

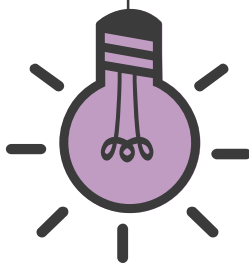
# Ajax

J a v a S c r i p t



01

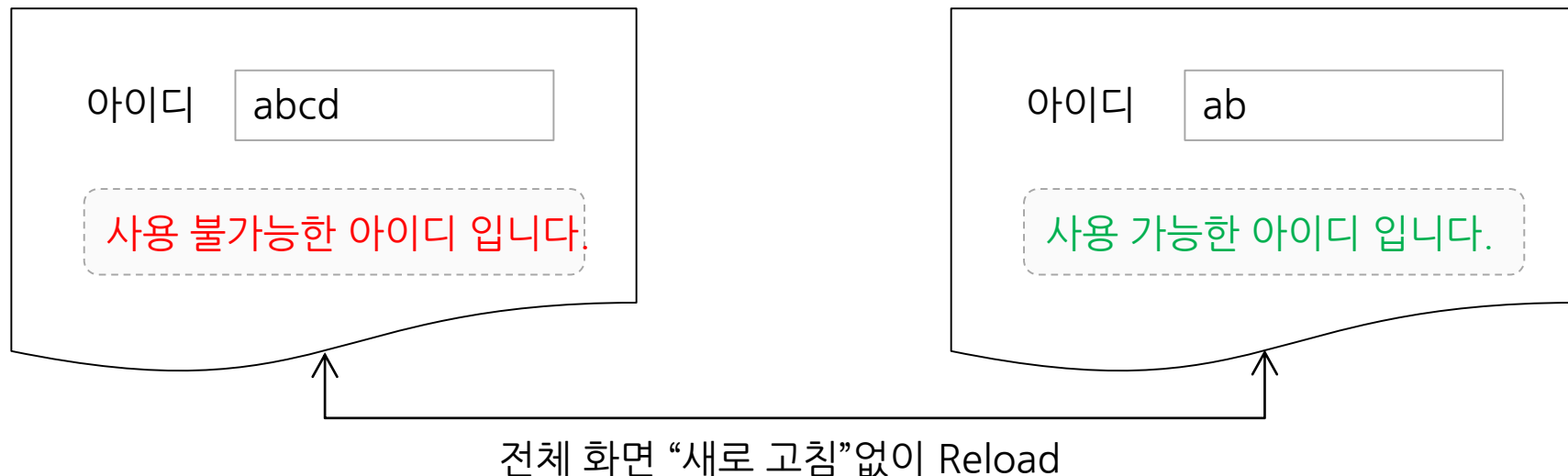
## Ajax의 소개



# 1. Ajax(Asynchronous JavaScript + XML) 란?

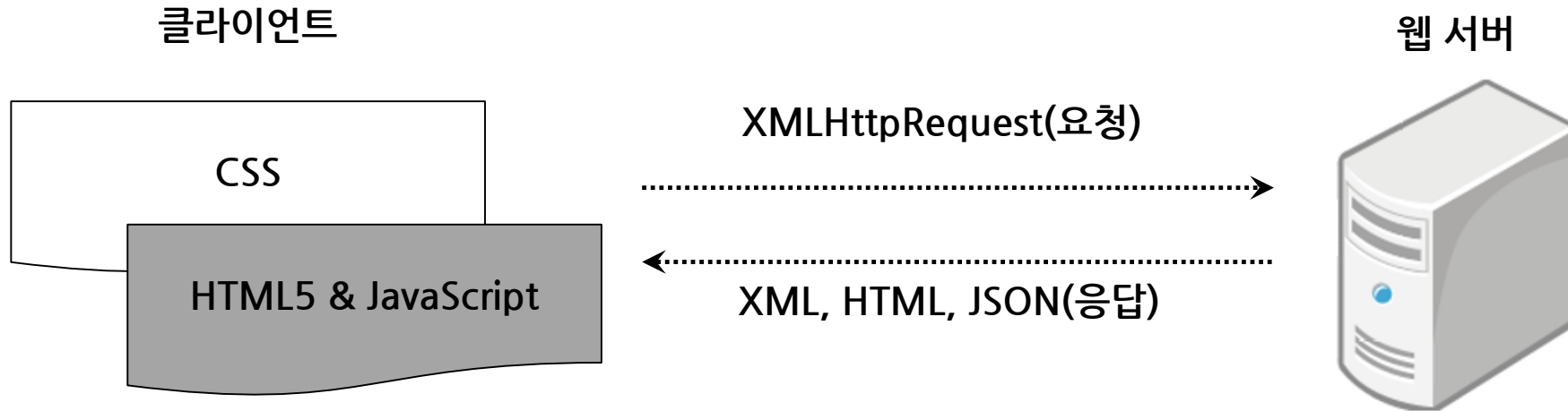
HTML, CSS, 자바스크립트, DOM, XML 등 기존에 사용되던 여러 기술을 함께 사용하는 새로운 개발 기법이다. Ajax는 웹 페이지 전체를 다시 로딩 하지 않고도, 웹 페이지의 일부분만을 갱신할 수 있게 해준다. 즉 Ajax를 이용하면 백그라운드 영역에서 서버와 통신하여, 그 결과를 웹 페이지의 일부분에만 표시할 수 있다.

