



## jQuery Ajax

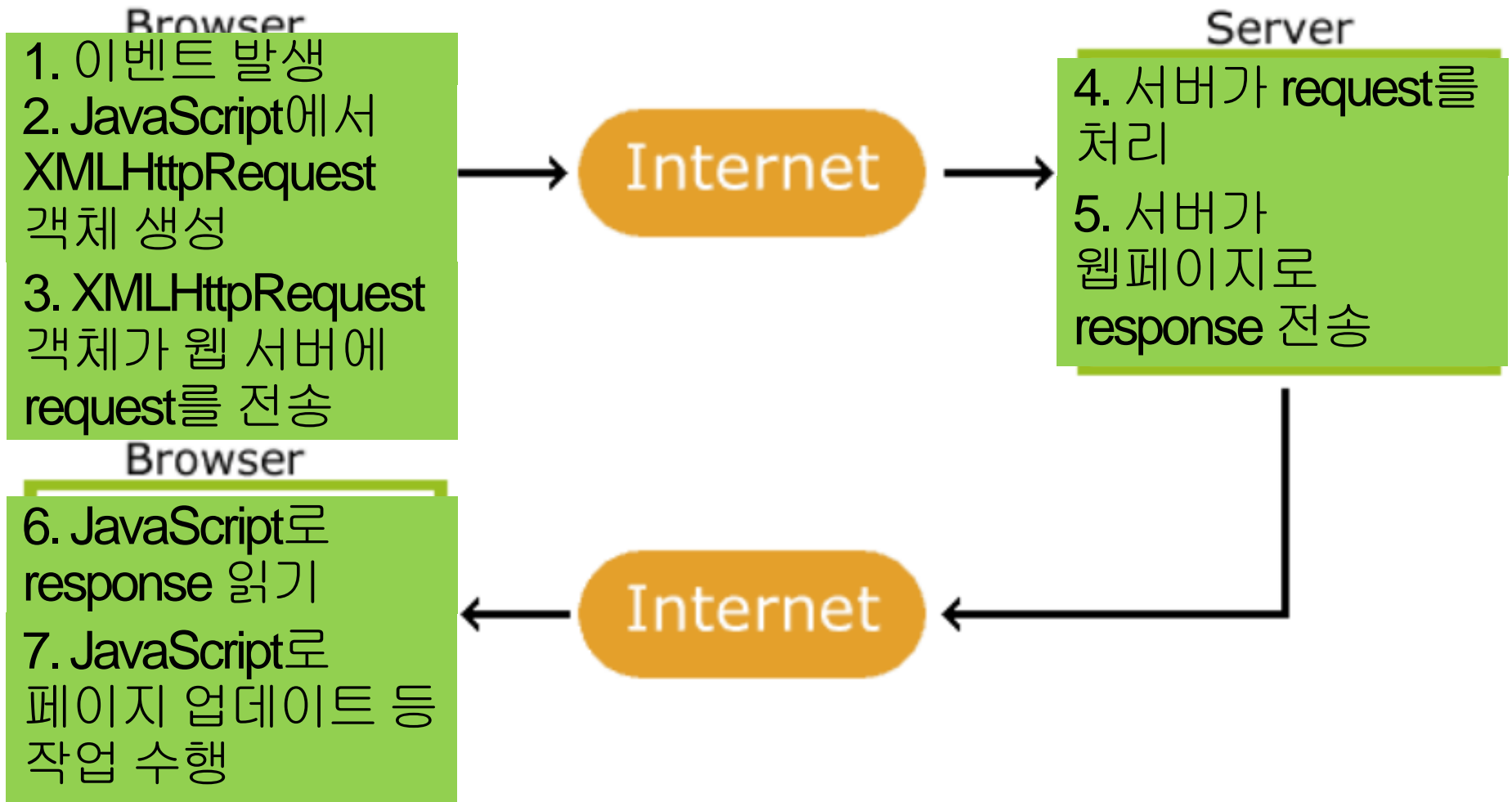
# AJAX 개요

---

## ◆ AJAX : Asynchronous JavaScript and XML

- 프로그래밍 언어가 아님
- 자바스크립트를 이용해서 비동기적으로 서버와 브라우저가 데이터를 주고 받는 방식을 의미
- 전체 페이지를 다시 로드 하지 않고 웹 페이지의 일부를 업데이트 할 수 있음
- 브라우저 기본 객체인 XMLHttpRequest를 이용해서 웹 서버로 데이터 요청

