Team Project

JSP & Servlet 기반의 MVC Model2를 이용한 업무관리페이지 설계 및 구현

훈련과정명: 스마트팩토리 공급망 구축을 위한 핵심 실무인재 양성과정

개발자:김태완, 정성현, 공성현, 한광진, 심준수

목차

Ⅰ. 프로젝트 주제

1. 주제

사내 업무를 관리할 수 있는 웹 페이지 작성

1. 주제 선정 이유

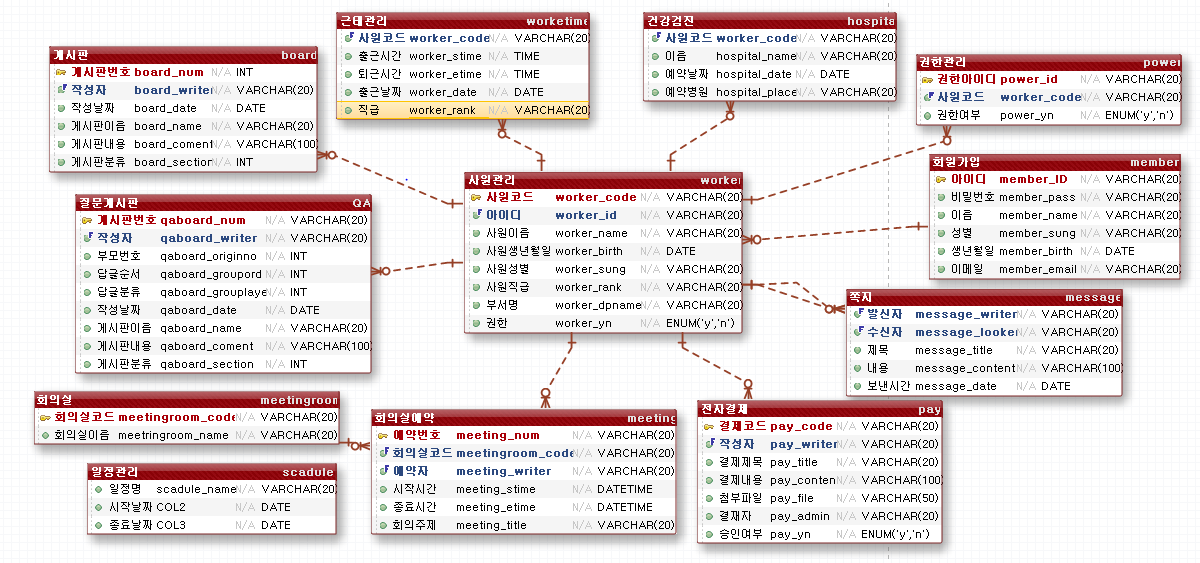
사내 업무 관리 페이지를 만들어봄으로써 수업 시간에 배웠던 내용을 직접 구현할 수 있다. 업무 관리 페이지는 다양한 기능을 요구하기 때문에 단순히 배웠던 내용에 그치지 않고 새로운 정보를 습득해야 하기 때문에 앞으로 개발자의 길을 걷게 될 예비 개발자들이 스스로 공부하고 문제를 해결해가는 습관을 기를 수 있다.

업무 관리 페이지는 사원 관리와, 일정 관리, 전자 결재, 게시판 등 다양한 기능을 요구하는 복합적인 프로젝트이다. 복잡한 기능을 구현하기 위해 다양한 시행착오를 겪으면서 문제를 해결함으로써 프로그래밍 실력을 키우는 과정이 될 것이다.

Ⅱ. 프로젝트 구조

1. 개요

1. 데이터베이스
2. 구조도



1. 데이터베이스 테이블 생성문과 기능 설명

-- 사원관리

CREATE TABLE `worker` (

`worker\_code` INT auto\_increment primary key, -- 사원코드

`worker\_id` VARCHAR(20) NOT NULL, -- 아이디

`worker\_name` VARCHAR(20) NOT NULL, -- 사원이름

`worker\_birth` DATE NOT NULL, -- 사원생년월일

`worker\_sung` VARCHAR(20) NOT NULL, -- 사원성별

`worker\_rank` VARCHAR(20) NOT NULL, -- 사원직급

`worker\_dpname` VARCHAR(20) NULL, -- 부서명

`worker\_yn` ENUM('y','n') NULL -- 권한);