사용 예)



## 2. Ajax 구동 방식

1. 자바스크립트 XMLHttpRequest 객체를 이용해서 서버에 데이터를 요청한다.
2. 웹 서버는 클라이언트가 요청한 데이터를 보여주기 위해 가공한다.
3. '새로 고침'을 할 필요가 없고 부분 부분만 로딩되므로 속도가 빠르다.  
(XML, HTML, JSON으로 응답)



## 3. Ajax의 4가지 구성요소

### 1. 자바스크립트

유저 이벤트 처리한다. 사용자가 마우스를 드래그 하거나 버튼을 클릭하면 XMLHttpRequest객체를 사용해서 웹 서버에 요청을 전송한다. 또한 XMLHttpRequest 객체로부터 응답이 오면 DOM, CSS등을 사용해서 화면을 조작한다.

### 2. XMLHttpRequest

서버와 데이터 교환할 때 웹 서버와 통신을 담당한다.

사용자의 요청을 웹 서버에 전달하고 웹 서버로부터 받은 결과를 웹 브라우저에 전달한다.

### 3. DOM(Document Object Model)

문서 정보의 구조를 나타낸다. 폼 등의 정보나 화면 구성을 조작할 때 사용한다.

### 4. CSS

글자 색, 배경색, 위치, 투명도 등 UI와 관련된 부분을 담당한다.

## XMLHttpRequest

- ① Ajax에서 서버와 클라이언트간에 통신을 담당하는 것으로 가장 중요한 객체이다.
- ② 서버의 응답결과가 HTML이 아닌 XML 또는 단순 텍스트이다.
- ③ 페이지 이동 없이 결과가 해당 화면에서 바로 반영한다.

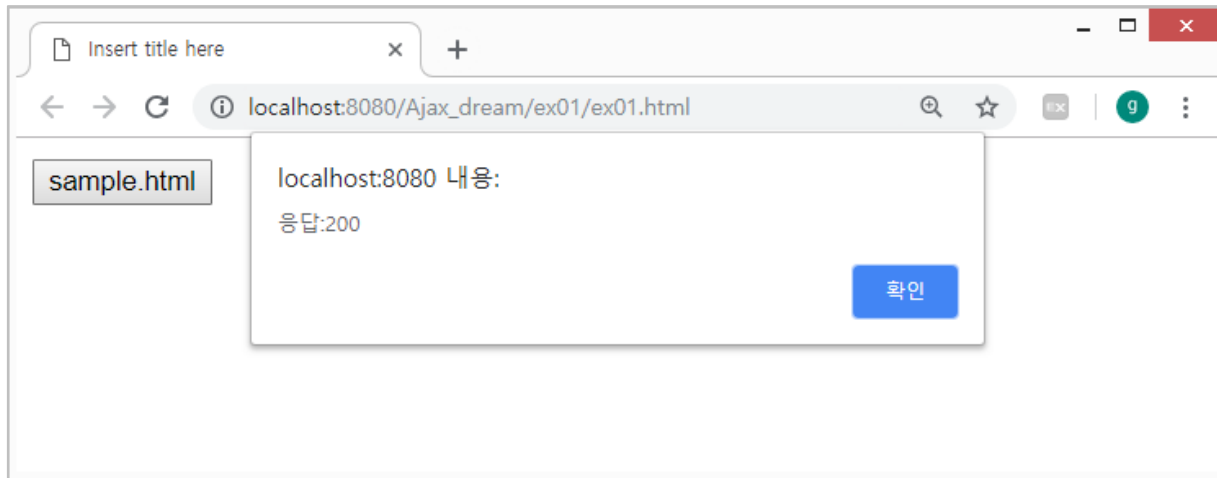
### (1) onreadystatechange 객체

상태의 변경이 발생했을 때 이벤트를 처리하기 위해 이벤트 핸들러를 기술한다.

