1. **Ajax : 서버와 데이터 교환을 하는 기술.**

<http://api.jquery.com/category/ajax/> ajax관련된 API 정보

웹페이지의 특정부분(한파트)를 업데이트하고 서버와 데이터를 교환하는 기술.

우리가 기존의 웹페이지의 내용을 변경하기 위해서는 웹페이지에 떠있는 페이지를 다시 로딩해야 했었다.

그런데 Ajax를 이용하게 되면 특정부분의 영역에 있는 데이터 수정시 전체 페이지를 로딩하지 않고 서버와 데이터 교환이 가능하다

샘플을 보면 기존에 있었던 텍스트 대신 바꿔 치기 한다.

전체가 로딩되지 않고 일부분만 텍스트가 변경되었다.

Ajax는 웹페이지의 전체내용을 다시 불러들이지 않고 웹페이지의 일부분을 배경에서 서버와 데이터 교환이 일어나 원하는 부분에 뿌려 줄 수 있는 기술이다.

jQuery에서는 Ajax와 관련하여 몇 개의 메소드를 제공하고 있다.

jQuery Ajax메소드를 사용하게 되면 외부데이터를 로딩해서 보여질 수 있다.

jQuery를 이용하지 않으면 Ajax이용이 어려울 수 있으나 jQuery를 이용하면 쉬운 코딩이 될 수 있다.

1. Http Request : GET vs. POST

$.get(URL, callback);

$.post(URL, {파라미터이름:전달파라미터값, callback);

$.ajax({/\* $.ajax(json형식으로 option 설정)

ex) $.ajax({url:'', type:'get', data:~ , success:function(data,status){} });

1. url : 요청 자원(서버)를 호출, 일반적으로 호출된 자원에서 json형식으로 데이터 return 해준다
2. type : 요청처리방식
3. data: 요청시, 함께 전달되는 데이터

ex. $(‘form’).serialize()

form안에 있는 데이터를 query로 변경하여 전달

1. dataType : 요청 후, 응답의 결과로 반환되는 데이터의 종류를 선언

ex. Xml, html, joson, text

1. success : 성공했을 때, 처리할 내용, 콜백함수로 나타냄.

ex. Function(data, status){}

1. error : 응답이 에러가 났을 때, 에러 상태에 대한 코드를 반환

ex. error : function(code){

code:에러에 대한 code 내용 전달...

}

\*/})

**MemberDao.java**

**public** **class** MemberDao {

**public** **static** **final** **int** ***EXISTENT*** = 0;

**public** **static** **final** **int** ***NONEXISTENT*** = 1;

**private** **static** MemberDao *instance* = **new** MemberDao();

**public** **static** MemberDao getInstance() {

**return** *instance*;

}

**private** MemberDao() {}

**private** Connection getConnection() **throws** SQLException {

Connection conn = **null**;

**try** {

Context ctx = **new** InitialContext();

DataSource ds = (DataSource) ctx.lookup("java:comp/env/jdbc/Oracle11g");

conn = ds.getConnection();

} **catch** (NamingException e) {

System.***out***.println(e.getMessage());

}

**return** conn;

}

// mId 중복체크

**public** **int** mIdConfirm(String mId) {

**int** result = ***EXISTENT***;

Connection conn = **null**;

PreparedStatement pstmt = **null**;

ResultSet rs = **null**;

String sql = "SELECT \* FROM MVC\_MEMBER WHERE mId=?";

**try** {

conn = getConnection();

pstmt = conn.prepareStatement(sql);

pstmt.setString(1, mId);

rs = pstmt.executeQuery();

**if**(rs.next()) {

result = ***EXISTENT***;

}**else** {

result = ***NONEXISTENT***;

}

} **catch** (Exception e) {

System.***out***.println(e.getMessage());

}**finally** {

**try** {

**if**(rs != **null**) rs.close();

**if**(pstmt != **null**) pstmt.close();

**if**(conn != **null**) conn.close();

} **catch** (Exception e) {

System.***out***.println(e.getMessage());

