

빅데이터 분석 결과 시각화

D3.js를 활용한 데이터 시각화

학습내용

- CSV를 활용한 데이터 시각화
- SVG를 활용한 데이터 시각화
- JSON을 활용한 데이터 시각화

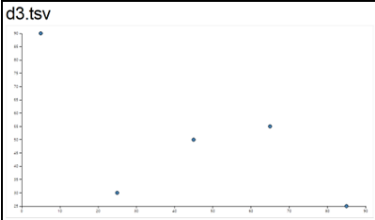
학습목표

- CSV 파일 형식에 대한 이해를 바탕으로 D3.js로 데이터를 시각화할 수 있다.
- SVG 파일 형식에 대한 이해를 바탕으로 D3.js로 데이터를 시각화할 수 있다.
- JSON 파일 형식에 대한 이해를 바탕으로 D3.js로 데이터를 시각화할 수 있다.

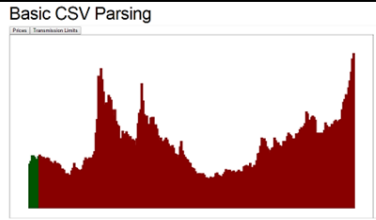
● CSV를 활용한 데이터 시각화

1. CSV 개요

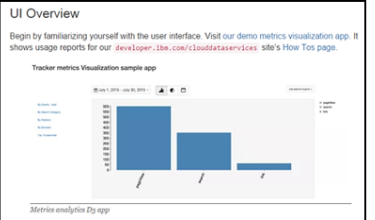
- ◆ D3.js에서 활용할 수 있는 데이터
 - 단순 배열 데이터 TSV, CSV, JSON, HTML, XML, TEXT 파일을 불러낼 수 있음



D3.tsv



D3.csv



JSON

<출처 : <http://bl.ocks.org/mbostock/3305937>>

<출처 : <http://bl.ocks.org/enjalot/1525346>>

<출처 : developer.ibm.com>

◆ D3.js 외부 데이터 종류별 특징

용어	의미
TSV	데이터가 탭으로 구분된 형식을 말하며 헤더가 있음
CSV	반점으로 구분한 텍스트, 헤더가 있음
JSON	Javascript Object Notation의 약자로, 간단하고 데이터 용량도 적으며 헤더가 없음
HTML	브라우저로 데이터를 보기 좋게 표시해주며 레이아웃이 있음
XML	태그를 이용하여 데이터를 표현하며 범용적인 데이터 포맷으로 SVG도 XML의 종류 중 하나임
TEXT	기본 문자로 구성된 데이터 파일

● CSV를 활용한 데이터 시각화

1. CSV 개요

◆ CSV의 정의

- CSV(Comma-Separated Values)
 - 쉼표를 기준으로 항목을 구분하여 저장한 데이터
 - 데이터베이스나 표 계산 소프트웨어 데이터를 보존하기 위해 이런 형식 사용
 - 텍스트 파일로 보존하여 문서 처리기나 편집기에서 열람 · 편집할 수 있음
 - CSV의 파일 확장자 : .csv

◆ CSV.html 소스

```
<!DOCTYPE html>
<html lang="ko">
<head>
<meta charset="utf-8">
<title>CSV 파일 불러오기</title>
<script src="//d3js.org/d3.v3.min.js"></script>
</head>

  <body>
    <script type="text/javascript">
      d3.csv("cook.csv", function(data) {
        console.log(data);
      });
    </script>
  </body>
</html>
```

