**데이터베이스 요구사항**

**사례연구**

**평가자료**

**- 내용제목-**

수업명 :java,python 기반의 AI 활용 응용

소프트웨어 개발자 양성과정

제출일 : 2020. 11. 02

작성자 :본인이름

□프로젝트 개요

|  |  |  |
| --- | --- | --- |
| **구분** | **내용** | **비고** |
| **제목** |  |  |
| **프로젝트**  **작성기간** | ~2020.11.02 |  |
| **조원 구성** |  |  |
| **프로젝트 내용** |  |  |
| **주요기능** |  |  |
|  |  |
|  |  |
| **학습내용** |  |  |
|  |  |
|  |  |
|  |  |

□최종 보고서

|  |  |  |
| --- | --- | --- |
| **구분** | **내용** | **비고** |
| **제목** |  |  |
| **프로젝트 내용** |  |  |
| **완료내용** |  | 프로젝트  개요와 동일 |
|  | 프로젝트  개요와 동일 |
|  | 프로젝트  개요와 동일 |
| **프로젝트 활동 후기** | | |
| **좋았던 점** |  |  |
| **아쉬웠던 점 및**  **어려웠던 점** |  |  |
|  |  |
|  |  |
|  |  |
| **학습 효과** |  |  |

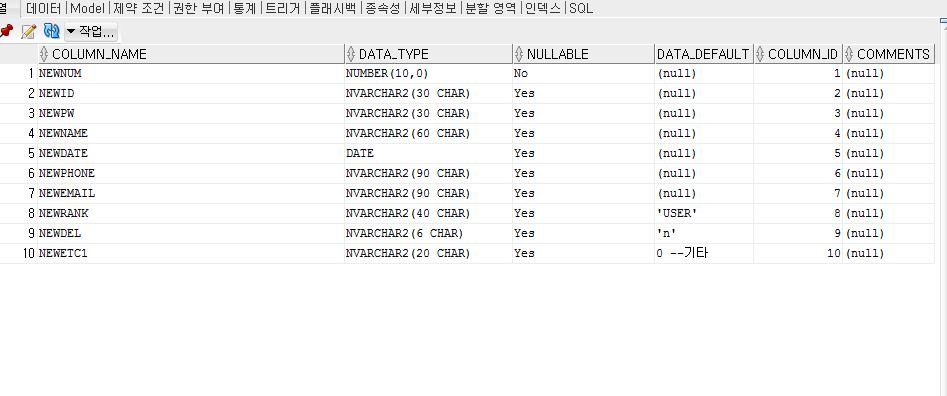
□프로젝트 세부 사항 분석 및 구조 설계

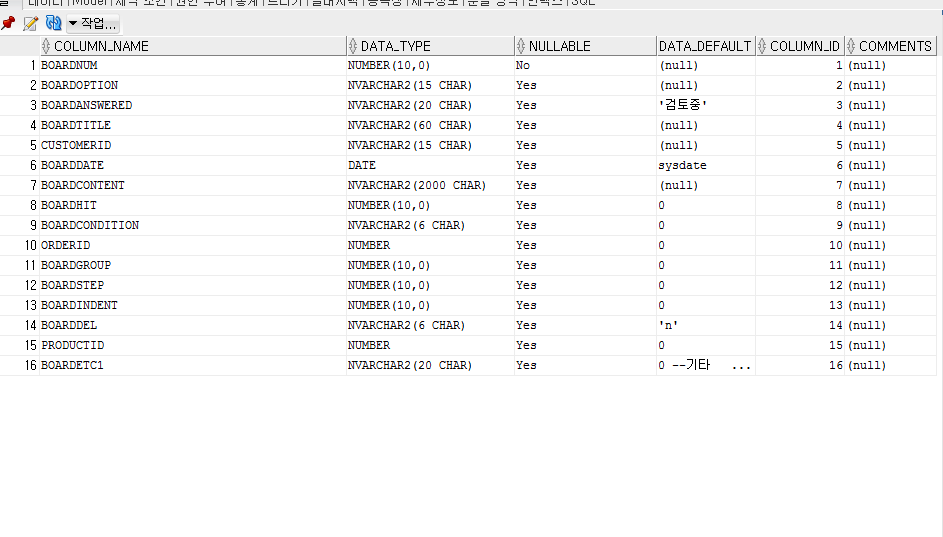
|  |  |  |  |
| --- | --- | --- | --- |
| **구분** | **주요 기능/구문** | **내용** | **입출력** |
| **1** |  |  |  |
| **2** |  |  |  |
| **3** |  |  |  |
| **4** |  |  |  |

