

## 6-2. JSON 표기법

JSON(JavaScript Object Notation)표기법은 서로다른 프로그래밍언어간에 데이터를 교환하기위한 표기법으로서 읽고 쓰기쉬운 표기법이다.

JSON 표기법은 데이터를 다음과같이 두가지형태로 표시한다.

- 이름/값 쌍(자바의 Map 이나 Hashtable과 같은방식)
- 배열
- 이름/값 쌍을 표현

```
{ 이름1:값1 , 이름2:값2 , 이름3:값3 }
```

예를들면

```
var countries={ ko: '대한민국' , fr: '프랑스' , uk: '영국'}  
var korName  =  countries.ko;  
var frName   =  countries['fr']
```

- 배열을 표현

```
[ 값0 , 값1 , 값2 , 값3 ]
```

예를들면

```
var countryCodes=[ 'ko' , 'fr' , 'uk' , 'us']  
var idx0  = countryCodes[0];  // 'ko'  
var idx2  = countryCodes [2]; // 'uk'
```

### ① JSON 표기법을 사용한 새로운 클래스정의

```
클래스 이름 = function (파라메타){  
  .. ..  
}  
클래스이름.prototype={  
  함수명:function(파라메타1) {.....},  
  함수명2:function(파라메타2){.....}  
}
```

JSON 표기법의 ‘이름/값’ 부분에서 함수이름이 ‘이름’ 부분에 들어가고 함수정의가 ‘값’ 부분에 들어간다.

Member 클래스를 JSON표기법으로 다시 작성해보면 다음과 같다.

member\_json.js

```
var member=new Object();
member.Member = function(name, id, securityNo) {
    this.name = name;
    this.id = id;
    this.securityNo = securityNo;
}
member.Member.prototype = {
    setValue: function(newName, newId, newSecurityNo) {
        this.name = newName;
        this.id = newId;
        this.securityNo = newSecurityNo;
    },
    getAge: function() {
        var birthYear = parseInt(this.securityNo.substring(0,
2));

        var code = this.securityNo.substring(6,7);
        if (code == '1' || code == '2') {
            birthYear += 1900;
        } else if (code == '3' || code == '4') {
            birthYear += 2000;
        }

        var today = new Date();
        return today.getFullYear() - birthYear;
    },
    toString: function() {
        return this.name + "[" + this.id + "]";
    }
}
```

① 자바 스크립트에서 패키지정의

자바스크립트에서 모듈을 만들다보면 다른사람이만든 모듈과 겹치는경우가 발생할수있다.이런문제를 해결하기위하여 보통자바나 C# 과같은 언어는 패키지 개념을 제공하고있다.

자바스크립트는 자체적으로 이런개념을 제공하지는 않지만 ,객체의 프로퍼티를 사용해서 패키지 개념을 적용할수있다.

- Ajax package 추가

```
var ajax=new Object();
ajax.Request=function(){  // Request 클래스 정의
    ....
}
ajax.Request.prototype={  //Request 클래스에 함수추가
    someFunction:function(){
        .....
    }
}
```