← JavaScript파일

← CSV 파일 로딩

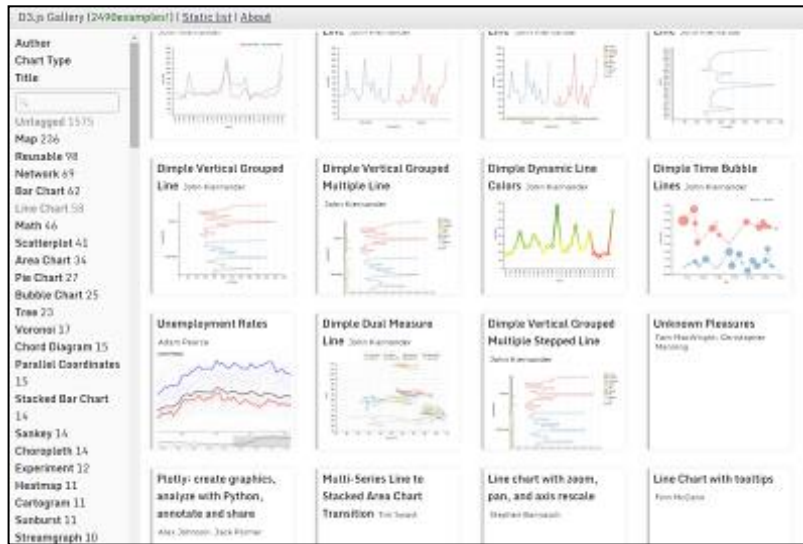
● CSV를 활용한 데이터 시각화

1. CSV 개요

◆ CSV 최신기술 동향

● 데이터 시각화 갤러리

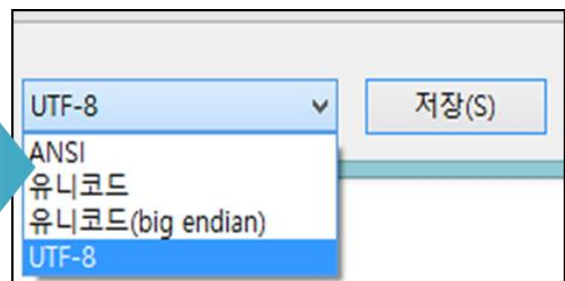
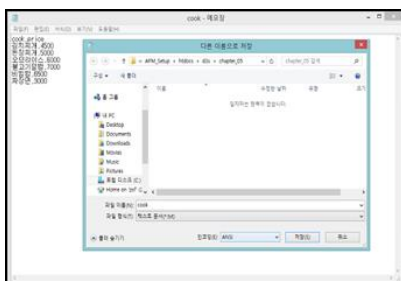
- 다양한 종류의 그래프와 트리 차트를 예제를 통하여 볼 수 있음
- D3.js 갤러리로 직접 소스를 통해 많은 예제를 볼 수 있어 학습하기 좋음



<출처 : <http://christopheviau.com/d3list/gallery.html>>

● CSV 한글 깨짐 방지

- CSV 파일을 로드 했을 때 한글이 깨지는 경우 CSV 파일을 utf-8로 다시 저장함



● CSV를 활용한 데이터 시각화

2. CSV 데이터 시각화 실습

◆ CSV 데이터 시각화 실습 순서

- ① 데이터 준비하기
- ② APM 서버에 데이터 저장하기
- ③ HTML 소스 작성하기
- ④ Console에서 CSV 파일 확인하기

● SVG를 활용한 데이터 시각화

1. SVG 개요

◆ SVG의 정의

- 스케일러블 벡터 그래픽스(Scalable Vector Graphics)
 - 2차원 벡터 그래픽을 표현하기 위한 XML 기반의 파일 형식
 - 1999년 W3C(World Wide Web Consortium)의 주도하에 개발된 오픈 표준의 벡터 그래픽 파일 형식
- SVG 형식의 이미지와 그 작동은 XML 텍스트 파일들로 정의됨
 - 검색화 · 목록화 · 스크립트화가 가능하며 필요하다면 압축도 가능함