# AJAX의 동작 원리



# 서버 Request(1/3)

- ◆ XMLHttpRequest 객체는 전체 페이지를 다시 로드하지 않고 웹 페이지의 일부를 업데이트 할 수 있게 함

```
var xhttp = new XMLHttpRequest();
```

속성	설명
onreadystatechange	readyState 값이 변경될 때 호출할 함수 정의
readyState	0: 요청 준비 1: 서버 연결 완료 2: 요청 받음 3: 요청 처리 중 4: 요청 처리 완료 후 응답 준비
responseText	서버로부터 반환된 응답 정보(문자열)
responseXML	서버로부터 반환된 응답 정보(XML)
status	요청에 대한 상태 코드 200: "OK" 403: "Forbidden"    404: "Not Found"
statusText	요청에 대한 상태 문자

# 서버 Request(2/3)

메소드	설명
new XMLHttpRequest()	XMLHttpRequest 객체 생성
abort()	현재 요청 취소
getAllResponseHeaders()	header 정보 반환
getResponseHeader( <i>key</i> )	header 정보 중 특정 값 반환
	요청 설정
open( <i>method, url, async, user, psw</i> )	<i>method</i> . 요청 유형(GET/ POST) <i>url</i> . 호출할 URL <i>async</i> . 비동기 호출 여부(true/false)
send()	서버로 요청 전송(GET 방식)
send( <i>string</i> )	서버로 요청 전송(POST 방식)
setRequestHeader()	요청 header에 정보 설정

# 서버 Request(3/3)

---

```
var xhttp = new XMLHttpRequest();
```

## ❖ Get 방식 요청-----

```
xhttp.open("GET",  
            "ajax_test.jsp?name=홍길동&age=30", true);  
xhttp.send();
```

## ❖ Post 방식 요청-----

```
xhttp.open("POST", "ajax_test.jsp", true);  
xhttp.setRequestHeader(  
    "Content-type", "application/x-www-form-urlencoded");  
xhttp.send("name=홍길동&age=30");
```

# 서버 Response(1/2)

## ◆ XMLHttpRequest의 서버 Response 속성과 메소드

속성	설명
<code>responseText</code>	문자열 형식의 응답 정보 반환
<code>responseXML</code>	XML 형식의 응답 정보 반환

메소드	설명
<code>getResponseHeader(<i>key</i>)</code>	header 정보 중 지정된 정보만 반환
<code>getAllResponseHeaders()</code>	모든 header 정보를 반환

# 서버 Response(2/2)

---

```
var xhttp = new XMLHttpRequest();
```

❖ 응답 -----

```
xhttp.onreadystatechange = function() {  
    if(this.readyState == 4 && this.status == 200) {  
        res = this.responseText;  
        resjson = JSON.parse(res);  
    }  
}
```

```
xhttp.open("GET", "ajax_test.jsp", true);  
xhttp.send();
```



# jQuery AJAX 사용법(1/3)

---

```
$("#execute").click(function(){
    $.ajax({
        url: "./time2.jsp"
        , type: "POST"
        , data: $("form").serialize()
        , dataType: "json"
        , success: function(data){
            $("#time").text(data);
        }
        , error: function(xhr){
            console.log(xhr);
            alert("오류발생");
        }
    }
    ),
})
```

# jQuery AJAX 사용법(2/3)

- ◆ data : 서버로 데이터를 전송할 때 사용한다.
- ◆ dataType  
서버측에서 전송한 응답데이터를 어떤 형식의 데이터로 해석할 것인가를 지정한다. 값으로 올 수 있는 것은 xml, json, script, html, text 이다.
- ◆ success  
성공했을 때 호출할 콜 백을 지정한다.  
function( PlainObject result, String textStatus, jqXHR  
jqXHR
- ◆ type  
데이터를 **전송하는 방법을** 지정한다. ("GET" 또는 "POST")
- ◆ error  
실패 했을 때 호출할 콜백을 지정한다  
function(jqXHR xhr, String status, String error)

# jQuery AJAX 사용법(3/3)

---

## ◆ ajax의 단축메뉴

`$.get( url, data, success, dataType);`

`$.post( url, data, success, dataType);`

`$.getJSON( url, data, success)`

## ※ form submit 사용법

```
var fm = document.getElementById("myForm");
```

```
fm.action = "p"; // 호출할 url
```

```
fm.method = "post"; // 호출 방식. post 또는 get
```

```
fm.submit(); // form 전송
```