□작성 과정, 실행결과

|  |  |  |
| --- | --- | --- |
| **구현 내용** | **결과** | **비고** |
|  |  |  |
|  |  |  |

□실행결과-1





□소스코드-1

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

pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Northwind</title>

<link rel="stylesheet"

href="https://use.fontawesome.com/releases/v5.6.3/css/all.css">

<link rel="stylesheet" href="../css/mainIndex.css">

<link rel="stylesheet" href="../css/mainOriginIndex.css">

<link rel="stylesheet" href="../css/signup.css">

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

<script type="text/javascript">

**function** signUp\_Check() { //회원가입 누를시 전체 체크

**var** form = document.registerform

//아이디 입력여부

**if** (!form.id.value) {

alert("아이디를 입력해주세요.");

**return** **false**;

}

**if**((form.id.value < "0" || form.id.value > "9") && (form.id.value < "A" || form.id.value > "Z") && (form.id.value < "a" || form.id.value > "z")){

alert("한글 및 특수문자는 아이디로 사용하실 수 없습니다.");

**return** **false**;

}

//아이디 중복체크를 했는지 여부

**if** (form.id\_Check.value == "idUncheck") {

alert("아이디 중복체크를 해주세요.");

**return** **false**;

}

//비밀번호 입력여부

**if** (!form.pwd.value) {

alert("비밀번호를 입력하세요.");

**return** **false**;

}

// 비밀번호와 비밀번호 확인에 입력된 값이 동일한지 확인

**if** (form.pwd.value != form.pwd\_Check.value) {

alert("비밀번호를 동일하게 입력하세요.");

**return** **false**;

}

// 이름 입력 여부

**if** (!form.name.value) {

alert("이름을 입력하세요.");

**return** **false**;

}

**if** (!form.birth.value) {

alert("생년월일을 입력하세요.");

**return** **false**;

}

**if** (isNaN(form.birth.value)) {

alert("생년월일은 숫자만 입력가능합니다.");

**return** **false**;

}

**if** (!form.phoneNum.value) {

alert("휴대폰 번호를 입력하세요.");

**return** **false**;

}

**if** (isNaN(form.phoneNum.value)) {

alert("휴대폰 번호는 -제외한 숫자만 입력해주세요.");

**return** **false**;

}

**if** (!form.email.value) {

alert("메일 주소를 입력하세요.");

**return** **false**;

}

**if** (!form.emailInput.value) {

alert("주소를 입력하세요.");

**return** **false**;

}

**if** (!form.agreement1.checked) {

alert("이용약관 동의 해주세요.");

**return** **false**;

}

**if** (!form.agreement2.checked) {

alert("개인정보수집 및 이용 동의 해주세요.");

**return** **false**;

}

}

//이메일 입력방식 선택

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

$('#emailSelect').change(**function**() {

$("#emailSelect option:selected").each(**function**() {

**if** ($(**this**).val() == '1') { //직접입력일 경우

$("#emailInput").val('ntt.com'); //값 초기화

$("#emailInput").attr("disabled", **false**); //활성화

} **else**{

$("#emailInput").val($(**this**).val()); //값 초기화

$("#emailInput").attr("readonly", **true**); //비활성화

}

});

});

});

**function** idCheck(){

**if**(document.registerform.id.value==""){

alert("아이디를 입력해주세요");

document.registerform.id.focus();

**return**;

}

**var** url = "../idCheck.so?userid=" + document.registerform.id.value;

window.open(url,"\_blank\_1", "toolbar=no, menubar=no, scrollbars=no, resizable=no, width=500, height=300");

}

// 아이디 입력창에 값 입력시 hidden에 idUncheck를 세팅한다.