}

}

**return** result;

}

}

**idConfirm.jsp**

<%@page import=*"com.tj.ex.dao.MemberDao"*%>

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

pageEncoding=*"UTF-8"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

</head>

<body>

<%

String mId = request.getParameter("mId");

MemberDao mDao = MemberDao.getInstance();

**int** result = mDao.mIdConfirm(mId);

**if**(result==MemberDao.EXISTENT){

out.print("중복된 ID입니다");

}**else**{

out.print("사용 가능한 ID입니다");

}

%>

</body>

</html>

**ex01\_get.html**

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<script src=*"https://code.jquery.com/jquery-3.4.1.min.js"*></script>

<script>

$(document).ready(**function**(){

$('button').click(**function**(){

**var** mId = $('#mId').val();

$.get('idConfirm.jsp?mId='+mId, **function**(data, status){

**if**(status == 'success'){

$('#result').html(data);

}

});

});

$('#mId').keyup(**function**(){

**var** mId = $('#mId').val();

$.get('idConfirm.jsp?mId='+mId, **function**(data, status){

**if**(status == 'success'){

$('#result').html(data);

}

});

});

});

</script>

</head>

<body>

<p>아이디 <input type=*"text"* name=*"mId"* id=*"mId"*>

<button>중복체크</button></p>

<span id=*"result"*></span>

</body>

</html>

**ex02\_Post.html**

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<title>Insert title here</title>

<style></style>

<script src=*"https://code.jquery.com/jquery-3.4.1.min.js"*></script>

<script>

$(document).ready(**function**(){

$('button').click(**function**(){

**var** mId = $('#mId').val();

$.post('idConfirm.jsp',{mId:mId}, **function**(data, status){

$('#result').html(data);

});

});

});

</script>

</head>

<body>

<p>아이디 <input type=*"text"* name=*"mId"* id=*"mId"*>

<button>중복체크</button></p>

<span id=*"result"*></span>

</body>

</html>

**ex03\_ajax.html**

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<title>Insert title here</title>

<style></style>

<script src=*"https://code.jquery.com/jquery-3.4.1.min.js"*></script>

<script>

$(document).ready(**function**(){

$('button').click(**function**(){

**var** mId = $('#mId').val();

$.ajax({

// url : 요청경로

// type : get방식 / post 방식

// data : 요청 파라미터와 파라미터값

// dataType : html/json/... 요청경로로 실행한 결과의 타입

// success : 요청경로로 실행한 응답이 성공하였을 때 수행할 콜백함수

// error : 요청경로로 실행한 응답이 실패되었을 때 수행할 콜백함수

url : 'idConfirm.jsp',

type : 'get',

data : 'mId='+mId,

dataType : 'html',

success : **function**(data){

$('#result').html(data);

},

error : **function**(code){

alert(code.status);

}

});

});

});

</script>

</head>

<body>

<p>아이디 <input type=*"text"* name=*"mId"* id=*"mId"*>

<button>중복체크</button></p>

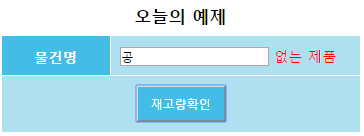
<span id=*"result"*></span>

</body>

</html>

**<오늘의 예제>**





**DROP** **TABLE** PRODUCT **CASCADE** CONSTRAINTS;

**CREATE** **TABLE** PRODUCT (

PNO NUMBER(4) **PRIMARY** **KEY**, -- 코드

PNAME VARCHAR2(100) **NOT** **NULL**, -- 물건명

PSTOCK NUMBER(4) **NOT** **NULL** -- 재고량

);