객체 프로퍼티	설 명
requestText	문자열로 이루어진 서버의 응답을 받는다.
status	서버로부터 응답 받는 HTTP상태코드로 숫자를 리턴 한다. 200 : 정상(ok) 404 : 요청한 페이지를 찾을 수 없음(page not found) 500 : 서버에러(Internal Server Error) (대표적으로 2xx는 성공을, 4xx는 클라이언트 오류, 5xx는 서버의 오류를 나타냄)

객체 프로퍼티	설 명
readyStatus	<p>요청객체의 상태를 리턴 한다.</p> <p>0 : uninitialized - open() 메서드가 아직 호출되지 않은 상태이다.</p> <p>1 : open - send() 메서드가 아직 호출되지 않은 상태이다.</p> <p>2 : sent - send()메서드의 호출이 끝나서 헤더와 status의 사용이 가능한 상태이며 로드완료이다.</p> <p>3 : receiving - 일부 데이터를 받을 수 있는 상태로 처리 중이다.</p> <p>4 : loaded - 모든 데이터를 받을 수 있는 상태로 완료이다.</p>
requestXML	<p>XML로 이루어진 서버의 응답을 받는다.</p> <p>이 프로퍼티는 파싱이 가능한 W3C DOM노드 트리구조의 메서드와 프로퍼티를 XML 문서 객체로 리턴 받는다.</p>
statusText	<p>서버로부터 HTTP상태를 문자로 리턴 받는다.</p> <p>예를 들어, "Not Found", "OK" 같은 문자열로 리턴 한다.</p>

## ex01.html / sample.html





## (2) XMLHttpRequest를 이용한 웹 서버 요청기술

- open()함수 : 요청의 초기화로 접속할 URL입력한다. (get, post 선택)
- send() 함수 : 웹 서버에 요청한다.

ex) get 방식 전달

```
httpRequest = getXMLHttpRequest();
```

```
httpRequest.open("GET","text.txt",true);
```

```
httpRequest.open("GET","test.jsp?id=abcd&pwd=1234",true);
```

```
httpRequest.send(null);
```

세 개의인자: 1.get or post, 2.접속할URL, 3.동기(false)/비 동기(true) 방식 지정한다.

2.접속할 URL은 보안상 이유로 접속할 URL은 현재 페이지와 같은 도메인에 있어야 한다.

ex) post방식 전달

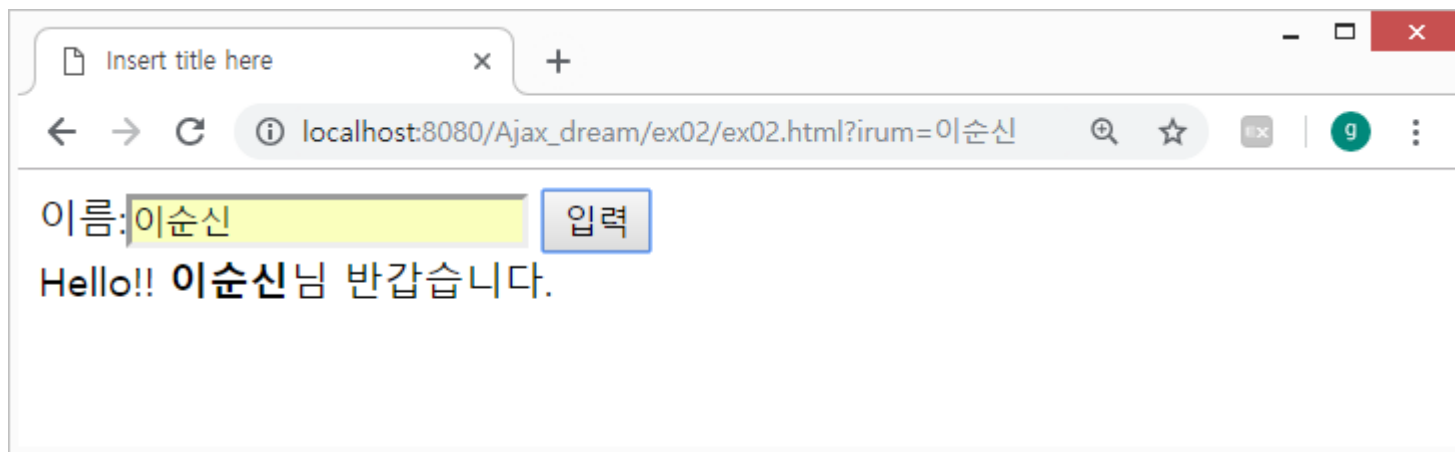
```
httpRequest = getXMLHttpRequest();
```

```
httpRequest.open("POST","text.jsp",true);
```

```
httpRequest.setRequestHeader("Content-Type","application/x-www-form-urlencoded")
```