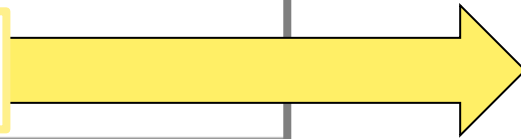
# 예제1(1/3)

이름:

나이:

post버튼 get버튼

Ajax전송



새로운 페이지로  
이동되어 응답  
받음

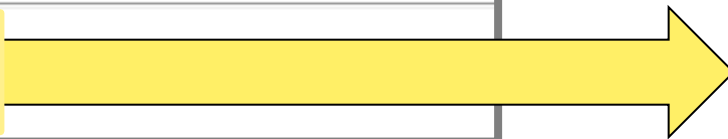
하이~

이름:

나이:

post버튼 get버튼

Ajax전송



페이지 이동없이,  
페이지 일부분에  
응답 결과를 받음

이름:

나이:

post버튼

Ajax전송

하이~

# 예제1(2/3)

## ◆ SUBMIT form 전송

<body>

<form id="myForm" action="../intro.html" method="get">

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

나이: <input type="text" name="age"><br><br>

<input type="button" onClick="fnSubmit()" value="post 버튼">

<input type="submit" value="get 버튼">

<hr>

<button type="button" id="sendAjax">Ajax 전송</button>

</form>

<span>

</body>

<script>

function myFunction() {

var fm = document.getElementById("myForm");

fm.method = "post";

fm.submit();

}

</script>

# 예제1(3/3)

---

## ◆ AJAX form 전송

```
$(document).ready(function() {  
    $("#sendAjax").click(function() {  
        $.ajax({  
            url : "../intro.html"  
            ,type : "GET"  
            ,success : function(data){  
                $('span').html(data);  
            }  
            ,error : function(jqXHR, textStatus,  
errorThrown){  
                alert("에러\n" + textStatus + " : " +  
errorThrown);  
            }  
        });  
    });  
});
```

# jQuery load 메서드(1/2)

◆ load() – 서버에서 데이터를 받아서 특정 요소에 넣는다.

```
$(selector).load(URL, data, callback);
```

- URL : 필수값. 로드 할 URL
- data : 서버로 전송할 data
- callback : load() 작업이 완료된 후 수행할 작업

```
function(response, status, xhr)
```

- response : 호출이 성공한 경우, 결과 콘텐츠를 포함  
(서버의 결과 메시지를 담아서 사용)
- status : 호출 상태
  - "success", "notmodified", "error", "timeout", "parsererror"
- xhr : XMLHttpRequest 객체

# jQuery load 메서드(2/2)

```
$(document).ready(function(){  
    $("tr").click(function(){  
        $("#divDetail").load("load_detail.html");  
    });  
});
```

...

```
<table>  
    <tr>  
        <td>1</td><td>홍길동</td><td>남자</td>  
    </tr>  
    <tr>  
        <td>2</td><td>임꺽정</td><td>남자</td>  
    </tr>  
</table>  
<hr>  
<div id="divDetail"></div>
```



# jQuery serialize 메서드

- ◆ Ajax를 이용해서 서버에 데이터를 보낼 때는 서버가 이해 할 수 있는 형식으로 바꿔야 함
- ◆ 데이터를 직렬화해서 하나의 객체로 만든 후 request에 담아 보냄
- ◆ jQuery에서 제공하는 직렬화 메소드는 serialize

---

serialize()	<code>\$("form").serialize();</code> → <code>FirstName=길동&amp;LastName=홍&amp;age=30</code>
-------------	---

serializeArray()	<code>\$("form").serializeArray();</code> → <code>[{name:"FirstName", value:"길동"},</code> <code>{name:"LastName", value:"홍"},</code> <code>{name:"age", value:"30"}]</code>
------------------	--

# jQuery ERROR

## ◆ 자주 발생 하는 error

- 404 에러 : 요청한 페이지/소스가 존재 하지 않을 때

## ◆ 에러처리(확인)

```
error : function(xhr, status, error){  
    alert("status :" + xhr.status  
    + "\r\ncode : " + xhr.statusText  
    + "\r\nmessage : " + xhr.responseText);  
}
```

localhost:9090 내용:

status :404

code : Not Found

message : <!doctype html><html lang="ko"><head><title>HTTP  
상태 404 – 찾을 수 없음</title><style type="text/css">body {font-  
family:Tahoma,Arial,sans-serif;} h1, h2, h3, b