◆ SVG 특징

- SVG 형식의 파일 편집
 - SVG 기반의 전문 그래픽 편집 프로그램을 사용하여 편집 가능
 - 물론 XML 파일로 되어 있으므로 문서 편집기로도 편집 가능
- SVG 파일 지원
 - 현재 마이크로소프트의 인터넷 인터넷 익스플로러 8 이전 버전을 제외한 대부분의 주요 웹 브라우저들은 SVG를 지원함
 - 인터넷 인터넷 익스플로러 에서는 SVG 파일을 보기 위해 별도의 플러그인을 수동으로 설치하여야 함
 - 설치 하지 않은 경우에는 웹 페이지 제작자가 구글 코드에서 개발중인 JavaScript 라이브러리, SVG Web을 웹 페이지 코드에 포함시켜야 함

● SVG를 활용한 데이터 시각화

1. SVG 개요

◆ SVG 장점

- XML로 기술하므로 웹 브라우저 상에서 열람할 수 있고 문서 편집기 등에서 편집할 수 있음
- 하이퍼링크를 이미지에 걸거나, JavaScript 등과 연동시킬 수도 있음
- 벡터 그래픽스이므로 확대나 축소를 해도 화질에는 변화가 없음

예)

```
<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
"http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">

<svg xmlns="http://www.w3.org/2000/svg"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:ev="http://www.w3.org/2001/xml-events" version="1.1"
baseProfile="full" width="800mm" height="600mm"> <!-- 중략 -->

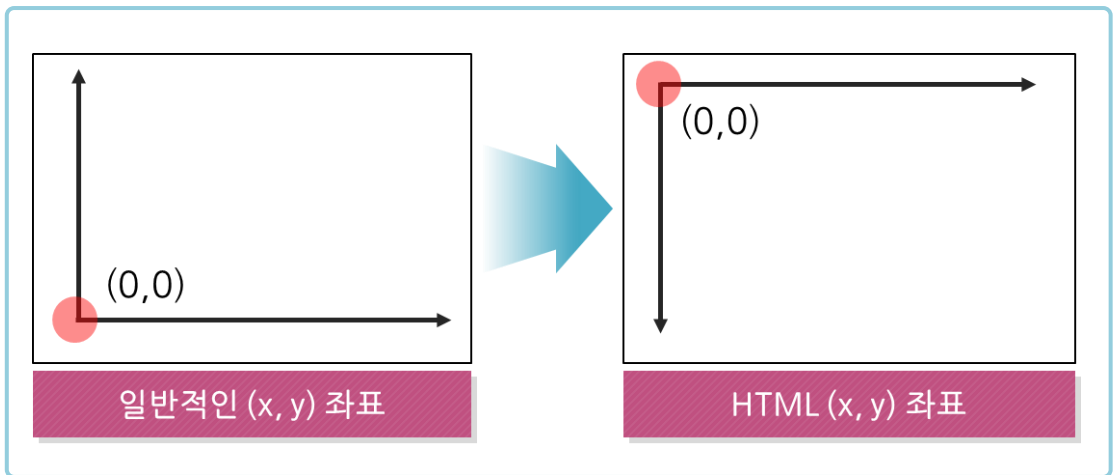
</svg>
```


● SVG를 활용한 데이터 시각화

1. SVG 개요

◆ SVG 좌표의 이해

- 그래프를 그릴 때 좌표는 일반적인 좌표와 HTML 기준의 좌표가 다름
- X와 y의 기준은 왼쪽 상단 부터 시작함



● SVG를 활용한 데이터 시각화

2. SVG 예제

◆ 원 그리기

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<svg height="100" width="100">
```

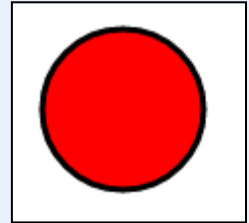
```
<circle cx="50" cy="50" r="40" stroke="black" stroke-width="3" fill="red" />
```

SVG가 당신의 브라우저에서 지원하지 않습니다.