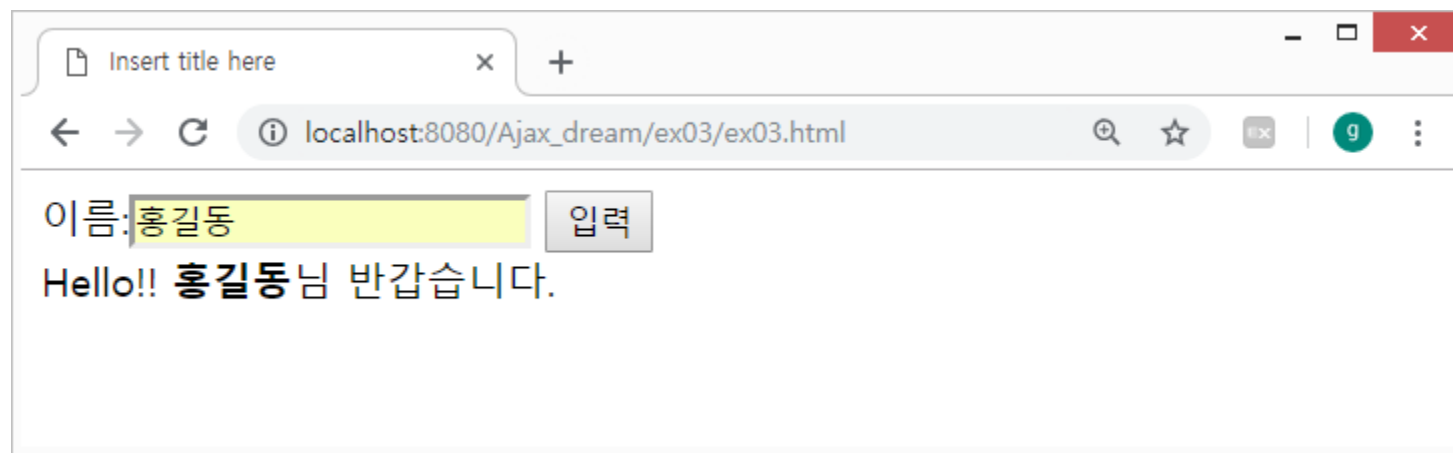
```
httpRequest.send("id=raeok&pwd=1234");
```

## GET방식 → ex02.html / hello.html



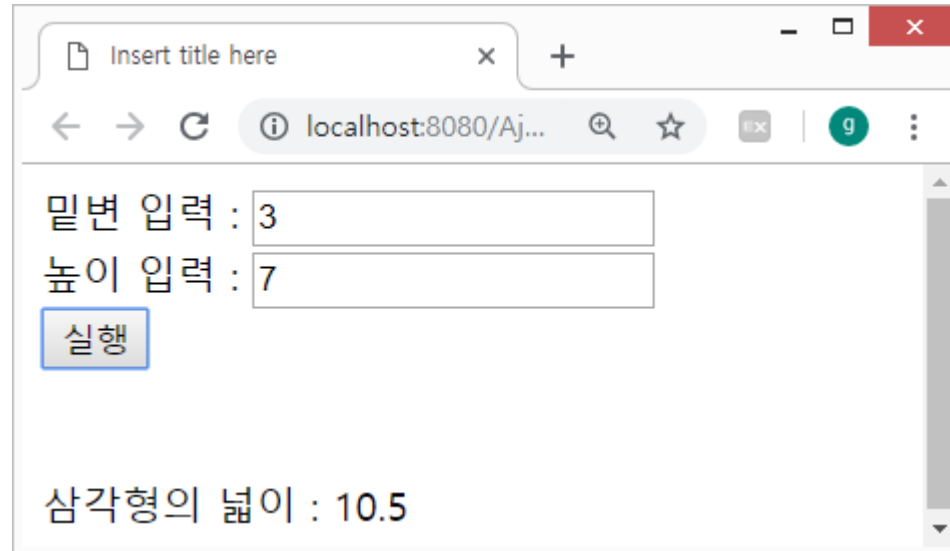
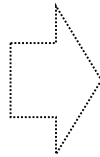
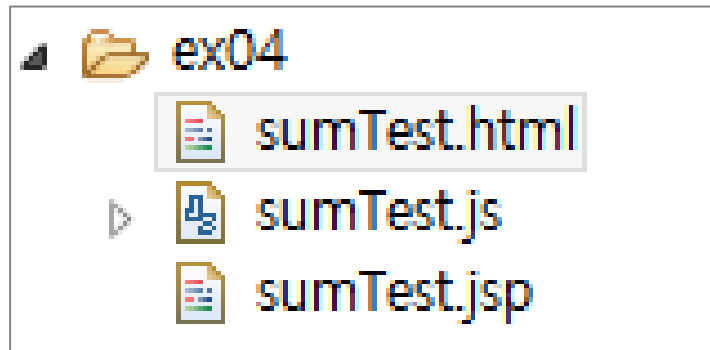
A screenshot of a web browser window. The address bar shows the URL `localhost:8080/Ajax_dream/ex02/ex02.html?irum=이순신`. The page content displays a form with the label "이름:" followed by a text input field containing "이순신" and a button labeled "입력". Below the form, the text "Hello!! 이순신님 반갑습니다." is shown.

