

```
eval('(' + 'kip = {"key1":"I am evald", "key2":"value2"}' + ')');
alert(kip.key1);
```

What's happening here?

- eval('(' + ourText + ')');
- eval is the JavaScript compiler and JSON is just JavaScript
- ourText is passed to the compiler, where an object is returned

Exercises

- Write your 'Info' page on your Facebook Profile as XML and JSON
- Write a JS function that takes JSON as input and displays it with correct <p> tags within a DIV element on the current HTML page