-- 근태관리

CREATE TABLE `worketime` (

`worker\_code` INT auto\_increment primary key, -- 사원코드

`worker\_stime` DATETIME NULL, -- 출근시간

`worker\_etime` DATETIME NULL, -- 퇴근시간

`worker\_date` DATE NULL, -- 출근날짜

`worker\_rank` VARCHAR(20) NULL -- 직급);

-- 회원가입

CREATE TABLE `member` (

`member\_id` VARCHAR(20) NOT NULL, -- 아이디

`member\_pw` VARCHAR(20) NOT NULL, -- 비밀번호

`member\_name` VARCHAR(20) NOT NULL, -- 이름

`member\_birth` varchar(30) NOT NULL, -- 생년월일

`member\_gender` VARCHAR(20) NOT NULL, -- 성별

`member\_email` VARCHAR(30) NOT NULL -- 이메일);

ALTER TABLE `member`

ADD CONSTRAINT `PK\_member` -- 회원가입 기본키

PRIMARY KEY (`member\_ID` -- 아이디);

-- 게시판

CREATE TABLE `board` (

`board\_num` INT auto\_increment primary key, -- 게시판번호

`board\_writer` INT NULL, -- 작성자

`board\_date` DATE NULL, -- 작성날짜

`board\_name` VARCHAR(20) NULL, -- 게시판이름

`board\_coment` VARCHAR(100) NULL, -- 게시판내용

`board\_section` INT NULL -- 게시판분류);

-- 건강검진

CREATE TABLE `hospital` (

`worker\_code` INT auto\_increment primary key, -- 사원코드

`hospital\_name` VARCHAR(20) NOT NULL, -- 이름

`hospital\_date` DATE NOT NULL, -- 예약날짜

`hospital\_place` VARCHAR(20) NOT NULL -- 예약병원);

-- 질문게시판

CREATE TABLE `QA` (

`qa\_num` INT auto\_increment primary key, -- 게시판번호

`qa\_id` varchar(20), -- 작성자

`qa\_subject` varchar(50), -- 글 제목

`qa\_content` text, -- 글 내용

`qa\_file` varchar(50), -- 파일

`qa\_re\_ref` int, -- 답글로 묶기

`qa\_re\_lev` int, -- 들여쓰기

`qa\_re\_seq` int, -- 답글 순서

`qa\_readcount` int, -- 조회수

`qa\_date` date -- 게시글 올린 시간);

-- 일정관리

CREATE TABLE `scaduler` (

`scadule\_name` VARCHAR(20) NULL, -- 일정명

`COL2` DATE NULL, -- 시작날짜

`COL3` DATE NULL -- 종료날짜);

-- 회의실예약

CREATE TABLE `meeting` (

`meeting\_num` INT auto\_increment primary key, -- 예약번호

`meetingroom\_code` INT NOT NULL, -- 회의실코드

`meeting\_writer` INT NOT NULL, -- 예약자

`meeting\_stime` DATETIME NOT NULL, -- 시작시간

`meeting\_etime` DATETIME NOT NULL, -- 종료시간

`meeting\_title` VARCHAR(20) NOT NULL -- 회의주제);

-- 회의실

CREATE TABLE `meetingroom` (

`meetingroom\_code` INT auto\_increment primary key, -- 회의실코드

`meetringroom\_name` VARCHAR(20) NULL -- 회의실이름);

-- 전자결제

CREATE TABLE `pay` (

`pay\_code` INT auto\_increment primary key, -- 결제코드

`pay\_writer` INT NOT NULL, -- 작성자

`pay\_title` VARCHAR(20) NULL, -- 결제제목

`pay\_content` VARCHAR(100) NULL, -- 결제내용

`pay\_file` VARCHAR(50) NULL, -- 첨부파일

'pay\_time' DATETIME NULL, -- 결제신청시간

`pay\_admin` VARCHAR(20) NULL, -- 결재자

'pay\_admintime' DATETIME NULL, -- 결재자시간

`pay\_yn` ENUM('y','n') NULL -- 승인여부);