## POST방식 → ex03.html / hello.html

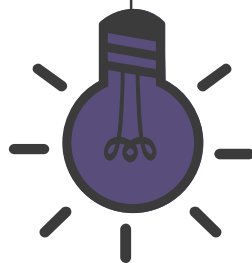


A screenshot of a web browser window. The address bar shows the URL `localhost:8080/Ajax_dream/ex03/ex03.html`. The page content displays a form with the label "이름:" followed by a text input field containing "홍길동" and a button labeled "입력". Below the form, the text "Hello!! 홍길동님 반갑습니다." is shown.

[문제] 밑변과 높이를 입력받아 삼각형의 넓이를 구하는 프로그램을 만들어 봅시다.



## 02 jQuery Ajax의 메서드



# 1. jQuery Ajax의 기본 함수

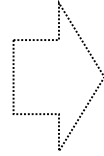
함수	설 명
<code>\$(selector).load()</code>	서버로부터 데이터를 받아서 요소 안에 HTML을 추가한다.
<code>\$.getScript();</code>	get방식 실행 후 Script데이터를 가져온다.
<code>\$(selector).serialize()</code>	form 태그 내의 name속성의 값을 모두 가지고 온다.
<code>\$(selector).each()</code>	선택한 요소가 여러 개 일 때 각각에 대해 반복하여 실행시킨다. 특정 조건을 만족할 때 반복에서 빠져 나오려면 return false를 사용한다.

## ex05.html / sample1.html / sample2.html

```
$('#demo').load("sample1.txt");
```

sample1.txt

반갑습니다



jQuery Ajax를 이용한 텍스트 불러오기

반갑습니다

텍스트

테이블

## ex05.html / sample1.html / sample2.html

```
$('#demo').load("sample2.txt");
```

### sample2.txt

```
<h2>*권장 도서*</h2>
```

```
<table border='1' width="300px">
```

```
<tr>
```

```
<th>과목</th><th>저자</th><th>가격</th>
```

```
</tr>
```

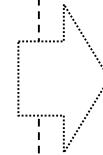
```
:
```

```
<tr>
```

```
<td>CSS3</td><td>제이슨</td><td>20,000</td>
```

```
</tr>
```

```
</table>
```



### jQuery Ajax를 이용한 텍스트 불러오기

#### \*권장 도서\*

과목	저자	가격
jQuery	주니어	15,000
HTML5	토미	25,000
CSS3	제이슨	20,000

텍스트

테이블



## ex06.html / sample3.html

```
$('#btn1').click(function(){  
    $('#demo').load("sample3.txt #p1");  
});  
$('#btn2').click(function(){  
    $('#demo').load("sample3.txt #p2");  
});
```

### sample3.txt

```
<p id="p1">good morning</p>  
<p id="p2">안녕하세요</p>
```



good morning

테스트1

테스트2

안녕하세요

테스트1

테스트2

## ex07.html / demo1.js

```
$.getScript("demo1.js");  
$("#submit").click(function(){  
    var irum=$('#user').val();  
    var sendData=setName(irum);  
    $("#result").html(sendData);
```

```
function setName(param) {  
    return "당신의 이름은 <b>" + param + "</b>입니다";  
}
```

이름을 입력하시오



이름을 입력하시오

당신의 이름은 **민들레**입니다

## ex08.html / demo2.js

```
$.getScript("demo2.js");  
$("#submit").click(function(){  
    var data1=$('#x').val();  
    var data2=$('#y').val();  
  
    var rs=addMethod(data1, data2);  
    $("#result").html(rs);  
});
```

```
function addMethod(data1, data2)  
{  
    parseInt("5")→5    parseInt("7")→7  
    var rs=parseInt(data1) + parseInt(data2);  
    return "두수의 합은 <b>" + rs + "</b>입니다";  
}
```