// 이렇게 하는 이유는 중복체크 후 다시 아이디 창이 새로운 아이디를 입력했을 때

// 다시 중복체크를 하도록 한다.

**function** inputIdChk() {

document.registerform.id\_Check.value = "idUncheck";

}

</script>

</head>

<body>

<div id="wrap">

<div id="header">

<**jsp:include** page="../section/headerMain.jsp" />

</div>

<div id="content">

<div class="wrap\_join">

<h2 class="sub\_title is\_line">간편 회원가입</h2>

<span id="add\_exp">정보를 정확히 입력해주셔야 아이디나 비밀번호 분실 시 찾으실 수 있습니다. <br><em class="red\_color">입력하신 정보는 회원가입 여부에만 사용되며 저장되지 않습니다.</em></span>

<form method="post" name="registerform" action="signupNew.so" onsubmit="return signUp\_Check()">

<input type="text" class="join\_form" name="id" autocapitalize="off" maxlength="5" size=200 placeholder="아이디 영문/숫자 1~5자" onkeydown="inputIdChk()" >

<input type="button" class="validateIdSubmit" name="validateIdSubmit" onclick="idCheck()" value="중복확인">

<input type="hidden" name="id\_Check" value="idUncheck" >

<input type="password" class="join\_form" name="pwd" maxlength="12" placeholder="비밀번호 영문, 숫자, 특수문자 조합 8~12자">

<input type="password" class="join\_form" name="pwd\_Check" maxlength="12" placeholder="비밀번호 다시 입력">

<input type="text" class="join\_form" name="name" maxlength="10" placeholder="이름">

<input type="text" class="join\_form" name="birth" maxlength="8" placeholder="생년월일 예: 19940101">

<input type="text" class="join\_form" name="phoneNum" maxlength="11" placeholder="휴대폰번호 -없이 입력">

<input type="text" class="email" name="email" placeholder="이메일">

<b>@</b>

<input type="text" class="email" name="email\_Input" placeholder="직접입력" id="emailInput">

<select title="선택옵션" class="validateIdSubmit" name="emailSelect" id="emailSelect">

<option value="1">직접입력</option>

<option value="naver.com">naver.com</option>

<option value="hanmail.net">hanmail.net</option>

<option value="nate.com">nate.com</option>

<option value="hotmail.com">hotmail.com</option>

<option value="gmail.com">gmail.com</option>

</select>

<br>

<br>

<input type="checkbox" name="agreement1"> (필수) 이용약관 동의 <br> <br>

<input type="checkbox" name="agreement2"> (필수) 개인정보수집 및 이용 동의 <br> <br>

<input type="submit" class="join\_btn" value="가입완료">

</form>

<div class='cont\_find'>

<div class='find\_account'>

<a href='mainIDFindPage.jsp' class='link\_find' id='findId'>아이디

찾기</a>&nbsp;&nbsp;| <a href='mainPwdFindPage.jsp' class='link\_find'

id='findId'>&nbsp;&nbsp;비밀번호 찾기 </a>

</div>

</div>

</div>

</div>

<div id="footer">

<**jsp:include** page="../section/footerMain.jsp" />

</div>

</div>

</body>

</html>

package com.human.controller;

import java.io.IOException;

import javax.servlet.RequestDispatcher;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import javax.servlet.http.HttpSession;

import com.human.command.member.CartListOrderCommand;

import com.human.command.member.FindIDCommand;

import com.human.command.member.FindPwCommand;

import com.human.command.member.IdCheckCommand;

import com.human.command.member.Index1By1PageCommand;

import com.human.command.member.Index9Command;

import com.human.command.member.IndexSearchQnaListDeleteCommand;

import com.human.command.member.IndexSearchQnaListPageCommand;

import com.human.command.member.IndexSearchQnaListUpdateCommand;

import com.human.command.member.MemberCommand;

import com.human.command.member.MyQnaPageCommand;

import com.human.command.member.NewJoinCommand;

