

# 221108 수업 -1

≡ 키워드	jQuery 비동기 함수
🕒 날짜	@2022년 11월 8일 오전 9:13

## jQuery AJAX 메소드

### jQuery AJAX Methods

AJAX is the art of exchanging data with a server, and update parts of a web page - without reloading the whole page. The following table lists all the jQuery AJAX methods: \$.ajaxPrefilter() Handle custom Ajax options or

 [https://www.w3schools.com/jquery/jquery\\_ref\\_ajax.asp](https://www.w3schools.com/jquery/jquery_ref_ajax.asp)



- ajax() 메서드

```
$("#button").click(function(){
    $.ajax({url: "demo_test.txt", success: function(result){
        $("#div1").html(result);
    }});
});
```

- getJSON() 메서드

```
$("#button").click(function(){
    $.getJSON("demo_ajax_json.js", function(result){
        $.each(result, function(i, field){
            $("#div").append(field + " ");
        });
    });
});
```

- load() Method

```
$("#button").click(function(){
    $("#div1").load("demo_test.txt");
});
```

원하는 3D 모션 생성 사이트

<https://loading.io/>

## 로딩 이미지

- 기본 틀

```
<script type="text/javascript">

    $(function(){
        $.ajax(
            {
                url: "Ex09_board_json.json",
                type: "POST", //httpReq.open("POST")
                dataType: "json", //서버가 클라이언트한테 응답 데이터 형식(text(json,html,txt,script), xml)
                //정상 처리될 경우
                success: function(data){

                },
                //에러일 경우 xmlhttp 객체를 받음
                error: function(xhr){

                },
                //readyState = 3, 서버에서 데이터가 오는 도중 다른것 클릭 못하도록
                beforeSend: function(){

                },
                //readyState = 4일 경우, 서버에서 데이터가 도착, 이미지 제거
                complete: function(){

                }
            }
        );
    });

</script>
```

- 예제

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>Insert title here</title>

    <style type="text/css">
        .wrap-loading {
            /*로딩 이미지*/
            position: fixed;
            top: 50%;
            left: 50%;
```

```

        margin-left: -21px;
        margin-top: -21px;
    }

    .display-none { /*감추기*/
        display: none;
    }
</style>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
<script type="text/javascript">

    $(function(){
        $.ajax(
            {
                url: "Ex09_board_json.json",
                type: "POST", //httpReq.open("POST")
                dataType: "json", //서버가 클라이언트한테 응답 데이터 형식(text(json,html,txt,script), xml)
                //정상 처리될 경우
                success: function(data){
                    console.log(data);
                    $.each(data, function(index, obj){
                        $('#treedata').append(obj.id + "/" + obj.name + "/" + obj.title + "/" + obj.content + "<br>");
                    });
                },
                //에러일 경우 xmlhttp 객체를 받음
                error: function(xhr){
                    console.log(xhr.status); //상태 번호(304,404,500)
                },
                //readyState = 3, 서버에서 데이터가 오는 도중 다른것 클릭 못하도록
                beforeSend: function(){
                    $('.wrap-loading').removeClass('display-none');
                },
                //readyState = 4일 경우, 서버에서 데이터가 도착, 이미지 제거
                complete: function(){
                    $('.wrap-loading').addClass('display-none');
                }
            }
        );
    });

</script>

</head>
<body>

    <!-- https://loading.io/ -->
    <div class="wrap-loading display-none">
        <div></div>
    </div>
    <table id="treedata" border="1">
    </table>

</body>
</html>

```

## json 파싱

- json으로 변환

### [Java] Gson 라이브러리 사용법 및 예제 ( Json 생성, 변환 )

이번에는 Gson 라이브러리로, Java에서 Json을 다루는 방법을 알아보도록 하겠습니다. Gson은 Gson 라이브러리 추가하기 Gson 객체 생성하기 Json 생성하기 Object -> Json 변환하기 Json -> Object 변환하기 Map -> Json 문자열 변환하기

☞ <https://hianna.tistory.com/629>



- Gson : <https://mvnrepository.com/artifact/com.google.code.gson/gson>
- Jackson : <https://mvnrepository.com/artifact/com.fasterxml.jackson.core/jackson-databind>
- JSON.simple : <https://mvnrepository.com/artifact/com.googlecode.json-simple/json-simple>

### LIB 제공

- <http://json-lib.sourceforge.net/index.html>
1. json-lib-2.4-jdk15.jar 필요
    - Lib 사용하기 위해서 아래 jar 파일 찾아서 다운로드



Json-lib requires (at least) the following dependencies in your classpath:

jakarta commons-lang 2.5  
jakarta commons-beanutils 1.8.0  
jakarta commons-collections 3.2.1  
jakarta commons-logging 1.1.1  
ezmorph 1.0.6

### POINT

- DTO 객체에 데이터를 담아야 JSON 변환 가능

### [서버쪽 코드]

<%

