-기존방식(동기화 방식)

[주소

[작성] -> 폼실행 -> 컨트롤러 -> 목록주소이동

ㄴ> jsp(history.back)] -> 작성

-비동기

주소+데이터

[작성] -> 컨트롤러

| <-

| 결과

|

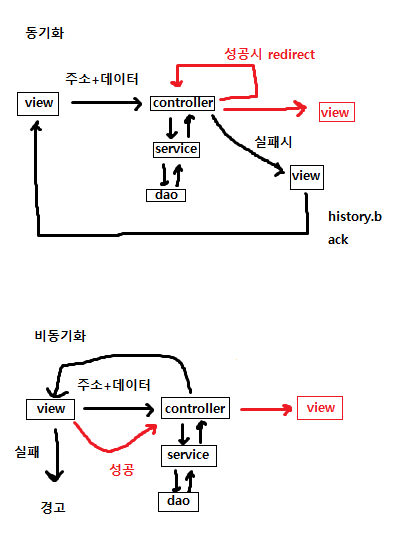
ㄴ> 목록이동

|

ㄴ> 경고

동기화 - 주소 바뀜(주소에 맞게 화면이 바뀜) / 폼, 로케이션을 통해 이동(화면 전체적으로 변경이 일어남)

비동기화 - 주소변경 없이 백그라운드에서 데이터만 주고 받음 / 부분적 변화 줄 때 용이(네이버 메인페이지 생각)



|  |
| --- |
| // RequestMapping의 method : 전송방식  // 현재는 post일때만 작동하겠다는 의미  // 옵션들은 추가적인 효과를 제공  // produces : 결과의 형태 지정  @RequestMapping(value = "/bWriteAjax", method = RequestMethod.POST, produces = "text/json;charset=UTF-8")  // ResponseBody : View로 인식. 돌려주는 결과가 View의 내용으로 인식됨.  @ResponseBody  public String bWriteAjax(@RequestParam HashMap<String,String> params, ModelAndView modelAndView) throws Throwable {  // ObjectMapper : jackson에서 제공. 객체정보를 문자열로 변환.(JSON 형태)  ObjectMapper mapper = new ObjectMapper();  Map<String, Object> modelMap = new HashMap<String, Object>();    try {  iTestService.bWrite(params);    modelMap.put("result", "success");  } catch (Throwable e) {  e.printStackTrace();  modelMap.put("result", "exception");  }  System.out.println(mapper.writeValueAsString(modelMap));  //writeValueAsString : 값을 문자열로 변환(JSON 형태)  return mapper.writeValueAsString(modelMap);    }      @RequestMapping(value = "/bListAjax", method = RequestMethod.POST, produces = "text/json;charset=UTF-8")  @ResponseBody  public String bListAjax(@RequestParam HashMap<String,String> params) throws Throwable {  ObjectMapper mapper = new ObjectMapper();  Map<String, Object> modelMap = new HashMap<String, Object>();    try {    int cnt = iTestService.getBCnt(params);    PagingBean pb = iPagingService.getPagingBean(Integer.parseInt(params.get("page")), cnt, 10, 5);    params.put("startCnt", Integer.toString(pb.getStartCount()));  params.put("endCnt", Integer.toString(pb.getEndCount()));    List<HashMap<String,String>> list = iTestService.getBList(params);    modelMap.put("list", list);  modelMap.put("pb", pb);  modelMap.put("result", "success");    } catch (Throwable e) {  e.printStackTrace();  modelMap.put("result", "exception");  }  System.out.println(mapper.writeValueAsString(modelMap));  return mapper.writeValueAsString(modelMap);  } |

|  |
| --- |
| $(document).ready(function() {  reloadList();    $("#logoutBtn").on("click", function() {  location.href = "bLogout";  });    $("#insertBtn").on("click", function() {  location.href = "bWrite";  });    $(".paging\_area").on("click", "span", function() {  $("#page").val($(this).attr("name"));  reloadList();  });    $("#searchBtn").on("click", function() {  $("#page").val("1");  reloadList();  });    $("[name='searchTxt']").on("keypress", function() {  if(event.keyCode == 13) {  $("#searchBtn").click();  return false;  }  });  });  function reloadList() {  var params = $("#actionForm").serialize();    $.ajax({  type : "post", //전송방식  url : "bListAjax", //주소  dataType : "json", //데이터 형태  data : params, //보낼데이터  success : function(res) { //성공 시 다음 함수를 실행하며, 돌아오는 값을 res란 이름으로 받는다.  if(res.result == "success") {  redrawList(res.list);  redrawPaging(res.pb);  } else {  alert("조회중 문제가 발생하였습니다.");  }  },  error : function(request, status, error) { // 실행 중 에러 발생 시  console.log("text : " + request.responseTxt); //반환텍스트  console.log("error : " + error); //에러내용  }  });  }  function redrawList(list) {  if(list.length == 0) {  var html = "";  html += "<tr>";  html += "<td colspan=\"5\">조회 결과가 없습니다.</td>";  html += "</tr>";    $(".b tbody").html(html);  } else {  var html = "";  for(var i = 0 ; i <list.length ; i++) {  html += "<tr>";  html += "<td>" + list[i].B\_NO + "</td>";  html += "<td>" + list[i].B\_TITLE + "</td>";  html += "<td>" + list[i].BM\_NM + "</td>";  html += "<td>" + list[i].B\_DT + "</td>";  html += "<td>" + list[i].B\_HIT + "</td>";  html += "</tr>";    $(".b tbody").html(html);  }  }  }  function redrawPaging(pb) {  var html = "<span name=\"1\">처음</span>&nbsp;";    if($("#page").val() == 1) {  html += "<span name=\"1\">이전</span>&nbsp;";  } else {  html += "<span name=\"" + ($("#page").val() \* 1 - 1) + "\">이전</span>&nbsp;";  }    for(var i = pb.startPcount ; i <= pb.endPcount ; i++) {  if(i == $("#page").val()) {  html += "<span name=\"" + i + "\"><b>" + i + "</b></span>&nbsp;";  } else {  html += "<span name=\"" + i + "\">" + i + "</span>&nbsp;";  }  }    if($("#page").val() == pb.maxPcount) {  html += "<span name=\"" + pb.maxPcount + "\">다음</span>&nbsp;";  } else {  html += "<span name=\"" + ($("#page").val() \* 1 + 1) + "\">다음</span>&nbsp;";  }  html += "<span name=\"" + pb.maxPcount + "\">마지막</span>";    $(".paging\_area").html(html);  } |