import com.human.command.member.NewLoginCommand;

import com.human.command.member.NoticeCommand;

import com.human.command.member.OrderSelectCommand;

import com.human.command.member.QnaBoardListCommand;

import com.human.command.member.QnaBoardReplyCommand;

import com.human.command.member.QnaCommand;

import com.human.command.member.QnaInsertCommand;

import com.human.command.member.ReviewDeleteCommand;

import com.human.command.member.ReviewInsertCommand;

import com.human.command.member.ReviewUpdateCommand;

import com.human.command.member.ReviewUpdateOkCommand;

import com.human.command.member.User\_selectCommand;

import com.human.mail.MailSend;

/\*\*

\* Servlet implementation class idCheckServlet

\*/

@WebServlet("\*.so")

public class MemberController extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#HttpServlet()

\*/

public MemberController() {

super();

// TODO Auto-generated constructor stub

}

/\*\*

\* @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse

\* response)

\*/

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

request.setCharacterEncoding("utf-8");

response.setContentType("text/html;charset=UTF-8");

MemberCommand bCom = null;

// 사용자에게 보여줄 view 설정

String viewPage = null;

String uri = request.getRequestURI();

System.out.println(uri);

String conPath = request.getContextPath();

System.out.println(conPath);

String com = uri.substring(conPath.length());

System.out.println(com);

if (com.equals("/idCheck.so")) {

//ID중복체크

viewPage = "member/signup\_CheckId.jsp";

bCom = new IdCheckCommand();

bCom.execute(request, response);

RequestDispatcher dispatcher = request.getRequestDispatcher(viewPage);

dispatcher.forward(request, response);

} else if (com.equals("/member/signupNew.so")) {

//회원가입

viewPage = "main\_Newjoin.jsp";

bCom = new NewJoinCommand();

bCom.execute(request, response);

RequestDispatcher dispatcher = request.getRequestDispatcher(viewPage);

dispatcher.forward(request, response);

} else if (com.equals("/member/login.so")) {

//로그인

bCom = new NewLoginCommand();

bCom.execute(request, response);

String temp = (String) request.getAttribute("temp");

if (temp == "1") {

String name = (String) request.getAttribute("name");

String rank = (String) request.getAttribute("rank");

String id = (String) request.getAttribute("id");

HttpSession session = request.getSession();

session.setAttribute("login\_Name", name);

session.setAttribute("login\_Id", id);

session.setAttribute("login\_Rank", rank);

session.setMaxInactiveInterval(10000);// 세션 유지 시간 자주 꺼져서 오래써봐야 의미없음

response.sendRedirect("../main/main.jsp");

} else {

viewPage = "../member/main\_login\_fail.jsp";

response.sendRedirect("../member/main\_login\_fail.jsp");

}

} else if (com.equals("/logout.so")) {

//로그아웃

viewPage = "main.jsp";

HttpSession session = request.getSession();

session.invalidate();

response.sendRedirect("main/main.jsp");

} else if (com.equals("/manager/logout.so")) {

//관리자 페이지 로그아웃

viewPage = "main.jsp";

HttpSession session = request.getSession();

session.invalidate();

response.sendRedirect("../main/main.jsp");

} else if (com.equals("/member/mainPwdFindPage.so")) {

// 비밀번호 찾기

bCom = new FindPwCommand();

bCom.execute(request, response);

if (request.getAttribute("newPw") == null) {

viewPage = "main\_Pwd\_Check\_fail.jsp";

} else {

// 메일 작업

String address = (String) request.getAttribute("newEmail");

String pwForSend = (String) request.getAttribute("newPw");

MailSend mailSend = new MailSend();

mailSend.PWmailSend(address, pwForSend);

viewPage = "main\_Pwd\_Check\_Succeed.jsp";

}

RequestDispatcher dispatcher = request.getRequestDispatcher(viewPage);

dispatcher.forward(request, response);

} else if (com.equals("/member/mainIDFindPage.so")) {

// 아이디 찾기

bCom = new FindIDCommand();

bCom.execute(request, response);

if (request.getAttribute("newID") == null) {

viewPage = "main\_Id\_Check\_fail.jsp";