## ② 클래스로 XMLHttpRequest 모듈만들기.

ajax.js

```
var ajax = {}; ajax.xhr = {};  
  
ajax.xhr.Request = function(url, params, callback, method) {  
    this.url = url; this.params = params; this.callback = callback;  
    this.method = method;this.send();  
}  
  
ajax.xhr.Request.prototype = { getXMLHttpRequest: function() {  
    if (window.ActiveXObject) {  
        try {return new ActiveXObject("Msxml2.XMLHTTP");  
        } catch(e) {try {  
            return new ActiveXObject("Microsoft.XMLHTTP");  
        } catch(e1) { return null; }}  
    } else if (window.XMLHttpRequest) {  
        return new XMLHttpRequest();  
    } else {return null;}},send: function() {  
    this.req = this.getXMLHttpRequest();  
    var httpMethod = this.method ? this.method : 'GET';  
    if (httpMethod != 'GET' && httpMethod != 'POST') {  
        httpMethod = 'GET';  
    }  
    var httpParams = (this.params == null || this.params == '') ?  
        null : this.params;  
    var httpUrl = this.url;  
    if (httpMethod == 'GET' && httpParams != null) {  
        httpUrl = httpUrl + "?" + httpParams;  
    }  
    this.req.open(httpMethod, httpUrl, true);  
    this.req.setRequestHeader(  
        'Content-Type', 'application/x-www-form-urlencoded');  
    var request = this;  
    this.req.onreadystatechange = function() {  
        request.onStateChange.call(request);  
    }  
    this.req.send(httpMethod == 'POST' ? httpParams : null);  
    },onStateChange: function() {  
        this.callback(this.req);  
    }}  
}
```

③ ajax.xhr.Request 클래스를 사용하는 방법은 다음과 같다.

```
new ajax.xhr.Request("books.jsp", null, loadedBooksXML, "GET");
function loadedBooksXML(req){
    if(req.readyState==4){
        if(req.status==200){
            xmlDoc = req.responseXML;
            doXSLT();
        }
    }
}
```

#### 4. 응답결과를 자바 스크립트 객체로 저장하기

member.xml

```
<?xml version="1.0" encoding="euc-kr" ?>
<result>
    <code>success</code>
    <data>
        <member>
            <name>배수지</name>
            <id>suzi</id>
            <sno>9410102000000</sno>
        </member>
    </data>
</result>
```

## loadMemberFromXML.html

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ko" lang="ko">
<head><meta http-equiv="content-type" content="text/html; charset=euc-kr" />
    <title>XML로부터 객체 생성</title>
<script type="text/javascript" src="js/ajax.js"></script>
<script type="text/javascript" src="js/member.js"></script>
<script type="text/javascript">
window.onload = function() {
    new ajax.xhr.Request("member.xml", "", viewInfo, "GET");
}
function viewInfo(req) {
    if (req.readyState == 4) {
        if (req.status == 200) {
            var docXML = req.responseXML;
            var code = docXML.getElementsByTagName("code")
                .item(0).firstChild.nodeValue;
            if (code == 'success') {
                var name = docXML.getElementsByTagName("name")
                    .item(0).firstChild.nodeValue;
                var id = docXML.getElementsByTagName("id")
                    .item(0).firstChild.nodeValue;
                var sno = docXML.getElementsByTagName("sno")
                    .item(0).firstChild.nodeValue;
                var mem = new Member(name, id, sno);
                alert(mem.toString() + ", " + mem.getAge() + "세");
            } else {
                alert("실패");
            }
        } else {
            alert("에러 발생:"+req.status);
        }
    }
}
</script></head>
<body></body>
</html>
```

## 5. JSON 응답결과를 자바스크립트 객체로 변환하기

```
<%@ page contentType="text/plain; charset=euc-kr" %>

{
    code: 'success',
    data: {
        member: {
            name: '배수지',
            id: 'suzi',
            sno: '9410102000000'
        }
    }
}
```

## loadMemberFromJSON.html

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ko"
lang="ko">
<head><meta http-equiv="content-type" content="text/html;
charset=euc-kr" />
    <title>JSON으로부터 객체 생성</title>
<script type="text/javascript" src="ja/ajax.js"></script>
<script type="text/javascript">
    window.onload = function() {
        new ajax.xhr.Request("member_json.jsp", "",
viewInfo, "GET");
    }
    function viewInfo(req) {
        if (req.readyState == 4) {
            if (req.status == 200) {
                var result = eval("(" +
req.responseText + ")");
                if (result.code == 'success') {
                    var m = result.data.member;
                    alert(m.id + "[" + m.name +
"]");
                } else {
                    alert("실패");
                }
            } else {
                alert("에러 발생:"+req.status);
            }
        }
    }
</script>
</head>
<body></body></html>
```