|  |
| --- |
| $(document).ready(function() {  /\* 엔터키 무력화 \*/  $("#actionForm input").on("keypress", function(event) {  if(event.keypress == 13) {  return false;  }  });    $("#cancelBtn").on("click", function() {  location.href ="bMain";  });    $("#writeBtn").on("click", function() {  if($.trim($("#bTitle").val()) == "") {  alert("제목을 입력하여 주세요.");  } else if($("#bCon").val() == "") {  alert("내용을 입력하여 주세요.");  } else {  var params = $("#actionForm").serialize();  /\* ajax 기본 형태 \*/  $.ajax({  type : "post", //전송방식  url : "bWriteAjax", //주소  dataType : "json", //데이터 형태  data : params, //보낼데이터  success : function(res) { //성공 시 다음 함수를 실행하며, 돌아오는 값을 res란 이름으로 받는다.  if(res.result == "success") {  location.href ="bMain";  } else {  alert("오류가 발생했습니다.");  }  },  error : function(request, status, error) { // 실행 중 에러 발생 시  console.log("text : " + request.responseTxt); //반환텍스트  console.log("error : " + error); //에러내용  }  });  }  });  }); |

|  |
| --- |
| <select id="getBCnt" parameterType="hashmap" resultType="Integer">  SELECT COUNT(\*) AS CNT  FROM B INNER JOIN BM M  ON B.BM\_NO = M.BM\_NO  WHERE B.B\_DEL = 1  <if test="searchTxt != null and searchTxt != ''">  <choose>  <when test="searchGbn == 0">  AND (  B.B\_TITLE LIKE '%' || #{searchTxt} || '%'  OR M.BM\_NM LIKE '%' || #{searchTxt} || '%'  OR B.B\_CON LIKE '%' || #{searchTxt} || '%'  )  </when>  <when test="searchGbn == 1">  AND B.B\_TITLE LIKE '%' || #{searchTxt} || '%'  </when>  <when test="searchGbn == 2">  AND M.BM\_NM LIKE '%' || #{searchTxt} || '%'  </when>  <when test="searchGbn == 3">  AND B.B\_CON LIKE '%' || #{searchTxt} || '%'  </when>  </choose>  </if>  </select>    <select id="getBList" parameterType="hashmap" resultType="hashmap">  SELECT B.B\_NO, B.B\_TITLE, B.BM\_NM, B.B\_DT, B.B\_HIT  FROM (SELECT B.B\_NO, B.B\_TITLE, M.BM\_NM, TO\_CHAR(B.B\_DT, 'YYYY-MM-DD') AS B\_DT, B.B\_HIT, ROW\_NUMBER() OVER(ORDER BY B.B\_NO DESC) AS RNUM  FROM B INNER JOIN BM M  ON B.BM\_NO = M.BM\_NO  WHERE B.B\_DEL = 1  <if test="searchTxt != null and searchTxt != ''">  <choose>  <when test="searchGbn == 0">  AND (  B.B\_TITLE LIKE '%' || #{searchTxt} || '%'  OR M.BM\_NM LIKE '%' || #{searchTxt} || '%'  OR B.B\_CON LIKE '%' || #{searchTxt} || '%'  )  </when>  <when test="searchGbn == 1">  AND B.B\_TITLE LIKE '%' || #{searchTxt} || '%'  </when>  <when test="searchGbn == 2">  AND M.BM\_NM LIKE '%' || #{searchTxt} || '%'  </when>  <when test="searchGbn == 3">  AND B.B\_CON LIKE '%' || #{searchTxt} || '%'  </when>  </choose>  </if>  ) B  WHERE B.RNUM BETWEEN #{startCnt} AND #{endCnt}  </select> |