} else {

// 메일 작업

String address = (String) request.getAttribute("newEmail");

String IDForSend = (String) request.getAttribute("newID");

System.out.println("메일작업 \n" + address + "\n" + IDForSend);

MailSend mailSend = new MailSend();

mailSend.IDmailSend(address, IDForSend);

viewPage = "main\_Id\_Check\_Succeed.jsp";

}

RequestDispatcher dispatcher = request.getRequestDispatcher(viewPage);

dispatcher.forward(request, response);

} else if (com.equals("/member/myqna\_view.so")) {

//고객센터 - 나의문의내역

viewPage = "customer8\_myqna.jsp";

bCom = new MyQnaPageCommand();

bCom.execute(request, response);

RequestDispatcher dispatcher = request.getRequestDispatcher(viewPage);

dispatcher.forward(request, response);

} else if (com.equals("/cart/login.so")) {

//장바구니 진입시 로그인체크

bCom = new NewLoginCommand();

bCom.execute(request, response);

String temp = (String) request.getAttribute("temp");

if (temp == "1") {

String name = (String) request.getAttribute("name");

String rank = (String) request.getAttribute("rank");

String id = (String) request.getAttribute("id");

HttpSession session = request.getSession();

session.setAttribute("login\_Name", name);

session.setAttribute("login\_Rank", rank);

session.setAttribute("login\_Id", id);

System.out.println("login\_Id : " + id);

session.setMaxInactiveInterval(10000);// 세션 유지 시간 자주 꺼져서 오래써봐야 의미없음

response.sendRedirect("../cart/cartSelect.po");

} else {

viewPage = "../member/main\_login\_fail.jsp";

response.sendRedirect("../member/main\_login\_fail.jsp");

}

}else if (com.equals("/member/notice\_view.so")) {

//footer 공지사항

viewPage = "notice.jsp";

bCom = new NoticeCommand();

bCom.execute(request, response);

RequestDispatcher dispatcher = request.getRequestDispatcher(viewPage);

dispatcher.forward(request, response);

}else if(com.equals("/myPage/index9.so")) {

//myPage 상품평내역

viewPage="/myPage/index9.jsp";

System.out.println("00");

bCom=new Index9Command();

bCom.execute(request, response);

RequestDispatcher dispatcher = request.getRequestDispatcher(viewPage);

dispatcher.forward(request, response);

}else if(com.equals("/myPage/index9\_2.so")) {

//myPage 상품평쓰기

viewPage="/myPage/index9.so";

System.out.println("9-2");

bCom=new ReviewInsertCommand();

bCom.execute(request, response);

RequestDispatcher dispatcher = request.getRequestDispatcher(viewPage);

dispatcher.forward(request, response);

}else if(com.equals("/myPage/index9\_update.so")) {

//myPage 상품평 수정

System.out.println("1");

bCom=new ReviewUpdateCommand();

viewPage="/myPage/index9\_1.jsp";

bCom.execute(request, response);

RequestDispatcher dispatcher = request.getRequestDispatcher(viewPage);

dispatcher.forward(request, response);

}else if(com.equals("/myPage/reviewUpdate.so")) {

//myPage 상품평 수정완료

System.out.println("2");

bCom=new ReviewUpdateOkCommand();

viewPage="/myPage/index9.so";

System.out.println("3");

bCom.execute(request, response);

RequestDispatcher dispatcher = request.getRequestDispatcher(viewPage);

dispatcher.forward(request, response);

}else if(com.equals("/myPage/index9\_delete.so")) {

//myPage 상품평삭제

System.out.println("delete");

bCom=new ReviewDeleteCommand();

viewPage="/myPage/index9.so";

System.out.println("4");

bCom.execute(request, response);

RequestDispatcher dispatcher = request.getRequestDispatcher(viewPage);

dispatcher.forward(request, response);

}else if(com.equals("/myPage/index8.so")) {

//myPage 상품평쓰기 선택

System.out.println("0");

bCom=new OrderSelectCommand();

viewPage="/myPage/index8.jsp";

bCom.execute(request, response);