```
</svg>
```

```
</body>
```

```
</html>
```



◆ 폴리라인 그리기

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<svg height="200" width="500">
```

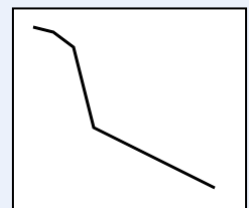
```
<polyline points="20,20 40,25 60,40 80,120 120,140 200,180 " style="fill:none;stroke:black;stroke-width:3" />
```

Sorry, your browser does not support inline SVG.

```
</svg>
```

```
</body>
```

```
</html>
```

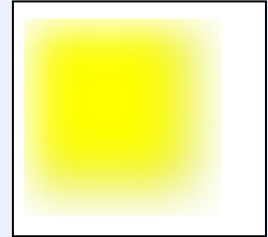


● SVG를 활용한 데이터 시각화

2. SVG 예제

◆ 필터 효과 주기

```
<svg height="110" width="110">
  <defs>
    <filter id="f1" x="0" y="0">
      <feGaussianBlur in="SourceGraphic" stdDeviation="15" />
    </filter>
  </defs>
  <rect width="90" height="90" stroke="green" stroke-width="3"
  fill="yellow" filter="url(#f1)" />
  Sorry, your browser does not support inline SVG.
</svg>
```



◆ Index.html SVG 파일 불러오기

```
<!DOCTYPE html>
<meta charset="utf-8">
<body>
  <script src="//d3js.org/d3.v3.min.js"></script>
  <script>
```

✓ SVG 파일 불러오기

```
d3.xml("rect01.svg", "image/svg+xml", function(error, xml) {
  if (error) throw error;
  document.body.appendChild(xml.documentElement);
});
```

● SVG를 활용한 데이터 시각화

2. SVG 예제

◆ 사각형 만들기

```
</script>
```

```
<?xml version="1.0" standalone="no"?>
```

```
<!DOCTYPE svg PUBLIC "-//W3C//DTD SVG 1.1//EN"
```

```
"http://www.w3.org/Graphics/SVG/1.1/DTD/svg11.dtd">
```

```
<svg width="600px" height="400px" viewBox="0 0 600 400"
```

```
xmlns="http://www.w3.org/2000/svg" version="1.1">
```

```
<desc>사각형 만들기</desc>
```

```
<!-- 'rect' 엘리먼트요소를 이용하여 사각형 그리기-->
```

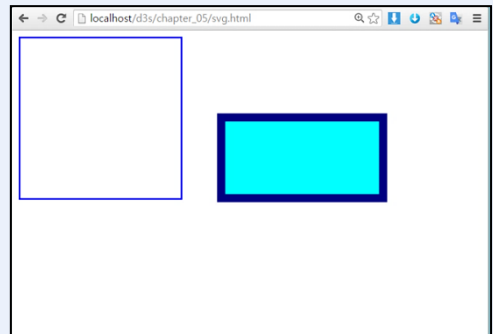
```
<rect x="1" y="1" width="200" height="200"
```

```
fill="none" stroke="blue" stroke-width="2"/>
```

```
<rect x="250" y="100" width="200" height="100"
```

```
fill="cyan" stroke="navy" stroke-width="10" />
```

```
</svg>
```



● SVG를 활용한 데이터 시각화

2. SVG 예제

- ◆ 브라우저 기반 SVG 확대 / 축소 및 이동 방법
 - 브라우저의 기본 기능 사용
 - 인터넷 익스플로러 9의 확대/축소 키

사용자 작업	바로가기 키	마우스 바로 가기 키
확대	Ctrl + 더하기(+)	Ctrl + <앞으로 휠 회전>
축소	Ctrl + 빼기(-)	Ctrl + <뒤로 휠 회전>
기본 확대 / 축소 수준으로 돌아가기	Ctrl + 0	해당 없음