-- 권한관리

CREATE TABLE `power` (

`power\_id` INT auto\_increment primary key, -- 권한아이디

`worker\_code` INT NULL, -- 사원코드

`power\_yn` ENUM('y','n') NULL -- 권한여부);

-- 쪽지

CREATE TABLE `message` (

`message\_writer` INT NULL, -- 발신자

`message\_looker` INT NULL, -- 수신자

`message\_title` VARCHAR(20) NULL, -- 제목

`message\_content` VARCHAR(100) NULL, -- 내용

`message\_date` DATE NULL -- 보낸시간);

-- 사원관리

ALTER TABLE `worker`

ADD CONSTRAINT `FK\_member\_TO\_worker` -- 회원가입 -> 사원관리

FOREIGN KEY (`worker\_id` )

REFERENCES `member` ( `member\_ID` -- 아이디);

-- 근태관리

ALTER TABLE `worketime`

ADD CONSTRAINT `FK\_worker\_TO\_worketime` -- 사원관리 -> 근태관리

FOREIGN KEY (`worker\_code` -- 사원코드)

REFERENCES `worker` ( -- 사원관리 `worker\_code` -- 사원코드);

-- 게시판

ALTER TABLE `board`

ADD CONSTRAINT `FK\_worker\_TO\_board` -- 사원관리 -> 게시판

FOREIGN KEY (`board\_writer` -- 작성자)

REFERENCES `worker` ( -- 사원관리`worker\_code` -- 사원코드);

-- 건강검진

ALTER TABLE `hospital`

ADD CONSTRAINT `FK\_worker\_TO\_hospital` -- 사원관리 -> 건강검진

FOREIGN KEY (`worker\_code` -- 사원코드)

REFERENCES `worker` ( -- 사원관리 `worker\_code` -- 사원코드);

-- 질문게시판과 회원가입 테이블 외래키 설정

ALTER TABLE `QA`

ADD CONSTRAINT `FK\_worker\_TO\_QA` -- 사원관리 -> 질문게시판

FOREIGN KEY (`qa\_id` -- 작성자)

REFERENCES `worker` ( -- 사원관리 `worker\_code` -- 사원코드);

ALTER TABLE `QA` ADD CONSTRAINT `FK\_member\_TO\_QA` FOREIGN KEY(qa\_id) REFERENCES member(member\_id);

-- 회의실예약

ALTER TABLE `meeting`

ADD CONSTRAINT `FK\_worker\_TO\_meeting` -- 사원관리 -> 회의실예약

FOREIGN KEY (`meeting\_writer` -- 예약자)

REFERENCES `worker` ( -- 사원관리 `worker\_code` -- 사원코드);

ALTER TABLE `meeting`

ADD CONSTRAINT `FK\_meetingroom\_TO\_meeting` -- 회의실 -> 회의실예약

FOREIGN KEY (`meetingroom\_code` -- 회의실코드)

REFERENCES `meetingroom` ( -- 회의실 `meetingroom\_code` -- 회의실코드);

-- 전자결제

ALTER TABLE `pay`

ADD CONSTRAINT `FK\_worker\_TO\_pay` -- 사원관리 -> 전자결제

FOREIGN KEY (`pay\_writer` -- 작성자)

REFERENCES `worker` ( -- 사원관리`worker\_code` -- 사원코드);

-- 권한관리

ALTER TABLE `power`

ADD CONSTRAINT `FK\_worker\_TO\_power` -- 사원관리 -> 권한관리

FOREIGN KEY (`worker\_code` -- 사원코드)

REFERENCES `worker` ( -- 사원관리 `worker\_code` -- 사원코드);

-- 쪽지

ALTER TABLE `message`

ADD CONSTRAINT `FK\_worker\_TO\_message` -- 사원관리 -> 쪽지

FOREIGN KEY (`message\_writer` -- 발신자)

REFERENCES `worker` ( -- 사원관리`worker\_code` -- 사원코드);

ALTER TABLE `message`

ADD CONSTRAINT `FK\_worker\_TO\_message2` -- 사원관리 -> 쪽지2

FOREIGN KEY (`message\_looker` -- 수신자)