```
Class.forName("oracle.jdbc.OracleDriver");
Connection conn = null; //Connection 인터페이스 ...
conn = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","kosa","1004");

Statement stmt = conn.createStatement(); //명령 객체 얻어오기
String sql = "select empno , ename , sal , job ,comm from emp";
ResultSet rs = stmt.executeQuery(sql); //DB서버에서 실행되고 결과를 생성
```



```

</script>
</head>
<body>
  <h3>EMP LIST</h3>
  <input type="button" value="emplist" id="btn">
  <hr>
  <table id="list" border="1">

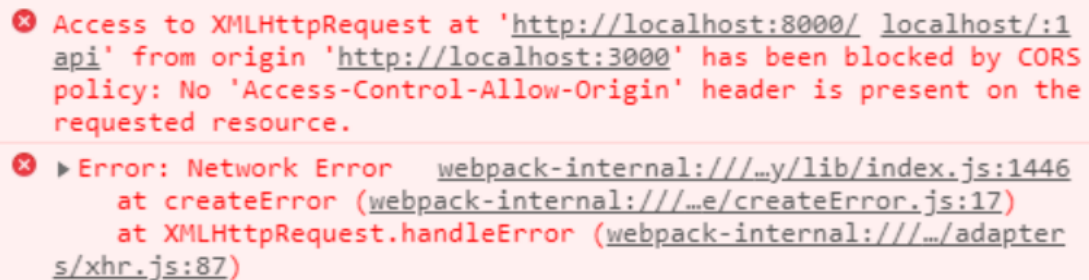
    </table>
</body>
</html>

```

- 서버 주소로 데이터 요청
- 위 쪽 서버랑 주소가 같아서 받아올 수 있음
  - 만약 다르면 크롬이 터트림
    - CORS 정책으로 인함

## CORS

- 우리가 가져오는 리소스들이 안전한지 검사하는 관문



```

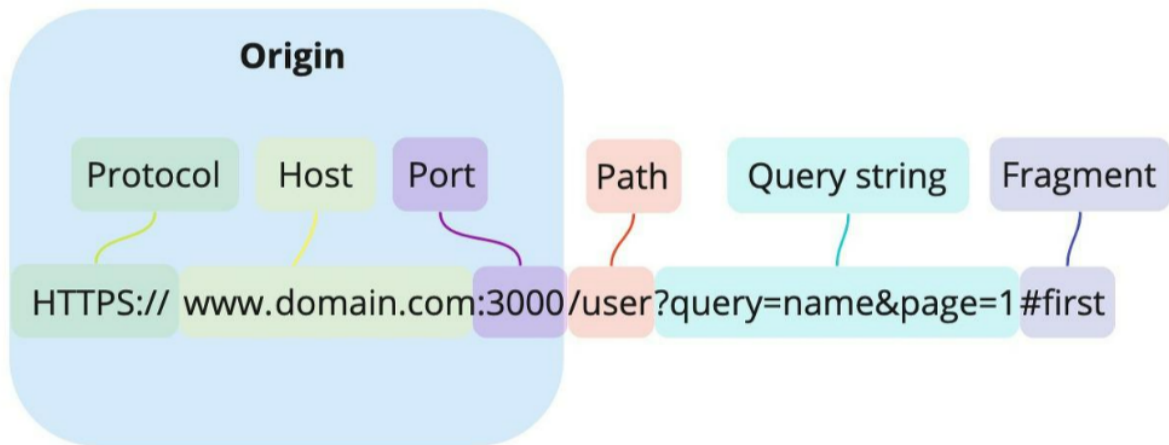
❌ Access to XMLHttpRequest at 'http://localhost:8000/ localhost/:1
api' from origin 'http://localhost:3000' has been blocked by CORS
policy: No 'Access-Control-Allow-Origin' header is present on the
requested resource.