● SVG를 활용한 데이터 시각화

3. SVG 데이터 시각화 실습

◆ SVG 데이터 시각화 실습 순서

- ① CSS 설계하기
- ② JavaScript 설계하기
- ③ 라인함수 작성하기
- ④ SVG 설정하기
- ⑤ SVG 그래프 결과화면 확인하기

● JSON를 활용한 데이터 시각화

1. JSON 개요

◆ JSON의 정의

- JSON(제이슨, JavaScript Object Notation)
 - 속성-값 쌍으로 이루어진 데이터 오브젝트를 전달하기 위해 인간이 읽을 수 있는 텍스트를 사용하는 개방형 표준 포맷
 - 비동기 브라우저 / 서버 통신(AJAX)을 위해, 넓게는 XML(AJAX가 사용)을 대체하는 주요 데이터 포맷임
 - 특히, 인터넷에서 자료를 주고 받을 때 그 자료를 표현하는 방법으로 알려져 있음
- 자료의 종류에 큰 제한은 없으며, 특히 컴퓨터 프로그램의 변수값을 표현하는 데 적합함
- 본래는 JavaScript 언어로부터 파생되어 JavaScript의 구문 형식을 따르지만 언어 독립형 데이터 포맷임
 - 프로그래밍 언어나 플랫폼에 독립적임
 - 구문 분석 및 JSON 데이터 생성을 위한 코드는 C, C++, C#, Java, JavaScript, 펄, 파이썬 등 수많은 프로그래밍 언어에서 쉽게 이용할 수 있음
- JSON 포맷은 본래 더글라스 크록포드가 규정함
- RFC 7159와 ECMA-404라는 두 개의 경쟁 표준에 의해 기술됨

RFC 7159	ECMA-404
<ul style="list-style-type: none">• 시맨틱, 보안적 고려 사항을 일부 제공하기도 함	<ul style="list-style-type: none">• 문법만 정의할 정도로 최소한으로만 정의되어 있음

- JSON의 공식 인터넷 미디어 타입 : application/JSON
- JSON의 파일 확장자 : .json

● JSON를 활용한 데이터 시각화

1. JSON 개요

◆ JSON 특징

- JSON은 텍스트로 이루어져 있으므로, 사람과 기계 모두 읽고 쓰기 쉬움
- 프로그래밍 언어와 플랫폼에 독립적이므로, 서로 다른 시스템간에 객체를 교환하기에 좋음
- JavaScript의 문법을 채용했기 때문에, JavaScript에서 eval 명령으로 곧바로 사용할 수 있음
 - 이런 특성은 JavaScript를 자주 사용하는 웹 환경에서 유리함
- 그러나 실질적으로 eval 명령을 사용하면 외부에서 악성 코드가 유입될 수 있음
- 모질라 파이어폭스 3.5, 인터넷 인터넷 익스플로러 8, 오페라 10.5, 사파리, 구글 크롬 등 대부분의 최신 웹 브라우저는 JSON 전용 파서 기능을 내장하고 있음
 - JSON 전용 파서 기능을 사용하는 것이 더 안전할 뿐만 아니라 빠른 방법임