**\*\*두수의 합 구하기\*\***

x :  y :

두수의 합은 12입니다

## ex09.html

```
<form>
```

```
이름 : <input type="text" name="name"><br>
```

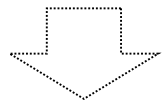
```
취미 : <input type="checkbox" name="hobby " value="climb">등산
```

```
      <input type="checkbox" name="hobby " value="fishing">낚시
```

```
      <input type="checkbox" name="hobby " value="travel">여행<br>
```

```
<input type="button" value="등록" id="show">
```

```
</form>
```



이름 :

취미 : ☒ 등산 ☒ 낚시 ☐ 여행

name=kim&hobby=climb&hobby=fishing

```
$("#show").click(function(){  
    var str=$("#form").serialize();  
    $("#result").text(str);  
});
```

## ex10.html

### CSS

```
<style type="text/css">
    .s1{color:red;}
    .s2{color:blue;}
    .s3{color:green;}
    .s4{color:brown;}
</style>
```

### jQuery Ajax

```
var i=1;
$('p').each(function(){
    $(this).addClass('s' + i);
    i++;
});
```



### HTML

```
<p>빨강</p>
<p>파랑</p>
<p>초록</p>
<p>갈색</p>
```

빨강  
파랑  
초록  
갈색

## 2. jQuery Ajax에서 서버와 연동하는 함수

[형식] \$.ajax({옵션속성:값});

```
$.ajax({  
  type: 'GET/POST',  
  url: url,  
  data: data  
  success: success,  
  dataType: dataType  
});
```

속 성	설 명
url	문자열 - 요청url 설정 ,데이터를 보내고 자하는 주소 ex) <form action="">
type	문자열 - GET/POST설정
[data]	요청 매개변수 설정
[datatype]	서버에서 내려온 data형식 ex) default : xml, json, script, text, html
success	성공 시 호출할 함수 설정
[error]	실패 시 호출할 함수 설정
async	불리언 - 동기/비동기 설정

## 2. jQuery Ajax에서 서버와 연동하는 함수

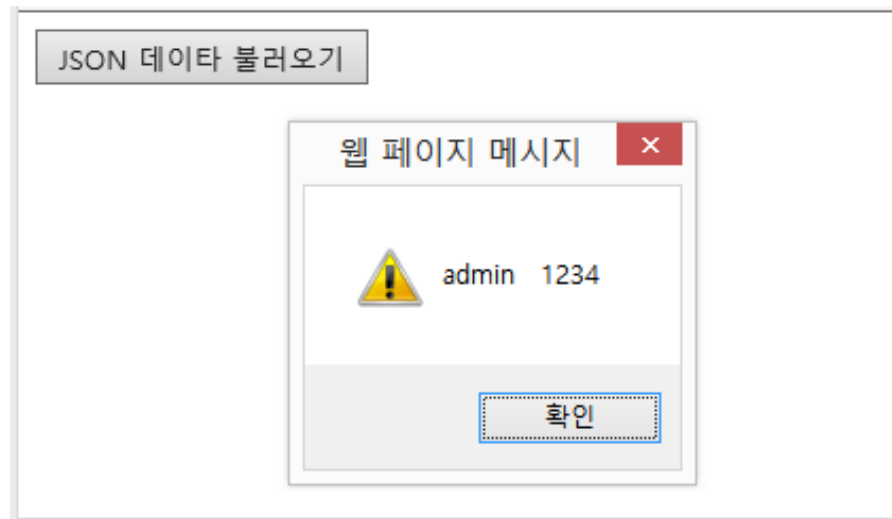
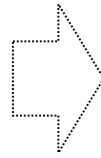
간편형 함수	설 명
<code>\$.get({옵션속성:값});</code>	get방식으로 Ajax를 실행 [형식] <code>\$.get(url, [data] ,[callback])</code>
<code>\$.post({옵션속성:값});</code>	post방식으로 Ajax를 실행 [형식] <code>\$.post(url, [data], [callback], [type])</code>
<code>\$.getJSON();</code>	get방식 실행 후 JSON데이터를 가져온다. [형식] <code>\$.getJSON (url, [data], [callback])</code> url : 호출할 파일(경로) data : url에 보낼 데이터 ex) id값, password값 callback : 처리 결과를 담당할 함수