REFERENCES `worker` ( -- 사원관리`worker\_code` -- 사원코드);

Ⅲ. 팀 구성 및 일정

1. 업무 분담

김태완 – 로그인, 질문 게시판

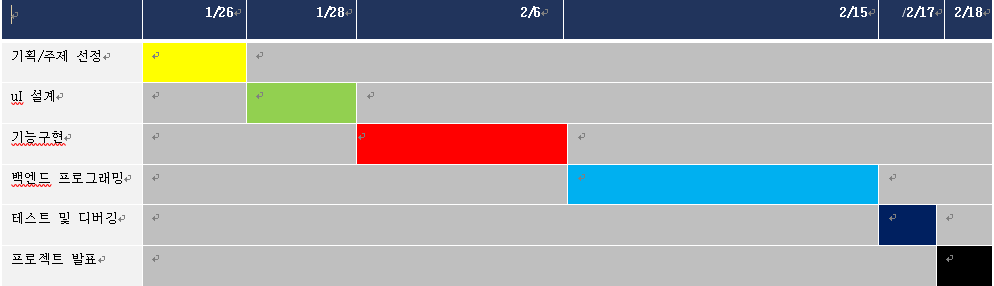
정성현 – 알림

공성현 – 전자결재

한광진 – 사원관리, 근태관리

심준수 – 일정관리

1. 개발 일정



Ⅳ. 스토리보드

Ⅴ. 소스 코드

|  |  |
| --- | --- |
| Controller.java | com.ts.controller; |
| package com.ts.controller;  import java.io.IOException;  import javax.servlet.ServletException;  import javax.servlet.http.HttpServletRequest;  import javax.servlet.http.HttpServletResponse;  public interface Controller {  public String requestHandler(HttpServletRequest request, HttpServletResponse response)  throws ServletException, IOException;  } | |

|  |  |
| --- | --- |
| BoardAddController.java | com.ts.controller; |
| package com.ts.controller;  import java.io.IOException;  import java.io.PrintWriter;  import javax.servlet.ServletContext;  import javax.servlet.ServletException;  import javax.servlet.http.HttpServletRequest;  import javax.servlet.http.HttpServletResponse;  import com.ts.model.QaBoardDAO;  import com.ts.model.QaBoardDTO;  import com.oreilly.servlet.MultipartRequest;  import com.oreilly.servlet.multipart.DefaultFileRenamePolicy;  public class BoardAddController implements Controller{  @Override  public String requestHandler(HttpServletRequest request, HttpServletResponse response)  throws ServletException, IOException {  String realFolder="";  String saveFolder="/boardUpload";  int fileSize=5\*1024\*1024;  ServletContext context=request.getServletContext();  realFolder=context.getRealPath(saveFolder);  //System.out.println(realFolder);    MultipartRequest multi=new MultipartRequest(request, realFolder, fileSize, "utf-8", new DefaultFileRenamePolicy());    QaBoardDTO dto= new QaBoardDTO();  dto.setQa\_id(multi.getParameter("qa\_id"));  dto.setQa\_subject(multi.getParameter("qa\_subject"));  dto.setQa\_content(multi.getParameter("qa\_content"));  //System.out.println(multi.getOriginalFileName((String) multi.getFileNames().nextElement()));  dto.setQa\_file(multi.getOriginalFileName((String) multi.getFileNames().nextElement()));    QaBoardDAO dao= new QaBoardDAO();  int succ=dao.boardInsert(dto);    response.setContentType("text/html; charset=utf-8");  PrintWriter out= response.getWriter();  if(succ>0) {  out.println("<script>alert('등록 성공');");  out.println("location.href='boardList.do';</script>");  } else {  out.println("<script>alert('등록 실패');");  out.println("location.href='boardList.do';</script>");  }  return null;  }  } | |

|  |  |
| --- | --- |
|  |  |
|  | |

|  |  |
| --- | --- |
|  |  |
|  | |

|  |  |
| --- | --- |
|  |  |
|  | |

|  |  |
| --- | --- |
|  |  |
|  | |

|  |  |
| --- | --- |
|  |  |
|  | |

|  |  |
| --- | --- |
|  |  |
|  | |

|  |  |
| --- | --- |
|  |  |
|  | |