● JSON를 활용한 데이터 시각화

1. JSON 개요

◆ JSON 소스파일

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title>D3: Loading data from a json file</title>
<script src="//d3js.org/d3.v3.min.js"></script>
</head>
<body>
<script type="text/javascript">
    d3.csv("hong.json", function(data) {
        console.log(data);
    });
</script>
</body>
</html>
```

← D3.js 파일 불러오기

← JSON파일 불러오기

● "hong.json"

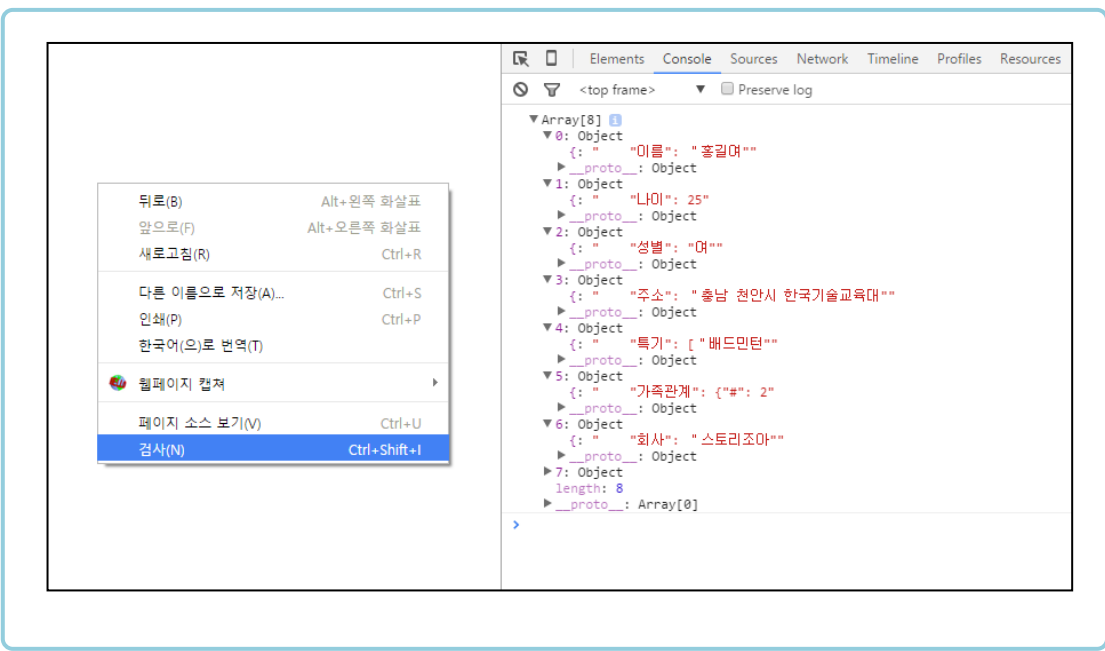
```
{
  "이름": "홍길여",
  "나이": 25,
  "성별": "여",
  "주소": "충남 천안시 한국기술교육대",
  "특기": ["배드민턴", "탁구"],
  "가족관계": {"#": 2, "아버지": "봉산", "소이": ""},
  "회사": "스토리조아"
}
```

● JSON를 활용한 데이터 시각화

1. JSON 개요

◆ JSON 불러오기

- 크롬브라우저에서 마우스 오른쪽 버튼을 눌러 검사를 선택함
- Console 메뉴 선택
- JSON 파일이 나타남



● JSON를 활용한 데이터 시각화

2 JSON 데이터 시각화 실습

◆ JSON 데이터 시각화 실습 순서

- ① CSS 코드 설계하기
- ② JavaScript 코드(JSON 데이터 입력하기)
- ③ SVG 스크립트 작성하기
- ④ JSON 시각화 결과 확인하기(상호작용 그래프)

1. CSV를 활용한 데이터 시각화

■ CSV 형식 파일 불러오기

- D3.js 에서 활용할 수 있는 데이터들, 단순 배열 데이터 TSV, CSV, JSON, HTML, XML, TEXT 파일을 불러냄

■ D3.js 외부 데이터 종류별 특징 요약

- TSV : 데이터가 탭으로 구분된 형식을 말하며 헤더가 있음
- CSV : 반점으로 구분한 텍스트, 헤더가 있음
- JSON : JavaScript object Notation의 약자로 간단하고 데이터 용량도 적으며, 헤더가 없음
- HTML : 브라우저로 데이터를 보기 좋게 표시해주며 레이아웃이 있음
- XML : 태그를 이용하여 데이터를 표현하며 범용적인 데이터 포맷이고, SVG도 XML의 종류 중 하나임
- TEXT : 기본 문자로 구성된 데이터 파일

■ CSV 최신기술 동향

- 데이터 시각화 갤러리로 다양한 종류의 그래프와 트리 차트를 예제를 통하여 볼 수 있음
- D3.js 갤러리로 많은 예제를 직접 소스를 통해 볼 수 있는 페이지가 있어 학습하기 좋은 곳임

■ CSV 한글 깨짐 방지