## ex11.html / json1.txt

```
$('#btn').click(function(){  
    $.getJSON("json1.txt", function(data){  
        alert(data.id+ " " +data.pw);  
    });  
});
```

json1.txt

```
{"id":"admin","pw":"1234"}
```



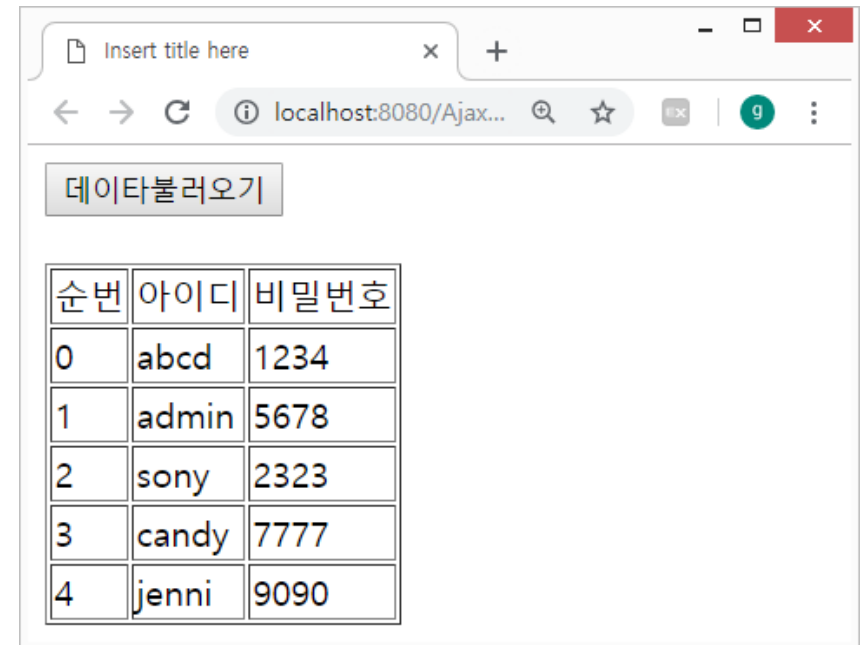
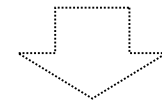
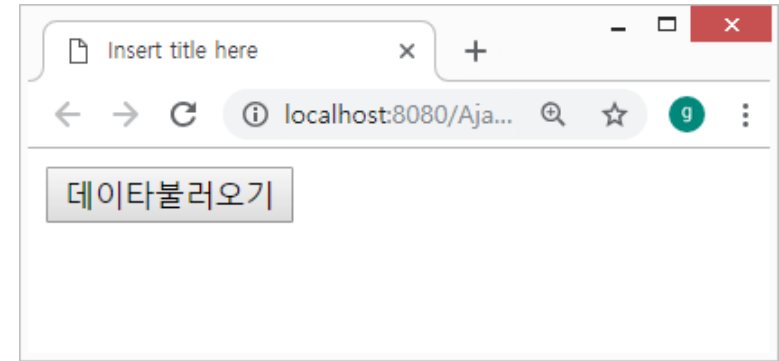


## ex12.html / json2.txt

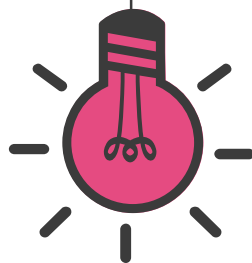
```
$('#btn').click(function(){
    $.getJSON("json2.txt",function(data){
        $("#display").empty();

        var table="<table border='1'><tr><td>순번</td>
                                <td>아이디</td>
                                <td>비밀번호</td>";

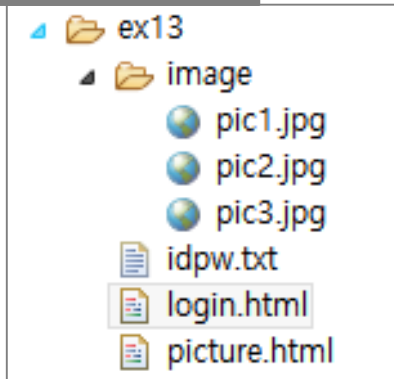
        //each()메서드를 이용해서 데이터 만큼 반복처리
        // entry.id 또는 entry["id"]으로 표현
        $.each(data, function(index, entry){
            table += "<tr>";
            table += "<td>" + index + "</td>";
            table += "<td>" + entry.id + "</td>";
            table += "<td>" + entry.pw + "</td></tr>";
        });
        table += "</table>";
        $('#display').append(table);
    });
});
```



