



# JSP 프로그래밍

## 6차시

- 달력 만들기
- XML과 JSON



## □ 사용자의 정보를 지속적으로 유지하기 위해 사용

○ http 프로토콜은 비 연결지향형 통신이므로 클라이언트에서 서버로 정보를 보내 처리 결과를 클라이언트가 받으면 기존의 정보를 잃어버리기 때문에 정보를 유지하는 방법이 필요하다.

○ 대표적으로 로그인, 장바구니 등이 있음





## □ 쿠키

- 클라이언트에 사용자 정보를 저장
- 클라이언트에 저장하므로 서버의 부하를 줄일 수 있지만
- 보안상 위험할 수 있다
- 데이터의 저장량이 1.2MB로 제한적이다.





## □ 세션

- 서버에 사용자 정보를 저장
- 서버 상에 존재하는 객체로 브라우저 단위 당 1개씩 존재
- 쿠키에 비하여 보안이 좋다
- 웹 브라우저를 닫기 전까지 유지



## □ 달력 만들기

- 소스 코드 그냥 따라 치는건 아무 의미가 없음
- 이해하고, 여태까지 배운 것들 응용하자
- 문제 해결 능력을 키우자
- 작은 부분으로 쪼개자
- 할 수 있는 것부터 하면서 성공해야 자신감도 차오른다.

# 달력 만들기



현재 년: 2015

현재 월: 11

<u>이전달</u>		2015년 11월			<u>다음달</u>	
일	월	화	수	목	금	토
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30					



## □ 문제 나누기

- 내가 자신있는 단위로 나누자
- 달력을 만들고
- 화면에 뿌리고
- 오늘 표시하고
- 이전달, 다음달 만들기



## □ XML이란?

- eXtensible Markup Language

- W3C에서 여러 특수 목적의 마크업 언어를 만드는 용도에서 권장되는 다목적 마크업 언어





## □ XML탄생배경

- HTML과 SGML의 한계를 극복하기 위한 대안
- 웹 사용자들의 다양한 요구
- SGML의 복잡성으로 인한 소프트웨어 개발의 어려움으로 인해 탄생



- XML은 주로 다른 종류의 시스템, 특히 인터넷에 연결된 시스템끼리 데이터를 쉽게 주고 받을 수 있게 하여 HTML의 한계를 극복할 목적으로 만들어짐
- 사람과 기계 모두가 읽을 수 있는 형식을 갖도록 규정하고 있다.



## □ JavaScript Object Notation

- 경량의 DATA-교환 형식
- 속성-값의 쌍으로 이루어진 구조
  - 자바의 Map과 비슷하다
- 오브젝트를 전달하기 위해 인간이 읽을 수 있는 텍스트를 사용하는 개방형 표준 포맷



## □ 특징

- 자바스크립트로부터 파생되었지만 언어에 독립적인 데이터 포맷이다.
- 프로그래밍 언어가 아님. 형식일뿐
- 공식 미디어 타입은 application/json
- 텍스트 기반이므로 사람과 기계 모두 읽고 쓰기 쉽다.



## □ 데이터 교환을 위한 단순한 포맷

- 통신 방법(Ajax), 프로그래밍 문법(Javascript)가 아님
- 단순히 데이터를 표시하는 표현 방법일 뿐

## □ xml보다 더 간단하게 표현이 가능하고, 파싱이 빠르다

- 현재 가장 많이 사용하는 데이터 교환 포맷



## □ 객체 샘플

- 객체는 이름/값 쌍의 집합으로, 중괄호{}를 사용한다. 이름은 문자열이기 때문에 반드시 따옴표를 하며, 값은 기본 자료형이거나 배열, 객체이다. 각 쌍들은 쉼표(,)로 구별된다. 각 쌍이 나오는 순서는 의미가 없다.
- {"name2": 50, "name3": "값3", "name1": true}



## □ 한 사람의 관한 정보를 갖는 JSON 객체

```
1 {  
2   "이름": "테스트",  
3   "나이": 25,  
4   "성별": "여",  
5   "주소": "서울특별시 양천구 목동",  
6   "특기": ["농구", "도술"],  
7   "가족관계": {"#": 2, "아버지": "홍판서", "어머니": "춘섬"},  
8   "회사": "경기 안양시 만안구 안양7동";  
9 }
```

# JSON



## xml 포맷

```
<boxOfficeResult>
  <boxofficeType>일별 박스오피스</boxofficeType>
  <showRange>20120101~20120101</showRange>
  <dailyBoxOfficeList>
    <dailyBoxOffice>
      <rnum>1</rnum>
      <rank>1</rank>
      <rankInten>0</rankInten>
      <rankOldAndNew>OLD</rankOldAndNew>
      <movieCd>20112207</movieCd>
      <movieNm>미션임파서블:고스트프로토콜</movieNm>
      <openDt>2011-12-15</openDt>
      <salesAmt>2776060500</salesAmt>
      <salesShare>36.3</salesShare>
      <salesInten>-415699000</salesInten>
      <salesChange>-13</salesChange>
      <salesAcc>40541108500</salesAcc>
      <audiCnt>353274</audiCnt>
      <audiInten>-60106</audiInten>
      <audiChange>-14.5</audiChange>
      <audiAcc>5328435</audiAcc>
      <scrnCnt>697</scrnCnt>
      <showCnt>3223</showCnt>
    </dailyBoxOffice>
  </dailyBoxOfficeList>
</boxOfficeResult>
```

## json 포맷

```
{
  - boxOfficeResult: {
    boxofficeType: "일별 박스오피스",
    showRange: "20120101~20120101",
    - dailyBoxOfficeList: [
      - {
        rnum: "1",
        rank: "1",
        rankInten: "0",
        rankOldAndNew: "OLD",
        movieCd: "20112207",
        movieNm: "미션임파서블:고스트프로토콜",
        openDt: "2011-12-15",
        salesAmt: "2776060500",
        salesShare: "36.3",
        salesInten: "-415699000",
        salesChange: "-13",
        salesAcc: "40541108500",
        audiCnt: "353274",
        audiInten: "-60106",
        audiChange: "-14.5",
        audiAcc: "5328435",
        scrnCnt: "697",
        showCnt: "3223"
      },
      - {
        rnum: "2",
        rank: "2",
        rankInten: "0",
        rankOldAndNew: "OLD",
        movieCd: "20112207",
        movieNm: "미션임파서블:고스트프로토콜",
        openDt: "2011-12-15",
        salesAmt: "2776060500",
        salesShare: "36.3",
        salesInten: "-415699000",
        salesChange: "-13",
        salesAcc: "40541108500",
        audiCnt: "353274",
        audiInten: "-60106",
        audiChange: "-14.5",
        audiAcc: "5328435",
        scrnCnt: "697",
        showCnt: "3223"
      }
    ]
  }
}
```