**INSERT** **INTO** PRODUCT **VALUES** (1, '공책', 3);

**INSERT** **INTO** PRODUCT **VALUES** (2, '볼펜', 10);

**INSERT** **INTO** PRODUCT **VALUES** (3, '지우개', 13);

**SELECT** \* **FROM** PRODUCT **WHERE** PNAME='공책'; -- DAO QUERY

**commit**;

**public** **class** **ProductDto** {

**private** **int** pno;

**private** String pname;

**private** **int** pstock;

**public** ProductDto() {}

**public** ProductDto(**int** pno, String pname, **int** pstock) {

**this**.pno = pno;

**this**.pname = pname;

**this**.pstock = pstock;

}

**public** **int** getPno() {**return** pno;}

**public** **void** setPno(**int** pno) {**this**.pno = pno;}

**public** String getPname() {**return** pname;}

**public** **void** setPname(String pname) {**this**.pname = pname;}

**public** **int** getPstock() {**return** pstock;}

**public** **void** setPstock(**int** pstock) {**this**.pstock = pstock;}

@Override

**public** String toString() {

**return** "ProductDto [pno=" + pno + ", pname=" + pname + ", pstock=" + pstock + "]";

}

}

**public** **class** **ProductDao** {

**private** **static** ProductDao *instance* = **new** ProductDao();

**public** **static** ProductDao getInstance() {

**return** *instance*;

}

**private** ProductDao() {};

**private** Connection getConnection() **throws** SQLException {

Connection conn = **null**;

**try** {

Context ctx = **new** InitialContext();

DataSource ds = (DataSource) ctx.lookup("java:comp/env/jdbc/Oracle11g");

conn = ds.getConnection();

} **catch** (NamingException e) {

System.***out***.println(e.getMessage());

}

**return** conn;

}

//SELECT \* FROM PRODUCT WHERE PNAME='공책'

**public** ProductDto getProduct(String pname) {

ProductDto product = **null**;

Connection conn = **null**;

PreparedStatement pstmt = **null**;

ResultSet rs = **null**;

String sql = "SELECT \* FROM PRODUCT WHERE PNAME=?";

**try** {

conn = getConnection();

pstmt = conn.prepareStatement(sql);

pstmt.setString(1, pname);

rs = pstmt.executeQuery();

**if**(rs.next()) {

product = **new** ProductDto();

product.setPno(rs.getInt("pno"));

product.setPname(pname);

product.setPstock(rs.getInt("pstock"));

}

} **catch** (Exception e) {

System.***out***.println(e.getMessage());

}**finally** {

**try** {

**if**(rs !=**null**) rs.close();

**if**(pstmt!=**null**) pstmt.close();

**if**(conn !=**null**) conn.close();

} **catch** (Exception e) {

System.***out***.println(e.getMessage());

}

}

**return** product;

}

}

**productStock.jsp**

<%@page import=*"com.tj.ex.dto.ProductDto, com.tj.ex.dao.ProductDao"*%>

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

pageEncoding=*"UTF-8"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

</head>

<body>

<%

String pname = request.getParameter("pname");

ProductDao pDao = ProductDao.getInstance();

ProductDto product = pDao.getProduct(pname);

**if**(product==**null**){

out.print("미취급상품");

}**else**{

out.print(product.getPstock()+"개 재고");

}

%>

</body>

</html>

**product.html**

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<title>Insert title here</title>

<style>

**table** {width:*400px*; margin: *0 auto*;}

**table** **th,** **input**[type=button] { width:*100px*; background-color: *#43BDE6*; color:*white*; padding: *10px*; text-align: *center*;}

**table** **td** { background-color: *#AFE0F0*; padding: *10px*; text-align: *center*; }

**table** **caption**{padding: *5px 5px 10px*; font-size: *1.2em*; font-weight: *bold*;}

**input**[type=text]*:FOCUS* { background-color: *lightyellow*; outline: *0*;}

*#stockCnt*{color:*red*;}

</style>

<script src=*"https://code.jquery.com/jquery-3.4.1.min.js"*></script>

<script>

$(document).ready(**function**(){

});

</script>

</head>

<body>

<table>

<caption>오늘의 예제1</caption>

<tr><th>물건명</th>

<td style="text-align: *left*;"><input type=*"text"* name=*"pname"*>

<span id=*"stockCnt"*> &nbsp; </span></td>

</tr>

<tr><td colspan=*"2"* style="text-align: *center*;">

<button id=*"confirm"*>재고량확인</button> <button id=*"alert"*>alert</button>

</td>

</tr>

</table>

</body>

</html>

**< MVC패턴 구조에서의 aJax >**

**위의 model1을 mvc패턴으로 구현**