## 03 jQuery Ajax 종합 실습



## 파일 설계도



## login.html / idpw.txt

(1) 계정과 패스워드를 입력한다

계정과 패스워드를 입력해 주세요

계정	<input type="text"/>	로그인
패스워드	<input type="password"/>	

아이디와 비밀번호를 입력해 주세요

계정과 패스워드 없이 로그인 할 때

계정과 패스워드를 입력해 주세요

계정  로그인

패스워드

id 또는 pw가 틀립니다

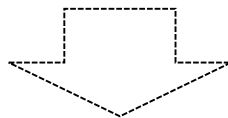
계정 또는 패스워드가 틀릴 때

계정과 패스워드를 입력해 주세요

계정  로그인

패스워드

계정과 패스워드가 맞았을 때



(2) 로그인 이 되면 이름과 보유 포인트를 출력한다

kim님 로그인 하셨습니다

현재 보유하신 포인트는 2500점 입니다

[사진첩으로 이동](#)

## picture.html

(3) "사진첩으로 이동" 을 클릭하면 3개의 이미지 보기 화면으로 이동합니다.

kim님 로그인 하셨습니다  
현재 보유하신 포인트는 2500점 입니다

[사진첩으로 이동](#)



---

## ❖ 웹 표준 검증 사이트

---

- W3C Mark Up 검증 서비스: <http://validator.kldp.org/>
- W3C CSS Validation 서비스: <http://jigsaw.w3.org/css-validator/>

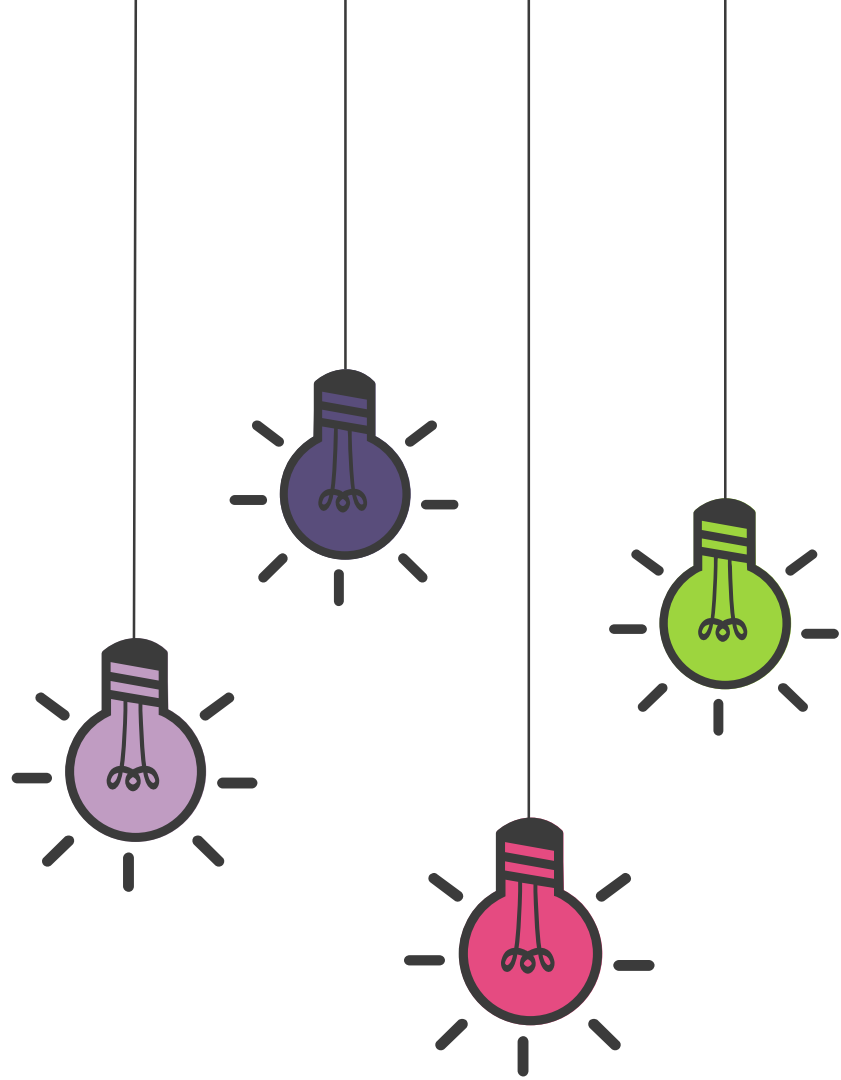


---

## ❖ 참고 사이트

---

- <http://www.w3schools.com>



# 감사합니다

THANK YOU FOR WATCHING



김 계 희  
jenni7@naver.com