RequestDispatcher dispatcher = request.getRequestDispatcher(viewPage);

dispatcher.forward(request, response);

}else if(com.equals("/board/qnaBoard.so")){

//productDetails 상품문의 글쓰기

bCom = new QnaCommand();

bCom.execute(request, response);

response.sendRedirect("askqna.jsp?check=1");

}else if (com.equals("/myPage/indexSearchQnaList\_view.so")) {

//myPage 상품문의내역 확인

viewPage = "indexSearch.jsp";

bCom = new IndexSearchQnaListPageCommand();

bCom.execute(request, response);

RequestDispatcher dispatcher = request.getRequestDispatcher(viewPage);

dispatcher.forward(request, response);

}else if (com.equals("/myPage/indexqna1by1\_view.so")) {

//myPage 일대일문의내역 확인

viewPage = "indexqna1by1.jsp";

bCom = new Index1By1PageCommand();

bCom.execute(request, response);

RequestDispatcher dispatcher = request.getRequestDispatcher(viewPage);

dispatcher.forward(request, response);

} else if (com.equals("/myPage/qnaUpdate.so")) {

//myPage 상품문의내역 수정

viewPage = "indexSearchQnaList\_view.so";

bCom = new IndexSearchQnaListUpdateCommand();

bCom.execute(request, response);

RequestDispatcher dispatcher = request.getRequestDispatcher(viewPage);

dispatcher.forward(request, response);

} else if (com.equals("/myPage/qnaDelete.so")) {

//myPage 상품문의내역 삭제

viewPage = "indexSearchQnaList\_view.so";

bCom = new IndexSearchQnaListDeleteCommand();

bCom.execute(request, response);

RequestDispatcher dispatcher = request.getRequestDispatcher(viewPage);

dispatcher.forward(request, response);

}else if(com.equals("/member/select\_goods.so")){

//고객센터 일대일문의 중 상품선택버튼

viewPage = "select\_goods.jsp";

bCom = new User\_selectCommand();

System.out.println("select\_goods.jsp연결");

bCom.execute(request, response);

RequestDispatcher dispatcher = request.getRequestDispatcher(viewPage);

dispatcher.forward(request, response);

}else if(com.equals("/member/select\_goods2.so")){

//고객센터 일대일문의 중 상품선택->장바구니 탭

viewPage = "select\_goods2.jsp";

bCom = new CartListOrderCommand();

System.out.println("select\_goods2.jsp연결");

bCom.execute(request, response);

RequestDispatcher dispatcher = request.getRequestDispatcher(viewPage);

dispatcher.forward(request, response);

}else if(com.equals("/member/customer7\_1by1.so")){

//고객센터 일대일문의

viewPage = "customer7\_1by1.jsp";

bCom = new User\_selectCommand();

bCom.execute(request, response);

RequestDispatcher dispatcher = request.getRequestDispatcher(viewPage);

dispatcher.forward(request, response);

}else if(com.equals("/member/customer\_inquiry.so")) {

//footer 일대일문의

viewPage= "myqna\_view.so";

bCom=new QnaInsertCommand();

bCom.execute(request, response);

RequestDispatcher dispatcher = request.getRequestDispatcher(viewPage);

dispatcher.forward(request, response);

}

}

/\*\*

\* @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse

\* response)

\*/

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

// TODO Auto-generated method stub

doGet(request, response);

}

}

package com.human.command.member;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import com.human.dao.MemberDao;

public class NewJoinCommand implements MemberCommand {

@Override

public void execute(HttpServletRequest request, HttpServletResponse response) {

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

System.out.println(id);

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

System.out.println(pwd);

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

System.out.println(name);

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

System.out.println(birth);

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

System.out.println(phoneNum);

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

System.out.println(email);

String emailInput = request.getParameter("email\_Input");

System.out.println(emailInput);

String emailTemp = email+"@"+emailInput;

MemberDao mDao = new MemberDao();

mDao.insertNewJoin(id,pwd,name,birth,phoneNum,emailTemp);

request.setAttribute("name", name);

}

}