❌ ▶ Error: Network Error  webpack-internal:///...y/lib/index.js:1446
   at createError (webpack-internal:///...e/createError.js:17)
   at XMLHttpRequest.handleError (webpack-internal:///.../adapter
s/xhr.js:87)

```

- HTML은 기본 Cross-Origin 정책을 따름
- XMLHttpRequest, Fetch API 등 script (비동기)는 Same-Origin 정책을 따름
  - 자바스크립트는 서로 다른 도메인에 대한 요청을 보안상 제한
  - Same-Origin-Policy 정책

origin 구분



### Same Origin 기준

URL	같은 출처	이유
https://github.com/about(기준)	O	프로토콜, 호스트, 포트가 동일
https://github.com/about?q=basic	O	프로토콜, 호스트, 포트가 동일
http://github.com	X	프로토콜이 다름
https://lienkooky.github.com	X	호스트가 다름

### 서버에서 Access-Control-Allow-Origin 세팅하기

```
//^ CORS 허용
response.setHeader("Access-Control-Allow-Origin", "*");
```

### 주의 사항

- JSONArray가 java.sql.Data를 net.sf.json.JSONException에 할당함 -> 오류 터트림
- Date > to\_char() 형식으로 변환해야 함

### 예제

[Server.jsp]

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%

    //객체를 json 객체로 변환하는 작업
    //1. JSONObject 사용해 1건의 데이터 변환 -> 객체 하나 생성
    //2. JSONArray 사용해 객체 배열 생성

    Member member = new Member();

    JSONObject objmember = JSONObject.fromObject(member);
    //{"address":"서울시 강남구","admin":"1","password":"1004","username":"kosa"}
    %>

    <%=objmember%>
    <hr>

    <%
        List<Member> memberlist = new ArrayList<>();
        memberlist.add(new Member("hong", "1004", "서울시 강남구", "0"));
        memberlist.add(new Member("kim", "1004", "서울시 용산구", "1"));
        memberlist.add(new Member("park", "1004", "서울시 종로구", "0"));

        JSONArray memberarray = JSONArray.fromObject(memberlist);

    %>
    <%=memberarray%>

```

## postman

- view쪽 코드 작성하기 전에 서버가 제대로 돌아가는지 확인



Explore

Search Postman

Invite

Upgrade

Import

Overview

GET http://127.0.0.1:8090/W

+

...

No Environment

...

http://127.0.0.1:8090/WebBasic\_JavaScript\_Jquery\_Ajax/Ex12\_jsonlib\_Server.jsp

Save

GET

http://127.0.0.1:8090/WebBasic\_JavaScript\_Jquery\_Ajax/Ex12\_jsonlib\_Server.jsp

Send

Params

Authorization

Headers (7)

Body

Pre-request Script

Tests

Settings

Cookies

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
Key	Value	Description		

Requests

Tests

...

Body

Cookies (1)

Headers (7)

Test Results

200 OK

1400 ms

1.31 KB

Save Response

Pretty

Raw

Preview

Visualize

HTML

```
1 [{"comm":0,"empno":7369,"ename":"SMITH","job":"CLERK","sal":800},{ "comm":300,"empno":7499,"ename":"ALLEN",
"job":"SALESMAN","sal":1600},{ "comm":200,"empno":7521,"ename":"WARD","job":"SALESMAN","sal":1250},
{ "comm":30,"empno":7566,"ename":"JONES","job":"MANAGER","sal":2975},{ "comm":300,"empno":7654,
"ename":"MARTIN","job":"SALESMAN","sal":1250},{ "comm":0,"empno":7698,"ename":"BLAKE","job":"MANAGER",
"sal":2850},{ "comm":0,"empno":7782,"ename":"CLARK","job":"MANAGER","sal":2450},{ "comm":0,"empno":7788,
"ename":"SCOTT","job":"ANALYST","sal":3000},{ "comm":3500,"empno":7839,"ename":"KING","job":"PRESIDENT",
"sal":5000},{ "comm":0,"empno":7844,"ename":"TURNER","job":"SALESMAN","sal":1500},{ "comm":0,"empno":7876,
"ename":"ADAMS","job":"CLERK","sal":1100},{ "comm":0,"empno":7900,"ename":"JAMES","job":"CLERK",
"sal":950},{ "comm":0,"empno":7902,"ename":"FORD","job":"ANALYST","sal":3000},{ "comm":0,"empno":7934,
"ename":"MILLER","job":"CLERK","sal":1300},{ "comm":100,"empno":100,"ename":"홍길동","job":"SALESMAN",
"sal":800}]
```

Cookies

Capture requests

Bootcamp

Runner

Trash