- CSV 파일을 로드 했을 때 한글이 깨지는 경우 CSV 파일을 utf-8 로 다시 저장함

2. SVG를 활용한 데이터 시각화

■ SVG 정의

- 웹용 벡터 그래픽으로 HTML 문서에 직접 포함시키거나 삽입할 수 있음
- 인터넷 익스플로러 8 이하 버전을 제외한 모든 브라우저에서 지원함
- 코드 작업으로 이미지 편집이 가능함
- 품질 손상 없이 확대 / 축소가 가능함

■ SVG 특징

- XML로 기술하므로 웹 브라우저 상에서 열람할 수 있고 문서 편집기 등에서 편집할 수 있음
- 하이퍼링크를 이미지에 걸거나, JavaScript 등과 연동시킬 수도 있음
- 벡터 그래픽이므로 확대나 축소를 해도 화질에는 변화가 없음

■ 브라우저 기반 SVG 확대 / 축소 및 이동

- SVG 그래픽을 확대 / 축소 및 이동하는 가장 간단한 방법은 브라우저의 기본 확대 / 축소 및 스크롤 기능을 사용함

■ SVG 좌표의 이해

- 그래프를 그릴 때 좌표는 일반적인 좌표와 HTML 기준의 좌표가 다름
- x와 y의 기준은 왼쪽 상단부터 시작함

3. JSON를 활용한 데이터 시각화

■ JSON 정의

- 속성-값 쌍으로 이루어진 데이터 오브젝트를 전달하기 위해 인간이 읽을 수 있는 텍스트를 사용하는 개방형 표준 포맷
- 비동기 브라우저 / 서버 통신 (AJAX)을 위해, 넓게는 XML(AJAX가 사용)을 대체하는 주요 데이터 포맷임
- 인터넷에서 자료를 주고 받을 때 그 자료를 표현하는 방법으로 알려져 있음
- 자료의 종류에 큰 제한은 없으며, 특히 컴퓨터 프로그램의 변수 값을 표현하는 데 적합함
- 본래는 JavaScript 언어로부터 파생되어 JavaScript의 구문 형식을 따르지만 언어 독립형 데이터 포맷임
- 즉, 프로그래밍 언어나 플랫폼에 독립적이므로, 구문 분석 및 JSON 데이터 생성을 위한 코드는 C, C++, C#, 자바, JavaScript, 펄, 파이썬 등 수많은 프로그래밍 언어에서 쉽게 이용할 수 있음
- JSON 포맷은 RFC 7159와 ECMA-404라는 두 개의 경쟁 표준에 의해 기술되고 있음

3. JSON를 활용한 데이터 시각화

■ JSON 장점

- JSON은 텍스트로 이루어져 있으므로, 사람과 기계 모두 읽고 쓰기 쉬움
- 프로그래밍 언어와 플랫폼에 독립적이므로, 서로 다른 시스템간에 객체를 교환하기에 좋음
- JavaScript의 문법을 채용했기 때문에, JavaScript에서 eval 명령으로 곧바로 사용할 수 있음
- JavaScript를 자주 사용하는 웹 환경에서 유리함
- 실질적으로 eval 명령을 사용하면 외부에서 악성 코드가 유입될 수 있음
- 모질라 파이어폭스 3.5, 인터넷 인터넷 익스플로러 8, 오페라 10.5, 사파리, 구글 크롬 등 대부분의 최신 웹 브라우저는 JSON 전용 파서 기능을 내장하고 있으므로 이런 기능을 사용하는 것이 더 안전할 뿐만 아니라 빠른 방법임