(color:white;background-color:#F2FDF6;) h1 (font-size:22px) h2

# 연습1 - MemberList

## 멤버리스트

아이디	이름	메일	전화번호
a001	김은대	pyoedab@lycos.co.kr	011-621-4615
b001	이뿐이	engelcd@pretty.net	016-888-9999
c001	신용환	kyh01e@hanmail.net	011-123-5678
d001	성윤미	dbs81f@hanmail.net	019-222-8877
e001	이혜나	bosiang@hanmail.net	011-432-8901
f001	신영남	SUPER-KHG@HANMAIL.NET	011-253-2121
g001	송경희	lim052@hanmail.net	017-356-3578
h001	라준호	wingl7@hanmail.net	019-522-1679
i001	최지현	pan@orgio.net	017-614-6914
j001	김윤희	maxsys@hanmail.net	018-332-8976
k001	오철희	equus@orgio.net	016-157-8765
l001	구길동	email815@hanmail.co.kr	016-322-8865
m001	박지은	happy@hanmail.net	016-252-0675
n001	탁원재	ping75@unitel.co.kr	019-632-2176
o001	배인정	tar-song@hanmail.net	011-622-5971
p001	오성순	sahra235@intz.com	017-810-7658
q001	육평희	kph@hanmail.net	017-823-2359
r001	정은실	econie@hanmail.net	016-533-8768
s001	안은정	songej@hanmail.net	019-222-8155
t001	성원태	bob6@hanmail.net	011-272-8657
u001	김성욱	pss576@orgio.net	018-273-9056
v001	이진영	gagsong@orgio.net	017-240-8766
w001	김형모	songone@hanmail.net	011-345-9877
x001	진현경	happysong@hanmail.net	019-223-8767

# 연습2 - lprodList

## ◆ lprodList

Lprod\_ID : 1  
Lprod\_GU : P101  
Lprod\_NM : 컴퓨터제품

Lprod\_ID : 2  
Lprod\_GU : P102  
Lprod\_NM : 전자제품

Lprod\_ID : 3  
Lprod\_GU : P201  
Lprod\_NM : 여성캐주얼

Lprod\_ID : 4  
Lprod\_GU : P202  
Lprod\_NM : 남성캐주얼

Lprod\_ID : 5  
Lprod\_GU : P301  
Lprod\_NM : 피혁잡화

Lprod\_ID : 6  
Lprod\_GU : P302  
Lprod\_NM : 화장품

Lprod\_ID : 7  
Lprod\_GU : P401  
Lprod\_NM : 음반/CD

Lprod\_ID : 8  
Lprod\_GU : P402  
Lprod\_NM : 도서

Lprod\_ID : 9  
Lprod\_GU : P403  
Lprod\_NM : 문구류

### Lprod리스트

LPROD_ID	LPROD_GU	LPROD_NM
1	P101	컴퓨터제품
2	P102	전자제품
3	P201	여성캐주얼
4	P202	남성캐주얼
5	P301	피혁잡화
6	P302	화장품
7	P401	음반/CD
8	P402	도서
9	P403	문구류

# 연습3 - buyerInfo

## ◆ 거래처관리

거래처이름	거래처상세정보
삼성컴퓨터 삼보컴퓨터 현주컴퓨터 대우전자 삼성전자 대현 마르쥔 LG패션 캠브리지 가파치 한국화장품 피리어스 참존	

# 연습3 - buyerInfo

## ◆ 거래처관리

거래처이름	거래처상세정보	
삼성컴퓨터	BUYER_ID	P30201
삼보컴퓨터	BUYER_NAME	한국화장품
현주컴퓨터	BUYER_LGU	P302
대우전자	BUYER_COMTEL	051-212-5312
삼성전자	BUYER_BANK	외환은행
대현	BUYER_BANKNO	333-355-568898
마르쥔	BUYER_BANKNAME	박한국
LG패션	BUYER_MAIL	hangook@hangook.com
캠브리지	BUYER_ZIP	614-728
가파치	BUYER_ADD1	부산 부산진구 부전1동 동아빌딩
한국화장품	BUYER_ADD2	309호
피리어스		
참존		

# 연습4 - 제품분류리스트(lprod\_prod)

## ◆ 제품분류리스트

컴퓨터제품 ▼

모니터 삼성전자15인치칼라 ▼

항목	내용
PROD_ID	P101000001
PROD_NAME	모니터 삼성전자15인치칼라
PROD_LGU	P101
PROD_BUYER	P10101
PROD_COST	210000
PROD_PRICE	290000
PROD_SALE	230000
PROD_OUTLINE	평면모니터의 기적
PROD_DETAIL	우리기술의 개가